MICRONAS

Application Note

HAL® 8xy, HAL 100x

Programmer Board

APN000046_03EN
Edition Sept. 19, 2013

Copyright, Warranty, and Limitation of Liability

The information and data contained in this document are believed to be accurate and reliable. The software and proprietary information contained therein may be protected by copyright, patent, trademark and/or other intellectual property rights of Micronas. All rights not expressly granted remain reserved by Micronas.

Micronas assumes no liability for errors and gives no warranty representation or guarantee regarding the suitability of its products for any particular purpose due to these specifications.

By this publication, Micronas does not assume responsibility for patent infringements or other rights of third parties which may result from its use. Commercial conditions, product availability and delivery are exclusively subject to the respective order confirmation.

Any information and data which may be provided in the document can and do vary in different applications, and actual performance may vary over time.

All operating parameters must be validated for each customer application by customers' technical experts. Any new issue of this document invalidates previous issues. Micronas reserves the right to review this document and to make changes to the document's content at any time without obligation to notify any person or entity of such revision or changes. For further advice please contact us directly.

Do not use our products in life-supporting systems, military, aviation, or aerospace applications! Unless explicitly agreed to otherwise in writing between the parties, Micronas' products are not designed, intended or authorized for use as components in systems intended for surgical implants into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the product could create a situation where personal injury or death could occur.

No part of this publication may be reproduced, photocopied, stored on a retrieval system or transmitted without the express written consent of Micronas.

Micronas Trademarks

– HAL

Micronas Patents

Sensor programming with VDD-Modulation protected by Micronas Patent No. EP 0 953 848.

Third-Party Trademarks

All other brand and product names or company names may be trademarks of their respective companies.

Contents

Page	Section	Title
5	**1.**	**Introduction**
5	1.1.	Features
6	1.2.	Supported Micronas Hall-Effect Sensors
8	**2.**	**Functional Description**
8	2.1.	Software
8	2.2.	Firmware
8	2.3.	Power-On Self-Test
9	2.4.	The COM Port Parameters
9	2.5.	Connectors
10	2.5.1.	Pinning of the Interface Connector
10	2.6.	LED Description
11	2.7.	Error Flag Codes
12	**3.**	**Usage of the Hall Programmer V 5.1**
12	3.1.	Tips for Safe Programming of the Sensors
13	3.2.	Recommended Operating Conditions
14	3.3.	Recommended Wiring
15	3.4.	Maintenance and Calibration
16	3.4.1.	Firmware Update
17	**4.**	**Operation Mode 0**
18	4.1.	Definition of the Protocol
19	4.2.	The Board Commands BCMD
20	4.2.1.	Monitoring of the Programming Voltage
21	4.3.	Programming of the Sensor
21	4.3.1.	Definition of Programming Pulses
21	4.3.2.	Definition of the Telegram
24	4.3.3.	Telegram Codes
25	4.3.4.	Number Formats
26	4.3.5.	Register Information (HAL805/HAL815/HAL817)
28	4.3.6.	Programming Information
29	4.4.	Data Formats
30	4.5.	The Programming Procedure
32	4.5.1.	The Calibration Procedure
35	**5.**	**Operation Mode 1**
36	5.1.	Definition of the Protocol
37	5.2.	The Board Commands BCMD
38	5.2.0.1.	HAL805, HAL815, HAL817, HAL810 and HAL1000
39	5.2.0.2.	HAL82x, HAL83x, HAL880, HAL1002
39	5.2.1.	Monitoring of the Programming Voltage
40	5.3.	Programming of the Sensor
40	5.3.1.	Definition of Programming Pulses
40	5.3.2.	Definition of the Telegram
43	5.3.3.	Telegram Codes
44	5.3.4.	Number Formats
46	5.3.5.	Register Information (HAL805 / HAL815 / HAL817)
48	5.3.6.	Register Information (HAL82x / HAL83x / HAL880 / HAL1002)
50	5.3.7.	Programming Information
51	5.4.	Data Formats
53	5.5.	The Programming Procedure
53	5.5.1.	Programming Procedure for HAL805, HAL810, HAL815, HAL817 and HAL1000
55	5.5.2.	The Calibration Procedure
58	5.5.3.	Programming Procedure for HAL82x, HAL83x, HAL880 and HAL1002
59	5.5.4.	The Calibration Procedure

Contents, continued

Page	Section	Title
62	**6.**	**Operation Mode 2**
63	6.1.	Definition of the Protocol
64	6.2.	The Board Commands BCMD
66	6.2.1.	Monitoring of the Supply Voltage
67	6.3.	Programming of the Sensor
67	6.3.1.	Definition of Programming Pulses
67	6.3.2.	Definition of the Telegram
71	6.3.3.	Telegram Codes
72	6.3.4.	Number Formats
73	6.3.5.	Register Information
75	6.3.6.	Programming Information
76	6.4.	Data Formats
77	6.5.	The Programming Procedure
78	6.5.1.	The Programmable Output Characteristic
79	6.5.2.	The Calibration Procedure
83	**7.**	**Operation Mode 3**
84	7.1.	Definition of the Protocol
85	7.2.	The Board Commands BCMD
86	7.2.1.	Monitoring of the Supply Voltage
87	7.3.	Programming of the Sensor
87	7.3.1.	Definition of Programming Pulses
87	7.3.2.	Definition of the Telegram
91	7.3.3.	Telegram Codes
92	7.3.4.	Number Formats
93	7.3.5.	Register Information
94	7.3.6.	Programming Information
95	7.4.	Data Formats
96	7.5.	The Programming Procedure
97	7.5.1.	The Programmable Output Characteristic
98	7.5.2.	The Calibration Procedure
102	**8.**	**Application Note History**

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

1. Introduction

The Hall programmer board V 5.1 is a general-purpose programming interface which is capable of addressing the programmable Micronas Hall-effect sensor families HAL 8xy and HAL 100x.

The Hall programmer board V 5.1 is fully software compatible to the Hall programmer board V 4.1. Hence, it is possible to replace older boards by V 5.1 while keeping the software programmer.

1.1. Features

- Communication with a PC by serial interface (RS232)
- Data transfer to/from PC board at a Baud rate of up to 57600
- Supervised programming of the Hall-effect sensors
- Output voltage level control
- Up to four Hall-effect sensors can be connected in parallel
- Easy firmware upgrade possible

1.2. Supported Micronas Hall-Effect Sensors

The board supports all programmable Micronas Hall sensors, each having its own special requirements with respect to the programmer board.

Therefore, the board can be run in different operation modes (board modes), which are addressed separately in subsequent sections of this application note.

Table 1–1: Overview Hall sensors and operation modes

Type	Operation Mode	Reference
HAL805	0 (Emulation V 4.1) 1	Section 4 Section 5
HAL810	0 (Emulation V 4.1) 1	Section 4 Section 5
HAL815	0 (Emulation V 4.1) 1	Section 4 Section 5
HAL817	0 (Emulation V 4.1) 1	Section 4 Section 5
HAL82x	1	Section 5
HAL83x	1	Section 5
HAL855	2	Section 6
HAL856	3	Section 7
HAL880	1	Section 5
HAL1000	0 (Emulation V 4.1) 1	Section 4 Section 5
HAL1002	1	Section 5

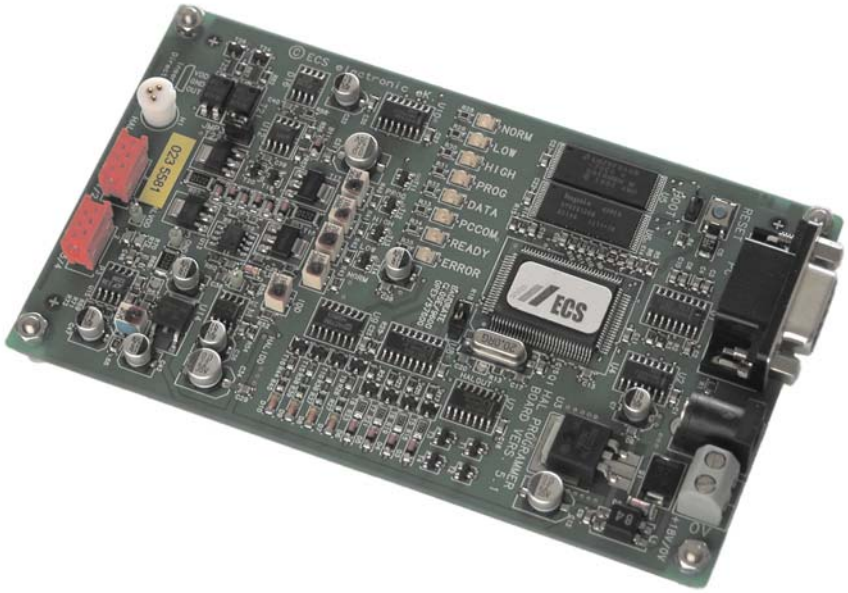

Fig. 1–1: Top view of the Hall programmer board V 5.1

2. Functional Description

The Hall programmer board V 5.1 serves as a communication interface between a PC and the programmable Hall sensor connected to the board.

With a specific programming software, command strings are sent to the board, which generate the serial protocol pattern for the sensor. After sending the protocol, the board reads back the answer of the Hall sensor or – in case of a communication error – generates an error flag.

Depending on the command, the board can send the answer of the Hall sensor back to the PC.

The Hall programmer board V 5.1 can be connected to a serial port of a PC (COM1, COM2, COM3, or COM4) using a 1:1 cable with SUB-D-9 plugs.

2.1. Software

For each of the programmable Hall sensor families, a specific PC software exists. This software provides a graphical user interface based on Microsoft Visual Basic or LabView.

For detailed information on the software, please refer to the software documentation enclosed in the software installation CD.

2.2. Firmware

The firmware of the board is stored in an on-board flash memory. The current version number of the firmware can be read out by sending a “v” command followed by a “t” command (see Table 4–2).

If the Micronas VB software is used, the firmware version number is read out by selecting the menu item:
Help → About.

It is easily possible to do a firmware update (see Section 3.4.1).

2.3. Power-On Self-Test

Firmware version 1.27 or higher provides a power-on self-test procedure:

– After setting the supply voltage or pushing the reset button, the board performs a self-test and measures all voltage levels.

– In case of an error, the status flag is set and the error LED ignites.

– If this happens, disconnect all sensors and try again. If the error persists, please contact your supplier.

2.4. The COM Port Parameters

The basic parameters for the setup of the serial communication are described in Table 2–1.

Table 2–1: COM port parameters

Parameter	Value	Remarks
Baud rate	9600 57600	Jumper 'BaudRate' set Jumper 'BaudRate' open
Data bits	8	
Stop bits	1	
Parity	EVEN	
Flow control	NONE	

2.5. Connectors

Up to four sensors can be connected to the programmer board V 5.1. For this purpose, two 6-pin connectors, HAL 1/2 and HAL 3/4, are supplied.

Note: In case of HAL 810, HAL 856 and HAL 1000 only one sensor can be connected to the board. These sensors do not support the required multi-programming function.

Alternatively, one Hall sensor can be inserted in the three-pin socket HAL beneath the connector HAL 1/2. The pins of this socket are connected parallel to pins 1, 3, and 5 of the connector HAL 1/2.

The male plug (**Amp 215083**) corresponding to the red connectors HAL 1/2 and HAL 3/4 can be ordered from Bürklin with the order no. **58F462**.

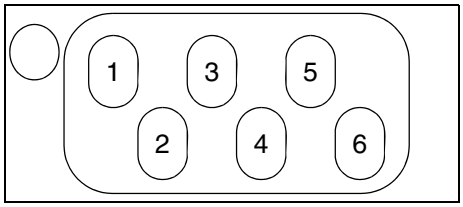

Fig. 2–1: Interface connector HAL 1/2 (HAL 3/4).

2.5.1. Pinning of the Interface Connector

The pinning of the interface connector is described in Table 2–2.

Table 2–2: Pinning HAL Interface

Pin No.	Description
1	Sensor input HAL 1/2: V_{DD} Sensor 1 HAL 3/4: V_{DD} Sensor 3
2	Sensor input HAL 1/2: V_{DD} Sensor 2 HAL 3/4: V_{DD} Sensor 4
3, 4	Common sensor ground
5	Sensor output HAL 1/2: V_{OUT} Sensor 1 HAL 3/4: V_{OUT} Sensor 3
6	Sensor output HAL 1/2: V_{OUT} Sensor 2 HAL 3/4: V_{OUT} Sensor 4

2.6. LED Description

Table 2–3: LED description

LED	Description	Remarks
NORM	Normal operating voltage (V_{DD})	5.0 V calibrated
LOW	Low level of telegram (V_{DD})	5.5 V calibrated
HIGH	High level of telegram (V_{DD})	8.0 V calibrated
PROG	Programming voltage level	12.5 V calibrated
READY	On when board ready for operation	
DATA	High level of telegram (V_{OUT})	
PCCOM	Receive data from PC	
ERROR	Error flag	Status = 1

2.7. Error Flag Codes

Each response sent from the programmer board to the PC contains a status or error flag.

Table 2–4: Error flag description

Error Flag	Description
0	No error; status OK
1	Unspecified system error
2	Output low-level detection failure
3	Missing acknowledge (ACK)
4	ACK time-out failure
5	Bit time < 1 ms
6	VPROG out of range

Note: For safe programming, the evaluation of the error flags is mandatory.

3. Usage of the Hall Programmer V 5.1

3.1. Tips for Safe Programming of the Sensors

The Hall programmer board V 5.1 generates all voltage levels according to the specification of the serial protocol.

V_{PROG} external generation

In case of the sensors HAL805, HAL815, HAL817, HAL810, HAL824, HAL825, HAL830, HAL835, HAL880, HAL1000 and HAL1002 the programming voltage V_{PROG} (which is necessary for the permanent storing of the data in the EEPROM memory) has to be generated externally, i.e. by the programmer board. If V_{PROG} is out of the specification limits, the programming of the EEPROM memory may be insufficient and the reliability of the sensor may be reduced. Hence, the Hall programmer board V 5.1 is equipped with an on-board A/D-converter which reads back the programming voltage V_{PROG}. The A/D-converter is polled every 80 µs.

If V_{PROG} drops below 12.4 V, an error flag is set and the ERROR LED on the programmer board ignites. Please ensure that your software evaluates the error flags (see Section 2.7).

V_{PROG} internal generation

In case of the sensors HAL855 and HAL856, the programming voltage is generated internally. The supply voltage of the Hall sensor during programming must be 5.0 V.

The width of both pulses (erase and prom) is defined by the programmer board. In case a customer-specific software is implemented, please ensure that the programming time is set to 100 ms.

We recommend supervising the programming process and verification of the results after programming the sensors (see Table 3–1).

Note: The programming voltage V_{PROG} is measured at the output of the Hall programmer board. However, this measurement cannot consider any voltage drop which may occur between the Hall programmer board and the Hall sensor.
The voltage drop between the programmer board and the Hall sensor must not exceed 50 mV.
Please ensure that the Hall sensor and the programmer board are connected low resistively to a common ground.

Note: Electrostatic discharges (ESD) may disturb the programming pulses. Please take precautions against ESD.

Table 3–1: Check list

Check Item	When
Status/error flag	After transmission of each write, read, or store command
Programming voltage	After sending a store command (ERASE and PROM)
Programming pulse width	Evaluation of software after implementation
Verify written data	After finishing the programming sequence
Lock function	After lock (Note: The Lock function is active after the next power-on reset.)
Analog test (functional test)	After locking

3.2. Recommended Operating Conditions

Functional operation at conditions other than those listed below is not implied and may result in insufficient programming of the connected sensor and may cause damage to the Hall programmer board.

Symbol	Parameter	Min.	Typ.	Max.	Unit	Conditions
V_{SUP}	Supply Voltage (DC)	17.0	18.0	20.0	V	
V_{ACp-p}	Supply Voltage AC Ripple	–	0	0.150	V_{p-p}	
I_{SUP}	Supply Current	50	120	300	mA	
C_L	Load Capacitance	–	–	2000[1)]	nF	
I_{OUT}	Continuous Output Current	–	–	60	mA	
T_A	Ambient Temperature Range	10	25	50	°C	

[1)] Example: If four sensors are connected to the programmer board, blocking capacitors of 500 nF between V_{DD} and GND of each sensor can be applied.

3.3. 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 3–2.

Note: The programmer board reads back the programming voltage every 80 µs in order to be able to detect short spikes on the output voltage line. If cables other than those specified in Table 3–2 are used, such spikes may not be detected by the board.

3.4. 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 5.1, nor readjust any trimming potentiometer. Otherwise, the board may be damaged, the sensor programming may be insufficient, and the reliability of the sensor reduced.

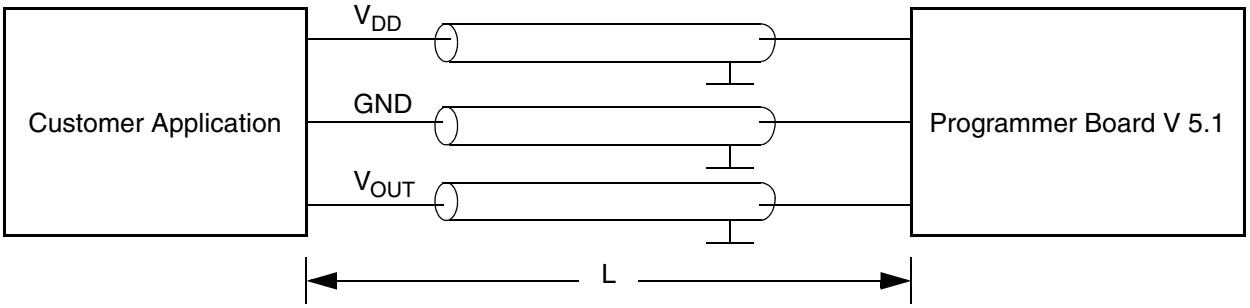

Fig. 3–1: Recommended wiring – schematic sketch

Table 3–2: 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	

3.4.1. Firmware Update

The HAL Programmer V5.1 is equipped with a flash-memory. Thus, firmware-upgrades can be done easily by flashing the new firmware into this memory. For this purpose the software-tool-kit ‘FLASH IT’ is provided.

1. Unzip the package ‘Flashit.zip’ into a suitable working directory. When unzipping with ‘WinZip’ a sub-folder ‘\Flashit’ will be generated.
2. Close the jumper ‘BOOT’ beneath the ‘Reset’-button of the HAL Programmer Board.
3. Push the ‘Reset’-button.
4. Open the jumper ‘BOOT’ again.
5. Start the executable ‘Bootload.exe’ by double-clicking the icon in \Flashit.
6. After hitting the icon, the flash process starts.

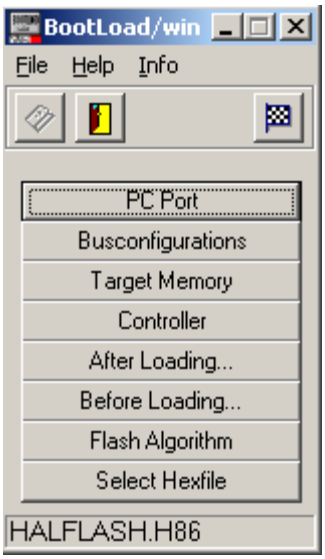

Fig. 3–2: Flash procedure

Note: The flash process lasts for about 10 – 15 min. After the process has finished, the board will be reset automatically and the new firmware will start working.

Note: The firmware to be loaded into the flash-memory is called ‘HALFLASH.H86’. It must be stored in the same folder where the executable is located.

4. Operation Mode 0

Emulation of the Programmer Board V 4.1: HAL805, HAL815, HAL817, HAL810, and HAL1000

This operation mode is the default mode. It is invoked after power-on or pushing the reset button. Alternatively, this operation mode can be set by sending the “j0” command.

Communication: PC → prog. board

Section 4.1 and Section 4.2 contain a detailed description of communication between the PC and the programmer board V 5.1. In particular, all board commands available in the operation mode “j0” are listed.

Communication: prog. board → sensor

The communication between the programmer board and the Hall sensor is described in Section 4.3

Data formats

The data formats for communication between both the PC and the board, and the board and the sensor are summarized in Section 4.4

Programming procedure

Finally, Section 4.5 depicts the programming procedure and gives an example of a complete programming sequence.

4.1. Definition of the Protocol

The protocol and the commands in the emulation mode are exactly the same as those of the programmer board V 4.1.

The general syntax is as follows (blanks are given for better readability only):

PC → Board:
STX BCMD [CMD CP ADR AP] [DAT3 DAT2 DAT1 DAT0 DP] ETX

Board → PC:
STX STATUS DAT3 DAT2 DAT1 DAT0 DP ETX

The characters in brackets [...] are optional, depending on the board command BCOM.

Table 4–1: Description of the characters

Character	Description
STX	ASCII character 2
ETX	ASCII character 3
BCMD	Board command
CMD	Command
CP	Command parity
ADR	Address
AP	Address parity
DAT3 DAT2 DAT1 DAT0	Data characters, each encoding 4 bit in HEX format. Example: 0 1 f 0 (HEX) = 496
DP	Data parity
STATUS	Status/error flag

4.2. The Board Commands BCMD

The board commands activate functions of the firmware. The characters following the BCMD are the parameters of this function. Some functions serve to control the board operation and do not address the Hall device.

The read, write, program, and lock functions communicate with the Hall device. The parameters of these functions are converted into a serial telegram and is sent to the connected sensor.

Table 4–2: Description of the board commands

BCMD	Parameters	Description
n	–	Switch V_{DD} on
o	–	Switch V_{DD} off
z[t]	t = 10...255 (as ASCII character) default t = 162 HAL8x5, HAL810, HAL1000: set t = 85	Set bit time slow mode bit time = t × 0.02 ms default bit time = 3.24 ms HAL8x5, HAL810, HAL1000: set bit time = 1.7 ms
u[t]	t = 10...255 (as ASCII character) HAL8x5, HAL810, HAL1000: set t = 200	Set programming pulse width width = t × 0.50 ms HAL8x5, HAL810, HAL1000: set width = 100 ms
a[t]	t = 10...255 (as ASCII character) recommended t = 25	Select sensor A, select pulse width = t × 4.0 µs recommended pulse width = 100 µs
b[t]	t = 10...255 (as ASCII character) recommended t = 25	Select sensor B, select pulse width = t × 4.0 µs recommended pulse width = 100 µs
e[p]	p = CMD CP ADR AP DAT3 DAT2 DAT1 DAT0 DP	Write HAL and echo data to PC
q[p]	p = CMD CP ADR AP	Read HAL and echo data to PC
m[p]	p = CMD CP ADR AP	Program HAL and echo program voltage to PC
l[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2	Lock HAL and echo program voltage to PC
v	–	Request firmware version
t	–	Request status and echo data to PC
j[p]	p = 0...3 (as HEX)	Switch board operation mode 0 = EMU Board V 4.1 (default) 1 = HAL805, HAL815, HAL817, HAL810, HAL82x, HAL83x, HAL880, and HAL100x 2 = HAL855 3 = HAL856

Examples:

(Blanks are given for better readability only)

– power on:
STX n ETX

– set bit time slow CHR$(85) = 'U':
STX z U ETX

– read register no. 2:
STX q 2 0 2 1 ETX

– write register no. 2:
STX e 3 1 2 1 0 0 0 A 1 ETX

– store sequence:
STX u CHR$(200) ETX (set programming time)
STX m 5 1 1 1 ETX (ERASE)
STX m 4 0 1 1 ETX (PROM)

– deactivate sensors:
STX e 3 1 F 0 0 8 0 F 0 ETX

– select sensor A:
STX a CHR$(25) ETX

– lock sensors
STX u CHR$(200) ETX (set programming time)
STX l 7 0 6 0 5 1 1 1 ETX (LOCK-ERASE)

4.2.1. Monitoring of the Programming Voltage

In case of ERASE, PROM, and LOCK, the programming pulse voltage is measured by the board, and a data string is sent back to the PC. The relation between the data and the programming voltage is

$$V_{PROG} = \frac{DAT}{4095} \cdot 6 \cdot 2.485\ \mathrm{V}$$

If the programming pulse voltage is out of the specification limits, the status bit is set to "1".

Examples:

– V_{PROG} = 12.15 V, Data board → PC:
STX 1 0 D 0 A 0 ETX
(DAT = 0 D 0 A)

– V_{PROG} = 12.50 V, Data board → PC:
STX 0 0 D 6 9 0 ETX
(DAT = 0 D 6 9)

4.3. Programming of the Sensor

4.3.1. Definition of Programming Pulses

The sensor is addressed by modulating a serial telegram on the supply voltage. The sensor answers with a serial telegram on the output pin.

The bits in the serial telegram have a different bit time for the V_{DD}-line and the output. The bit time for the V_{DD}-line is defined through the length of the Sync bit at the beginning of each telegram. The bit time for the output is defined through the Acknowledge bit.

A logical “0” is coded as no voltage change within the bit time. A logical “1” is coded as a voltage change between 50% and 80% of the bit time. After each bit, a voltage change occurs.

4.3.2. Definition of the Telegram

Each telegram starts with the Sync bit (logical 0), 3 bits for the Command (COM), the Command Parity bit (CP), 4 bits for the Address (ADR), and the Address Parity bit (AP).

There are 4 kinds of telegrams:

Write a register

After the AP bit, follow 14 Data bits (DAT) and the Data Parity bit (DP). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge bit (logical 0) on the output (see Fig. 4–2).

Read a register

After evaluating this command, the sensor answers with the Acknowledge bit, 14 Data bits, and the Data Parity bit on the output (see Fig. 4–3).

Programming the EEPROM cells

After evaluating this command, the sensor answers with the Acknowledge bit. After the delay time t_w, the supply voltage rises up to the programming voltage (see Fig. 4–4).

Activate a sensor

If more than one sensor is connected to the supply line, selection can be done by first deactivating all sensors. The output of all sensors will be pulled to ground by the internal 10 kΩ resistors. With an activate pulse on the appropriate output pin, an individual sensor can be selected. All following commands will only be accepted from the activated sensor (see Fig. 4–5).

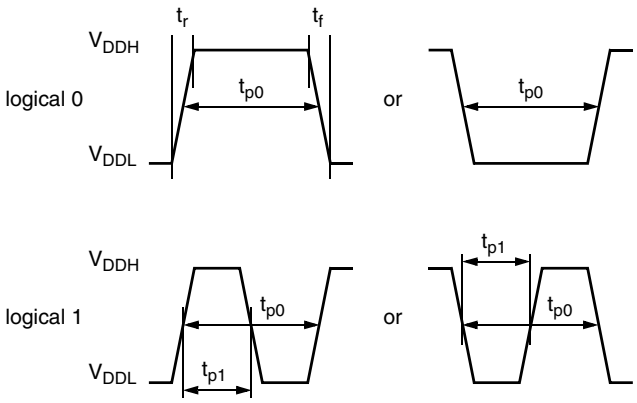

Fig. 4–1: Definition of logical 0 and 1 bit

Table 4–3: Telegram parameters

Symbol	Parameter	Pin	Min.	Typ.	Max.	Unit	Remarks
V_{DDL}	Supply voltage for low level during programming	1	5	5.6	6	V	
V_{DDH}	Supply voltage for high level during programming	1	6.8	8.0	8.5	V	
t_r	Rise time	1			0.05	ms	
t_f	Fall time	1			0.05	ms	
t_{p0}	Bit time on V_{DD}	1	1.7	1.75	1.8	ms	t_{p0} is defined through the Sync bit
t_{pOUT}	Bit time on output pin	3	2	3	4	ms	t_{pOUT} is defined through the Acknowledge bit
t_{p1}	Voltage change for logical 1	1, 3	50	65	80	%	% of t_{p0} or t_{pOUT}
V_{DDPROG}	Supply voltage for programming the EEPROM	1	12.4	12.5	12.6	V	
t_{PROG}	Programming time for EEPROM	1	95	100	105	ms	
t_{rp}	Rise time of programming voltage	1	0.2	0.5	1	ms	
t_{fp}	Fall time of programming voltage	1	0		1	ms	
t_w	Delay time of programming voltage after Acknowledge	1	0.5	0.7	1	ms	
V_{act}	Voltage for an Activate pulse	3	3	4	5	V	
t_{act}	Duration of an Activate pulse	3	0.05	0.1	0.2	ms	

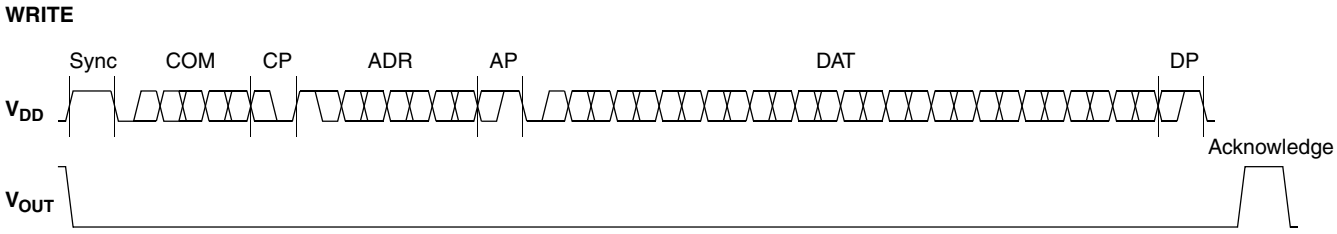

Fig. 4–2: Telegram for coding a Write command

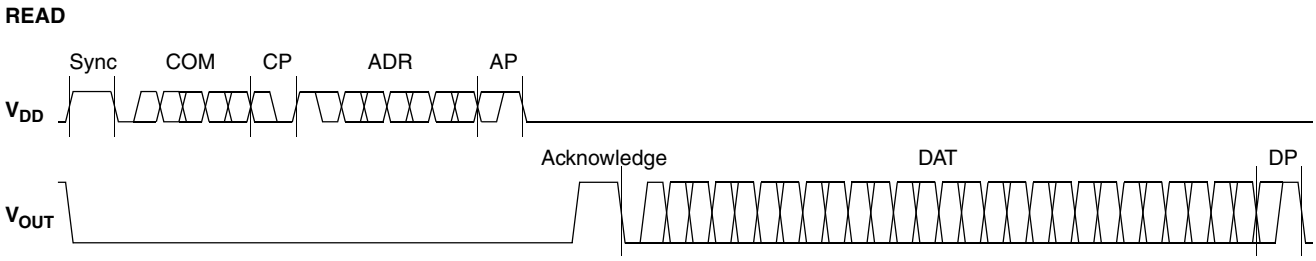

Fig. 4–3: Telegram for coding a Read command

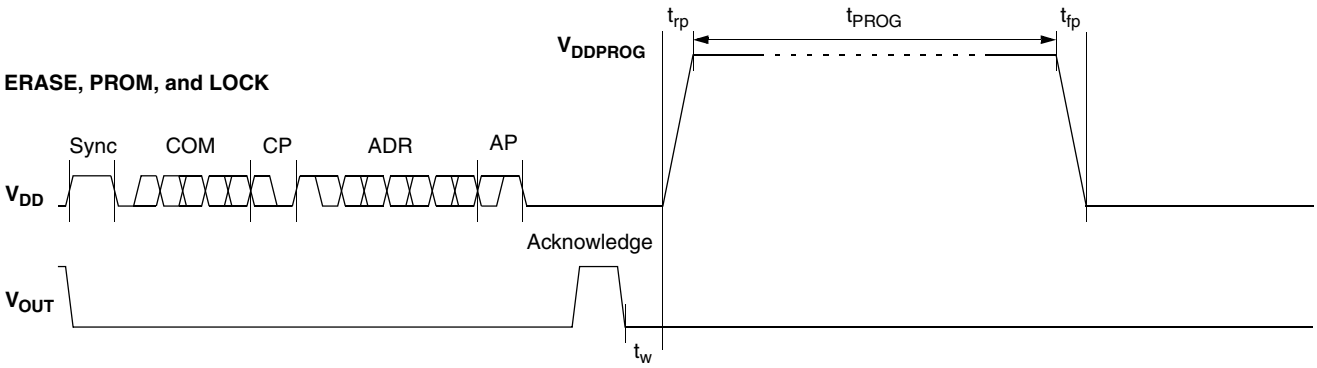

Fig. 4–4: Telegram for coding the EEPROM programming

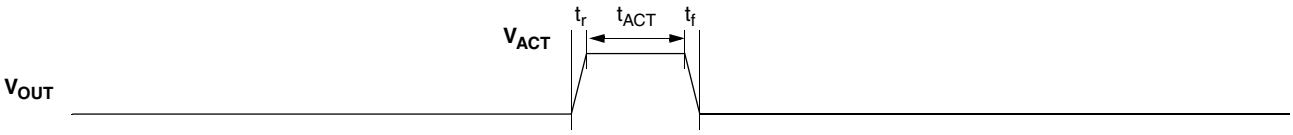

Fig. 4–5: Activate pulse

4.3.3. Telegram Codes

Sync Bit

Each telegram starts with the Sync bit. This logical “0” pulse defines the exact timing for t_{p0}.

Command Bits (COM)

The Command code contains 3 bits and is a binary number. Table 4–4 shows the available commands and the corresponding codes for the HAL805, HAL815, HAL817 and HAL1000.

Command Parity Bit (CP)

This Parity bit is “1” if the number of zeros within the 3 Command bits is uneven. The Parity bit is “0”, if the number of zeros is even.

Address Bits (ADR)

The Address code contains 4 bits and is a binary number. Table 4–5 shows the available addresses for the HAL805, HAL815, HAL817 and HAL1000 registers.

Address Parity Bit (AP)

This Parity bit is “1” if the number of zeros within the 4 Address bits is uneven. The Parity bit is “0” if the number of zeros is even.

Data Bits (DAT)

The 14 Data bits contain the register information.

The registers use a different number formats for the Data bits. These formats are explained in Section 4.3.4

In the Write command, the last bits are valid. If, for example, the TC register (6 bits) is written, only the last 6 bits are valid.

In the Read command, the first bits are valid. If, for example, the TC register (6 bits) is read, only the first 6 bits are valid.

Data Parity Bit (DP)

This parity bit is “1” if the number of zeros within the binary number is even. The parity bit is “0” if the number of zeros is uneven.

Acknowledge

After each telegram, the output answers with the Acknowledge signal. This logical “0” pulse defines the exact timing for t_{pOUT}.

Table 4–4: Available commands

Command	Code	Explanation
READ	2	read a register
WRITE	3	write a register
PROM	4	program all nonvolatile registers (except the lock bits)
ERASE	5	erase all nonvolatile registers (except the lock bits)
LOCK	7	lock the whole device and switch permanently to the analog-mode

4.3.4. Number Formats

Binary number: The most significant bit is given as first, the least significant bit as last digit.

Example:

101001 represents 41 decimal.

Signed binary number: The first digit represents the sign of the following binary number (1 for negative, 0 for positive sign).

Example:

0101001 represents +41 decimal
1101001 represents –41 decimal

Two's-complementary number: The first digit of positive numbers is "0", the rest of the number is a binary number. Negative numbers start with "1". In order to calculate the absolute value of the number, calculate the complement of the remaining digits and add "1".

Example:

0101001 represents +41 decimal
1010111 represents –41 decimal

Table 4–5: Available register addresses

Register	Code	Data Bits	Format	Customer	Remark
CLAMP LOW	1	10	binary	read/write/program	low clamping voltage
CLAMP HIGH	2	11	binary	read/write/program	high clamping voltage
VOQ	3	11	two compl. binary	read/write/program	
SENSITIVITY	4	14	signed binary	read/write/program	
MODE	5	6	binary	read/write/program	range and filter settings
LOCK	6	1	binary	lock	lock bit
ADC-READOUT	7	14	two compl. binary	read	
TC	11	6	signed binary	read/write/program	
TCSQ	12	5	binary	read/write/program	
DEACTIVATE	15	12	binary	write	deactivate the sensor

4.3.5. Register Information (HAL805/HAL815/HAL817)

Terminology:

REGISTER: Name of the register or register value

Register: Name of the parameter

CLAMP LOW

– The register range is from 0 up to 1023.
– The register value is calculated by:

$$CLAMP\ LOW = \frac{Low\ Clamping\ Voltage}{V_{DD}} \cdot 2048$$

CLAMP HIGH

– The register range is from 0 up to 2047.
– The register value is calculated by:

$$CLAMP\ HIGH = \frac{High\ Clamping\ Voltage}{V_{DD}} \cdot 2048$$

VOQ

– The register range is from –1024 up to 1023.
– The register value is calculated by:

$$VOQ = \frac{V_{OQ}}{V_{DD}} \cdot 1024$$

SENSITIVITY

– The register range is from –8192 up to 8191.
– The register value is calculated by:

$$SENSITIVITY = Sensitivity \cdot 2048$$

TC and TCSQ

– The TC register range is from –31 up to 31.
– The TCSQ register range is from 0 up to 31.

Please refer to the data sheet for the recommended values.

MODE

– The register range is from 0 up to 63 and contains the settings for FILTER and RANGE:

$$MODE = FILTER \cdot 8 + RANGE$$

Please refer to the data sheet for the available FILTER and RANGE values.

ADC-READOUT

– This register is read only.

– The register range is from –8192 up to 8191.

DEACTIVATE

– This register can only be written.

– The register has to be written with 2063 decimal (80F hexadecimal) for the deactivation.

– The sensor can be reset with an Activate pulse on the output pin or by switching off and on the supply voltage.

Note: The register information of the HAL810 is slightly different. However, it is quite easy to transform voltage values (HAL8x5/HAL817) into output duty cycle (HAL810) by multiplication with

$$\frac{100\%}{5V}$$

For detailed register information of the HAL810, please refer to the data sheet.

4.3.6. Programming Information

If the content of any register (except the lock registers) is to be changed, the desired value must first be written into the corresponding RAM register. Before reading out the RAM register again, the register value must be permanently stored in the EEPROM.

Permanently storing a value in the EEPROM is done by first sending an ERASE command followed by sending a PROM command. The address within the ERASE and PROM commands is not important. ERASE and PROM act on all registers in parallel.

If all HAL805, HAL815, HAL817 and HAL1000 registers are to be changed, all writing commands can be sent one after the other, followed by sending one ERASE and PROM command at the end.

During all communication sequences, the customer has to check if the communication with the sensor was successful. This means that the acknowledge and the parity bits sent by the sensor have to be checked by the customer. If the Micronas programmer board is used, the customer has to check the error flags sent from the programmer board.

Note: For production and qualification tests, it is mandatory to set the Lock bit after final adjustment and programming. The Lock function is active after the next power-up of the sensor. Micronas also recommends sending an additional ERASE command after sending the LOCK command (This is done automatically when the board command “l” is used for locking the sensor.).
The success of the Lock process should be checked by reading at least one sensor register after locking and/or by an analog check of the sensor’s output signal.
Electrostatic Discharges (ESD) may disturb the programming pulses. Please take precautions against ESD.

4.4. Data Formats

The protocol between the programmer board and the Hall sensor defines fixed lengths for the data, command, and address words to be transmitted:

– Command (CMD): 3 bit

– Address (ADR): 4 bit

– Data (DAT): 14 bit

– Command Parity (CP): 1 bit

– Address Parity (AP): 1 bit

– Data Parity (DP): 1 bit

Reminder: For communication with the programmer board, the 14 data bits are encoded in four ASCII characters: DAT3, DAT2, DAT1, and DAT0.

As every register has a different length, not all of the 14 bits are used. Furthermore, there is a difference between the write format (transfer board → IC) and the read format (transfer IC → board). Table 4–6 shows which of the 14 bits are valid for each register (write format and read format).

Table 4–6: Data formats

Register	Char	DAT3				DAT2				DAT1				DAT0			
	Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CLAMP LOW	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–
CLAMP HIGH	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
VOQ	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SENSITIVITY	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
MODE	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
ADC-READOUT	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
TC	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
TCSQ	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
DEACTIVATE	Write	–	–	–	–	1	0	0	0	0	0	0	0	1	1	1	1
V: valid, –: ignore, bit order: MSB first																	

4.5. The Programming Procedure

The general programming procedure is described in detail for the HAL805 (HAL815/ HAL817).

The procedure for the HAL810 is quite similar. However, the multiprogramming (i.e. the programming of more than one sensor connected to the same supply voltage) does not apply for the HAL810. Additionally, some register names are different for the HAL810.

Table 4–7: Cross reference register names

Register HAL8x5	Register HAL810
CLAMP LOW	MIN DUTY
CLAMP HIGH	MAX DUTY
SENSITIVITY	DC SENSITIVITY
VOQ	DCOQ

An example of a programming procedure is given in Fig. 4–6.

First, the programmer board has to be initialized. If more than one sensor is connected to the same supply line, the sensors have to be addressed sequentially. This is done by the multiprogramming loop: After deactivating all sensors, the sensor to be programmed is activated again by sending a pulse on the corresponding output line, followed by a dummy read command.

Note: The flow chart shown in Fig. 4–6 is intended as a simple example. The multiprogramming loop can also be done within the calibration procedure.

Please keep in mind that the board commands “e...”, “q...”, “m...”, and “t” cause the board to send back an eight character string to the PC. This string must be read out of the serial port before sending the next command.

Example:

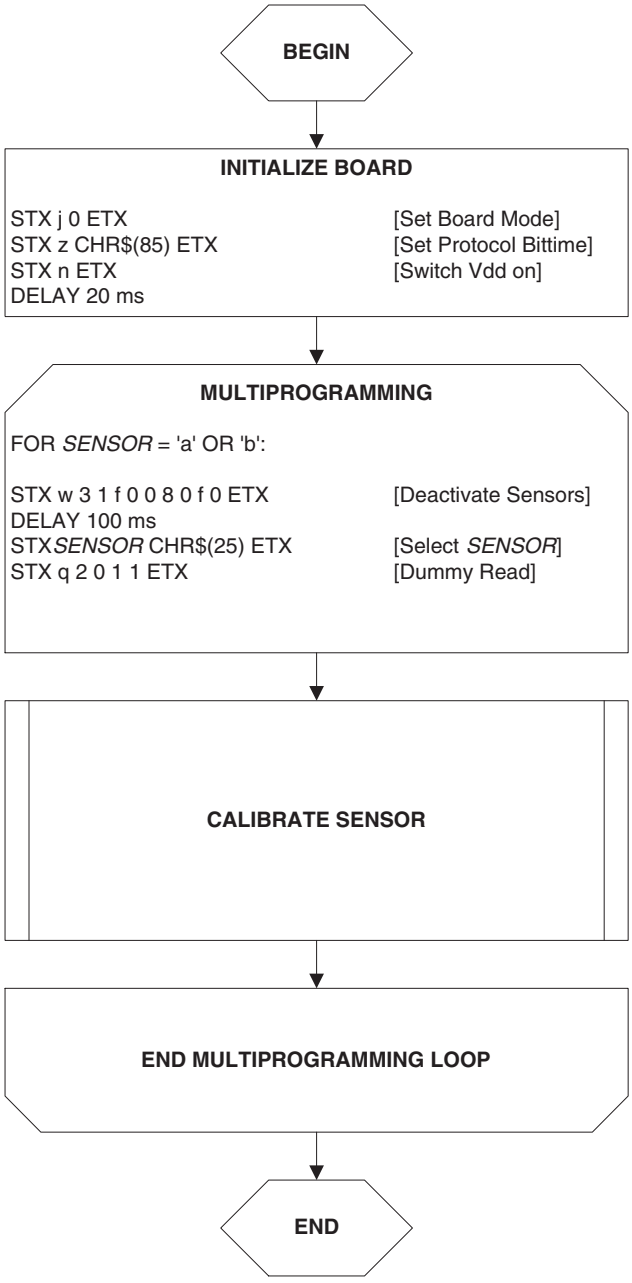

Fig. 4–6: General procedure

4.5.1. The Calibration Procedure

The output characteristic is determined by four parameters:

– Clamp Low specifies the lower clamping voltage of the output curve.

– Clamp High specifies the higher clamping voltage of the output curve.

– V_{OQ} corresponds to the output voltage at zero ADC-Readout.

– Sensitivity corresponds to the increase of the output voltage with magnetic field. A sensitivity of “1” is equivalent to a voltage increase of 5 V at an increase of 2048 of the ADC-Readout.

The following simple formula relates the analog output voltage to the ADC-Readout (V_{DD} is considered to be 5.0 V):

$$V_{OUT} = \frac{Sensitivity \cdot \text{ADC-Readout} \cdot 5\ \text{V}}{2048} + V_{OQ}$$

The calibration procedure is shown in Fig. 4–7. The register values are intended as example:

Step 1: Programming of the parameters which do not require individual adjustment:

– Clamp Low, Clamp High, Filter Frequency, Magnetic Range, TC, and TCSQ.

Step 2: Get the ADC-Readout value in the first calibration point and assign a nominal output voltage ($V_{OUT,1}$) to that point.

Step 3: Get the ADC-Readout value in the second calibration point and assign a nominal output voltage ($V_{OUT,2}$) to that point.

Step 4: Calculate and program the values of SENSITIVITY and VOQ.

$$Sensitivity = \frac{(V_{OUT2} - V_{OUT1})}{(ADCReadout_2 - ADCReadout_1)} \cdot \frac{2048}{5V}$$

$$SENSITIVITY = Sensitivity \cdot 2048$$

$$V_{OQ} = V_{OUT1} - ADCReadout_1 \cdot Sensitivity \cdot \frac{5V}{2048}$$

$$VOQ = V_{OQ} \cdot \frac{1024}{5V}$$

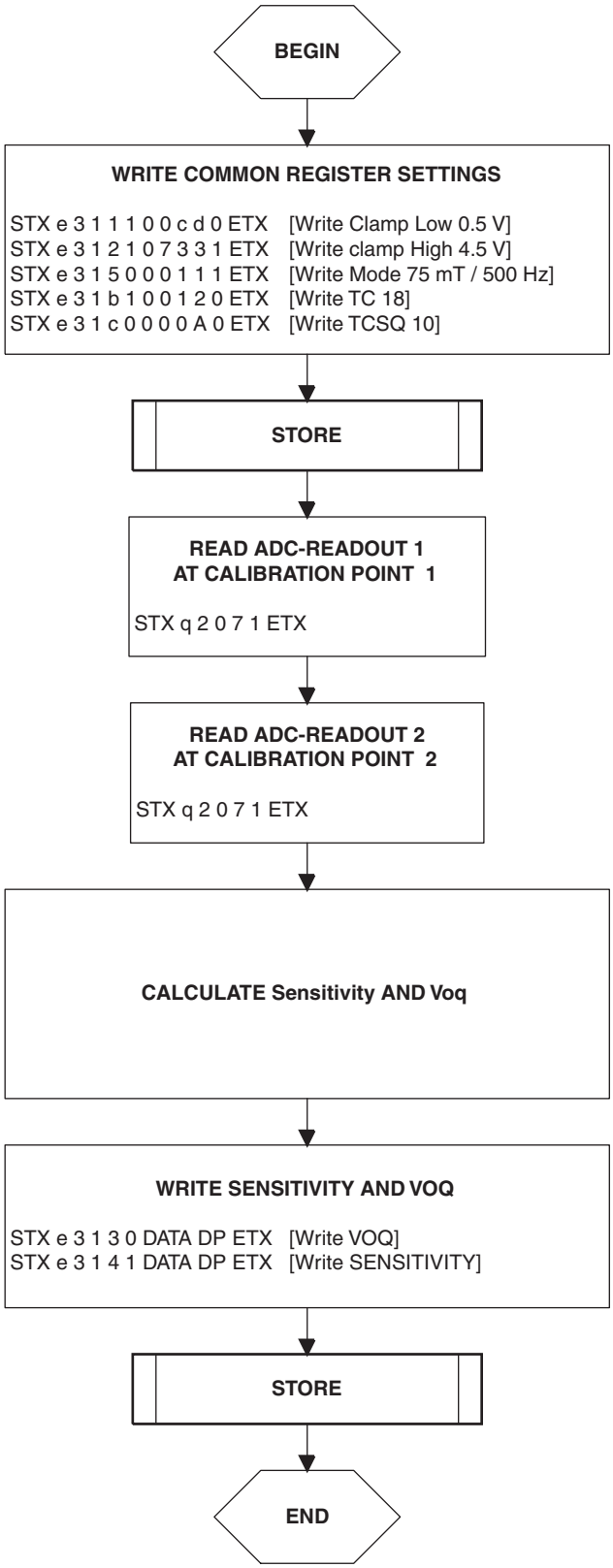

Fig. 4–7: Calibration procedure

Step 5:

Store the registers permanently.

The “Store” sequence is given in Fig. 4–8. The permanent storing of data into the EEPROM memory is a two-step process. First, all zero bits (ERASE), then all one bits (PROM) are written.

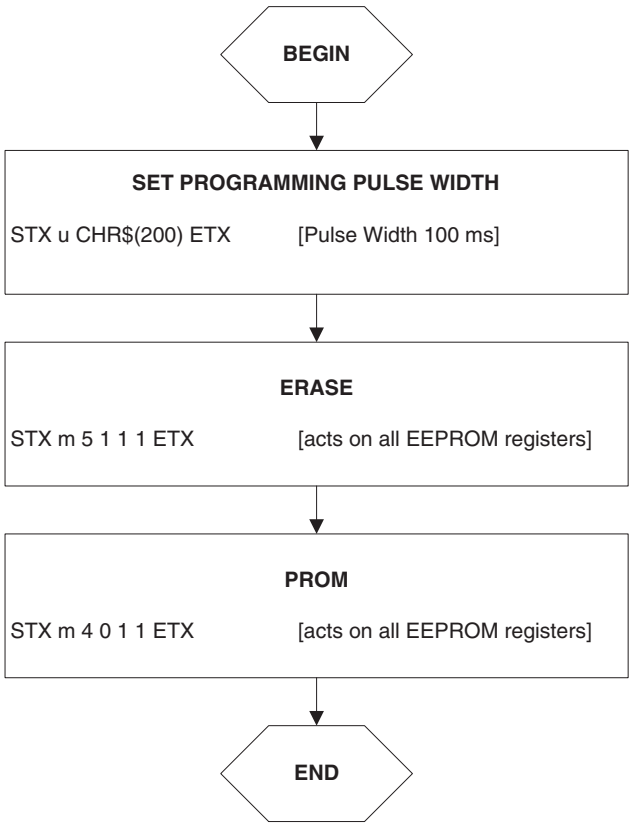

Fig. 4–8: Store sequence

5. Operation Mode 1

HAL805, HAL815, HAL817, HAL810, HAL82x, HAL83x, HAL880, HAL1000 and HAL1002

This operation mode is set by sending the “j1” command. Unlike the emulation mode described in the previous section, this operation mode allows the use of additional features which are not implemented in the Hall Programmer Board V 4.1.

Communication: PC → prog. board

Section 5.1 and Section 5.2 contain a detailed description of communication between the PC and the programmer board V 5.1. Particularly, all board commands available in the operation mode “j1” are listed.

Communication: prog. board → sensor

The communication between the programmer board and the Hall sensor is described in Section 5.3

Data formats

The data formats for communication between both the PC and the board, and the board and the sensor are summarized in Section 5.4

Programming procedure

Finally, Section 5.5 depicts the programming procedure and gives an example of a complete programming sequence.

5.1. Definition of the Protocol

The general syntax is as follows (blanks are given for better readability only):

PC → Board:
STX BCMD [CMD CP ADR AP] [DAT3 DAT2 DAT1 DAT0 DP] ETX

Board → PC:
STX STATUS DAT3 DAT2 DAT1 DAT0 DP ETX

The characters in brackets [...] are optional, depending on the board command BCOM.

Table 5–1: Description of the characters

Character	Description
STX	ASCII character 2
ETX	ASCII character 3
BCMD	Board command
CMD	Command
CP	Command parity
ADR	Address
AP	Address parity
DAT3 DAT2 DAT1 DAT0	Data characters, each encoding 4 bit in HEX format. Example: 0 1 f 0 (HEX) = 496
DP	Data parity
STATUS	Status/error flag

5.2. The Board Commands BCMD

The board commands activate functions of the firmware. The characters following the BCMD are the parameters of this function. Some functions serve to control the board operation and do not address the Hall device.

The read, write, program, and lock functions communicate with the Hall device. The parameters of these functions are converted into a serial telegram and sent to the connected sensor.

Table 5–2: Description of the board commands

BCMD	Parameters	Description
n	–	Switch V_{DD} on
o	–	Switch V_{DD} off
z[t]	t = 10...255 (as ASCII character) default: t = 85	Set bit time slow mode bit time = t × 0.02 ms default: bit time = 1.7 ms
u[t]	t = 10...255 (as ASCII character) set t = 100	Set programming pulse width width = t × 1.00 ms set pulse width = 100 ms
h[nt]	n = 1... 4 (as HEX) t = 10...255 (as ASCII character) recommended t = 25	Select sensor n, (default = 1) select pulse width = t × 4.0 µs recommended pulse width = 100 µs
e[p]	p = CMD CP ADR AP DAT3 DAT2 DAT1 DAT0 DP	Write HAL and echo data to PC
q[p]	p = CMD CP ADR AP	Read HAL and echo data to PC
m[p]	p = CMD CP ADR AP	Program HAL and echo program voltage to PC
l[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2	Lock HAL and echo program voltage to PC This command is not valid for HAL 82x, HAL 83x, and HAL880 and HAL 1002!
v	–	Request firmware version Before reading back the firmware version this command needs to be followed by the “t“ command.
t	–	Request status and echo data to PC
j[p]	p = 0...3 (as HEX)	Switch board operation mode 0 = EMU board V 4.1 (default) 1 = HAL805, HAL815, HAL817, HAL810, HAL82x, HAL83x, HAL880, HAL1000, HAL1002 2 = HAL855 3 = HAL856

Examples:

Note: Blanks are given for better readability only

5.2.0.1. HAL805, HAL815, HAL817, HAL810 and HAL1000

```
STX j 1 ETX                    // set board operation mode
STX n ETX                      // power on
STX z U ETX                    // set bit time slow CHR$(85) = 'U'
STX q 2 0 2 1 ETX              // read register no. 2
STX e 3 1 2 1 0 0 0 A 1 ETX    // write register no. 2
// store sequence start ----------------------------------------------------------
STX u CHR$(100) ETX            // set programming time
STX m 5 1 1 1 ETX              // ERASE
STX m 4 0 1 1 ETX              // PROM
// store sequence end ------------------------------------------------------------
STX e 3 1 F 0 0 8 0 F 0 ETX    // deactivate sensors
STX h 1 CHR$(25) ETX           // select sensor 1
// lock sequence start -----------------------------------------------------------
STX u CHR$(100) ETX            // set programming time
STX l 7 0 6 0 5 1 1 1 ETX      // LOCK-ERASE
// lock sequence end -------------------------------------------------------------
```

5.2.0.2. HAL82x, HAL83x, HAL880, HAL1002

STX j1 ETX// set board operation mode

STX n ETX// power on

STX z U ETX // set bit time slow CHR$(85) = 'U'

STX q 2 0 2 1 ETX// read register no. 2

STX e 3 1 2 1 0 0 0 A 1 ETX// write register no. 2

// store sequence start ----------

STX u CHR$(100) ETX // set programming time

STX m 5 1 0 0 ETX // ERASE

STX m 4 0 0 0 ETX // PROM

// store sequence end ----------

STX e 3 1 F 0 0 8 0 F 0 ETX// deactivate sensors

STX h 1 CHR$(25) ETX// select sensor 1

STX q 2 0 2 1 ETX// dummy read

// lock sequence start ----------

STX u CHR$(100) ETX// set programming time

STX e 3 1 6 0 0 0 0 1 0 ETX // set LOCK bit

STX m 5 1 0 0 ETX // ERASE

STX m 4 0 0 0 ETX // PROM

5.2.1. Monitoring of the Programming Voltage

In case of ERASE, PROM, and LOCK, the programming pulse voltage is measured by the board, and a data string is sent back to the PC. The relation between the data and the programming voltage is

$$V_{PROG} = \frac{DAT}{4095} \cdot 6 \cdot 2.485\ \mathrm{V}$$

If the programming pulse voltage is out of the specification limits, the status bit is set to "1".

Examples:

- V_{PROG} = 12.15 V, Data Board → PC:
  STX 1 0 D 0 A 0 ETX
  (DAT = 0 D 0 A)
- V_{PROG} = 12.50 V, Data Board → PC:
  STX 0 0 D 6 9 0 ETX
  (DAT = 0 D 6 9)

5.3. Programming of the Sensor

5.3.1. Definition of Programming Pulses

The sensor is addressed by modulating a serial telegram on the supply voltage. The sensor answers with a serial telegram on the output pin.

The bits in the serial telegram have a different bit time for the V_{DD}-line and the output. The bit time for the V_{DD}-line is defined through the length of the Sync bit at the beginning of each telegram. The bit time for the output is defined through the Acknowledge bit.

A logical “0” is coded as no voltage change within the bit time. A logical “1” is coded as a voltage change between 50% and 80% of the bit time. After each bit, a voltage change occurs.

5.3.2. Definition of the Telegram

Each telegram starts with the Sync bit (logical 0), 3 bits for the Command (COM), the Command Parity bit (CP), 4 bits for the Address (ADR), and the Address Parity bit (AP).

There are 4 kinds of telegrams:

Write a register

After the AP bit, follow 14 Data bits (DAT) and the Data Parity bit (DP). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge bit (logical 0) on the output (see Fig. 5–2).

Read a register

After evaluating this command, the sensor answers with the Acknowledge bit, 14 Data bits, and the Data Parity bit on the output (see Fig. 5–3).

Programming the EEPROM cells

After evaluating this command, the sensor answers with the Acknowledge bit. After the delay time t_w, the supply voltage rises up to the programming voltage (see Fig. 5–4).

Activate a sensor

If more than one sensor is connected to the supply line, selection can be done by first deactivating all sensors. The output of all sensors will be pulled to ground. With an Activate pulse on the appropriate output pin, an individual sensor can be selected. All following commands will only be accepted from the activated sensor (see Fig. 5–5).

Note: In case of HAL 82x, HAL83x, HAL 880 and HAL1002 an additional 10 kΩ pull-down resistor is required during programming. Firmware rev. 1.33 can be used alternatively.

Lock a sensor

To lock the EEPROM registers, the lock bit has to be set. In case of HAL 82x, HAL83x, HAL880 and HAL1002 it is necessary to write the lock bit into the lock register (see Fig. 5–2). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge Bit (logical 0) on the output. In order to store the lock bit permanently, an erase and program command have to be sent to the sensor. The is the same procedure as mentioned above (see Programming the EEPROM cells and Fig. 5–4). The EEPROM registers are locked **after** a power on reset. For HAL 805, HAL 815, HAL817, HAL 810 and HAL1000 the lock command must be used.

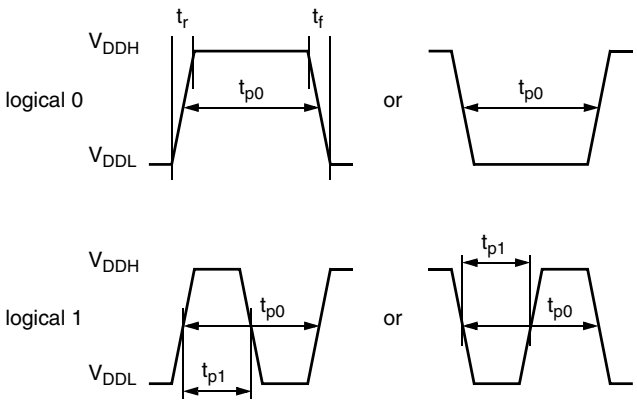

Fig. 5–1: Definition of logical 0 and 1 bit

Table 5–3: Telegram parameters

Symbol	Parameter	Pin	Min.	Typ.	Max.	Unit	Remarks
V_{DDL}	Supply voltage for low level during programming	1	5	5.6	6	V	
V_{DDH}	Supply voltage for high level during programming	1	6.8	8.0	8.5	V	
t_r	Rise time	1	–	–	0.05	ms	
t_f	Fall time	1	–	–	0.05	ms	
t_{p0}	Bit time on V_{DD}	1	1.7	1.75	1.8	ms	t_{p0} is defined through the Sync bit
t_{pOUT}	Bit time on output pin	3	2	3	4	ms	t_{pOUT} is defined through the Acknowledge bit
t_{p1}	Voltage change for logical 1	1, 3	50	65	80	%	% of t_{p0} or t_{pOUT}
V_{DDPROG}	Supply voltage for programming the EEPROM	1	12.4	12.5	12.6	V	
t_{PROG}	Programming time for EEPROM	1	95	100	105	ms	
t_{rp}	Rise time of programming voltage	1	0.2	0.5	1	ms	
t_{fp}	Fall time of programming voltage	1	0	–	1	ms	
t_w	Delay time of programming voltage after Acknowledge	1	0.5	0.7	1	ms	
V_{act}	Voltage for an Activate pulse	3	3	4	5	V	
t_{act}	Duration of an Activate pulse	3	0.05	0.1	0.2	ms	

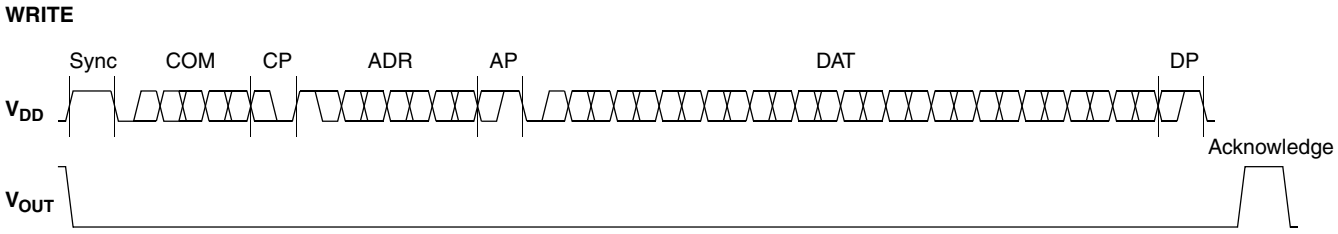

Fig. 5–2: Telegram for coding a Write command

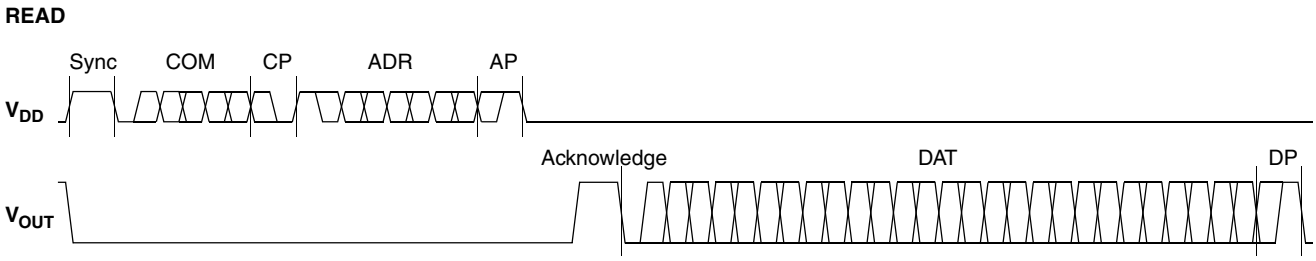

Fig. 5–3: Telegram for coding a Read command

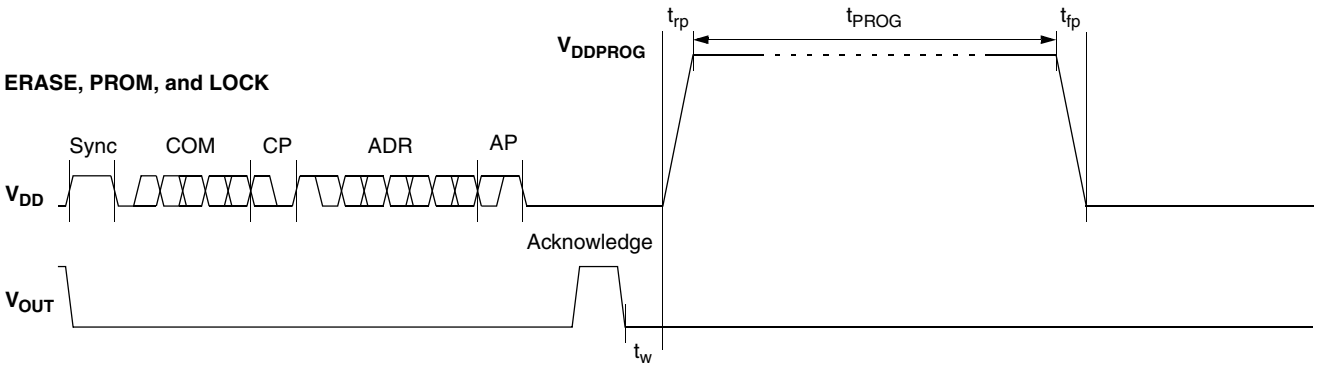

Fig. 5–4: Telegram for coding the EEPROM programming

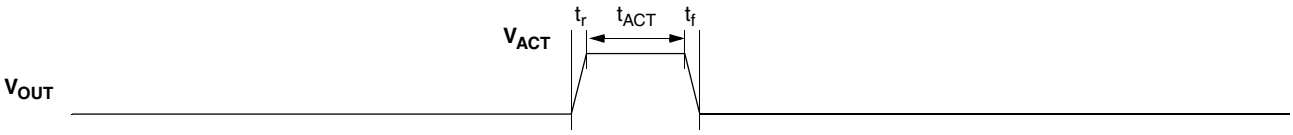

Fig. 5–5: Activate pulse

5.3.3. Telegram Codes

Sync Bit

Each telegram starts with the Sync bit. This logical “0” pulse defines the exact timing for t_{p0}.

Command Bits (COM)

The Command code contains 3 bits and is a binary number. Table 5–4 shows the available commands and the corresponding codes for the HAL805, HAL815, HAL817 and HAL1000 and Table 5-5 for the HAL82x, HAL83x, HAL880 and HAL1002 registers.

Command Parity Bit (CP)

This parity bit is “1” if the number of zeros within the 3 Command bits is uneven. The parity bit is “0”, if the number of zeros is even.

Address Bits (ADR)

The Address code contains 4 bits and is a binary number. Table 5–5 and Table 5-6 shows the available addresses for the HAL805, HAL815, HAL817 and HAL1000 registers and Table 5-5 for the HAL82x, HAL83x, HAL880 and HAL1002 registers.

Address Parity Bit (AP)

This parity bit is “1” if the number of zeros within the 4 address bits is uneven. The parity bit is “0” if the number of zeros is even.

Data Bits (DAT)

The 14 Data bits contain the register information.

The registers use a different number formats for the Data bits. These formats are explained in Section 5.3.4

In the Write command, the last bits are valid. If, for example, the TC register (6 bits) is written, only the last 6 bits are valid.

In the Read command, the first bits are valid. If, for example, the TC register (6 bits) is read, only the first 6 bits are valid.

Data Parity Bit (DP)

This parity bit is “1” if the number of zeros within the binary number is even. The parity bit is “0” if the number of zeros is uneven.

Acknowledge

After each telegram, the output answers with the Acknowledge signal. This logical “0” pulse defines the exact timing for t_{pOUT}.

Table 5–4: Available commands

Command	Code	Explanation
READ	2	read a register
WRITE	3	write a register
PROM	4	program all nonvolatile registers (except the lock bits)
ERASE	5	erase all nonvolatile registers (except the lock bits)
Please note: For HAL82x, HAL83x, HAL880 and HAL1002 the LOCK bit is set by using the WRITE command followed by a PROM and ERASE (store) sequence.		
LOCK	7	lock the whole device and switch permanently to the analog-mode (only for HAL805, HAL815, HAL817 and HAL1000)

5.3.4. Number Formats

Binary number: The most significant bit is given as first, the least significant bit as last digit.

Example:

101001 represents 41 decimal.

Signed binary number: The first digit represents the sign of the following binary number (1 for negative, 0 for positive sign).

Example:

0101001 represents +41 decimal
1101001 represents –41 decimal

Two's-complementary number: The first digit of positive numbers is "0", the rest of the number is a binary number. Negative numbers start with "1". In order to calculate the absolute value of the number, calculate the complement of the remaining digits and add "1".

Example:

0101001 represents +41 decimal
1010111 represents –41 decimal

Table 5–5: Available register addresses of HAL805, HAL815, HAL817 and HAL1000

Register	Code	Data Bits	Format	Customer	Remark
CLAMP LOW	1	10	binary	read/write/program	low clamping voltage
CLAMP HIGH	2	11	binary	read/write/program	high clamping voltage
VOQ	3	11	two compl. binary	read/write/program	
SENSITIVITY	4	14	signed binary	read/write/program	
MODE	5	6	binary	read/write/program	range and filter settings
LOCK	6	1	binary	lock	lock bit
ADC-READOUT	7	14	two compl. binary	read	
TC	11	6	signed binary	read/write/program	
TCSQ	12	5	binary	read/write/program	
DEACTIVATE	15	12	binary	write	deactivate the sensor

Table 5–6: Available register addresses of HAL82x, HAL83x, HAL880 and HAL1002

Register	Code	Data Bits	Format	Customer	Remark
CLAMP LOW	1	8	binary	read/write/program	low clamping voltage
CLAMP HIGH	2	9	binary	read/write/program	high clamping voltage
VOQ	3	11	two compl. binary	read/write/program	
SENSITIVITY	4	14	signed binary	read/write/program	
MODE (only HAL82x and HAL880)	5	8	binary	read/write/program	range, filter, output mode, interface bit time settings
MODE (only HAL83x)	5	10	binary	read/write/program	range, filter, output mode, offset correction settings
LOCK	6	2	binary	read/write/program	lock bit
GP1..3	8	13	binary	read/write/program	It is only possible to program this register if the mode register bit zero is set to 1.
D/A-READOUT	9	14	binary	read	Bit sequence is reversed during read sequence.
TC	11	10	binary	read/write/program	bit 0 to 2 TCSQ bit 3 to 7 TC bit 7 to 9 TC-RANGE
GP0	12	13	binary	read/write/program	It is only possible to program this register if the mode register bit zero is set to 1.
DEACTIVATE	15	12	binary	write	deactivate the sensor

5.3.5. Register Information (HAL805 / HAL815 / HAL817)

Terminology:

REGISTER: Name of the register or register value

Register: Name of the parameter

CLAMP LOW

– The register range is from 0 up to 1023.
– The register value is calculated by:

$$CLAMP\ LOW = \frac{Low\ Clamping\ Voltage}{V_{DD}} \cdot 2048$$

CLAMP HIGH

– The register range is from 0 up to 2047.
– The register value is calculated by:

$$CLAMP\ HIGH = \frac{High\ Clamping\ Voltage}{V_{DD}} \cdot 2048$$

VOQ

– The register range is from –1024 up to 1023.
– The register value is calculated by:

$$VOQ = \frac{V_{OQ}}{V_{DD}} \cdot 1024$$

SENSITIVITY

– The register range is from –8192 up to 8191.
– The register value is calculated by:

$$SENSITIVITY = Sensitivity \cdot 2048$$

TC and TCSQ

– The TC register range is from –31 up to 31.
– The TCSQ register range is from 0 up to 31.

Please refer to the data sheet for the recommended values.

MODE

– The register range is from 0 up to 63 and contains the settings for FILTER and RANGE:

$$MODE = FILTER \cdot 8 + RANGE$$

Please refer to the data sheet for the available FILTER and RANGE values.

ADC-READOUT

– This register is read only.

– The register range is from –8192 up to 8191.

DEACTIVATE

– This register can only be written.

– The register has to be written with 2063 decimal (80F hexadecimal) for the deactivation.

– The sensor can be reset with an Activate pulse on the output pin or by switching off and on the supply voltage.

Note: The register information of the HAL810 is slightly different. However it is quite easy to transform voltage values (HAL8x5) into output duty cycle (HAL810) by multiplication with

$$\frac{100\%}{5V}$$

For detailed register information of the HAL810, please refer to the data sheet.

5.3.6. Register Information (HAL82x / HAL83x / HAL880 / HAL1002)

Terminology:

REGISTER: Name of the register or register value

Register: Name of the parameter

CLAMP LOW

– The register range is from 0 up to 1023.
– The register value is calculated by:

$$CLAMP\ LOW = \frac{Low\ Clamping\ Voltage \times 2}{V_{DD}} \times 255$$

CLAMP HIGH

– The register range is from 0 up to 511.
– The register value is calculated by:

$$CLAMP\ HIGH = \frac{High\ Clamping\ Voltage}{V_{DD}} \times 511$$

VOQ

– The register range is from –1024 up to 1023.
– The register value is calculated by:

$$VOQ = \frac{V_{OQ}}{V_{DD}} \cdot 1024$$

SENSITIVITY

– The register range is from –8192 up to 8191.
– The register value is calculated by:

$$SENSITIVITY = Sensitivity \cdot 2048$$

TC

– The TC register range is from 0 up to 1023.

– The register value is calculated by:

$$TC = GROUP \times 256 + TCValue \times 8 + TCSQValue$$

Please refer to the data sheet for the recommended values.

MODE (HAL82x, HAL880)

– The register range is from 0 up to 255 and contains the settings for FILTER and RANGE:

$$MODE = OUTPUTMODE \times 32 + BITRATE \times 16 + FILTER \times 8 + RANGE \times 2 + EnableProgGPRegisters$$

Please refer the data sheet for the available FILTER and RANGE values.

MODE (HAL83x, HAL1002)

– The register range is from 0 up to 1023 and contains the settings for FILTER, RANGE, OUTPUTMODE and OFFSET CORRECTION:

$$MODE = RANGE \times 512 + SIGNOC \times 256 + OUTPUTMODE \times 32 + FILTER \times 8 + RANGE \times 2 + OFFSETCORRECTION$$

Please refer the data sheet for the available RANGE, OUTPUTMODE, FILTER and OFFSETCORRECTION values.

D/A-READOUT

– This register is read only.

– The register range is from 0 up to 16383.

DEACTIVATE

– This register can only be written.

– The register has to be written with 2063 decimal (80F hexadecimal) for the deactivation.

The sensor can be reset with an Activate pulse on the output pin or by switching off and on the supply voltage.

5.3.7. Programming Information

If the content of any register (except the lock registers) is to be changed, the desired value must first be written into the corresponding RAM register. Before reading out the RAM register again, the register value must be permanently stored in the EEPROM.

Permanently storing a value in the EEPROM is done by first sending an ERASE command followed by sending a PROM command. For HAL 82x, HAL83x, HAL880 and HAL1002 the address within the ERASE and PROM commands **must be zero**. ERASE and PROM act on all registers in parallel.

Note: To store data in the GP register (only HAL82x, HAL83x, HAL880 and HAL1002) it is necessary to set bit number 0 of the MODE register to one, before sending an ERASE and PROM command. Otherwise the data stored in the GP register will not be changed.

If all HAL805, HAL815, HAL817, HAL82x, HAL83x, HAL880, HAL1000 and HAL1002 registers are to be changed, all writing commands can be sent one after the other, followed by sending one ERASE and PROM command at the end.

During all communication sequences, the customer has to check if the communication with the sensor was successful. This means that the acknowledge and the parity bits sent by the sensor have to be checked by the customer. If the Micronas programmer board is used, the customer has to check the error flags sent from the programmer board.

Note: For production and qualification tests, it is mandatory to set the Lock bit after final adjustment and programming. The Lock function is active after the next power-up of the sensor. In case of HAL 805/815/817/810/1000 Micronas also recommends sending an additional ERASE command after sending the LOCK command (This is done automatically when the board command “l” is used for locking the sensor.).
The success of the Lock process should be checked by reading at least one sensor register after locking and/or by an analog check of the sensor’s output signal.
Electrostatic Discharges (ESD) may disturb the programming pulses. Please take precautions against ESD.

5.4. Data Formats

The protocol between the programmer board and the Hall sensor defines fixed lengths for the data, command, and address words to be transmitted:

– Command (CMD): 3 bit

– Address (ADR): 4 bit

– Data (DAT): 14 bit

– Command Parity (CP): 1 bit

– Address Parity (AP): 1 bit

– Data Parity (DP): 1 bit

Reminder: For communication with the programmer board, the 14 data bits are encoded in four ASCII characters: DAT3, DAT2, DAT1, and DAT0.

As every register has a different length, not all of the 14 bits are used. Furthermore, there is a difference between the write format (transfer board → IC) and the read format (transfer IC → board). Table 5–7 and Table 5-7 show which of the 14 bits are valid for each register (write format and read format).

Table 5–7: Data formats of HAL805, HAL810, HAL815, HAL817 and HAL1000

Register	Char	DAT3				DAT2				DAT1				DAT0			
	Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CLAMP LOW	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–
CLAMP HIGH	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
VOQ	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SENSITIVITY	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
MODE	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
ADC-READOUT	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
TC	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
TCSQ	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
DEACTIVATE	Write	–	–	-	-	1	0	0	0	0	0	0	0	1	1	1	1
V: valid, –: ignore, bit order: MSB first.																	

Table 5–8: Data formats of HAL82x, HAL83x, HAL880 and HAL1002

Register	Char	DAT3				DAT2				DAT1				DAT0			
	Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CLAMP LOW	Write	–	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	–	–	–	–	–	–
CLAMP HIGH	Write	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	–	–	–	–	–
VOQ	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SENSITIVITY	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
MODE (onlyHAL82x and HAL880)	Write	–	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	–	–	–	–	–	–
MODE (only HAL83x)	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–
LOCKR	Write	–	–	–	–	–	–	–	–	–	–	–	–	–	–	V	V
	Read	–	–	V	V	–	–	–	–	–	–	–	–	–	–	–	–
GP1..3	Write	–	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	–
D/A-READOUT	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
TC	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–
GP0	Write	–	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	–
DEACTIVATE	Write	–	–	–	–	1	0	0	0	0	0	0	0	1	1	1	1

V: valid, –: ignore, bit order: MSB first

5.5. The Programming Procedure

5.5.1. Programming Procedure for HAL805, HAL810, HAL815, HAL817 and HAL1000

The general programming procedure is described in detail for the HAL805 (HAL815/ HAL817).

The procedure for the HAL810 is quite similar. However, the multiprogramming (i.e. the programming of more than one sensor connected to the same supply voltage) does not apply for the HAL810. Additionally, some register names are different for the HAL810.

Table 5–9: Cross reference register names

Register HAL8x5/ HAL817	Register HAL810
CLAMP LOW	MIN DUTY
CLAMP HIGH	MAX DUTY
SENSITIVITY	DC SENSITIVITY
VOQ	DCOQ

An example of a programming procedure is given in Fig. 5–6.

First, the programmer board has to be initialized. If more than one sensor is connected to the same supply line, the sensors have to be addressed sequentially. This is done by the multiprogramming loop: After deactivating all sensors, the sensor to be programmed is activated again by sending a pulse on the corresponding output line, followed by a dummy read command.

Note: The flow chart given in Fig. 5–6 is intended as a simple example. The multiprogramming loop can also be done within the calibration procedure.

Please keep in mind that the board commands “e...”, “q...”, “m...”, and “t” cause the board to send back an eight character string to the PC. This string must be read out of the serial port before sending the next command.

Example:

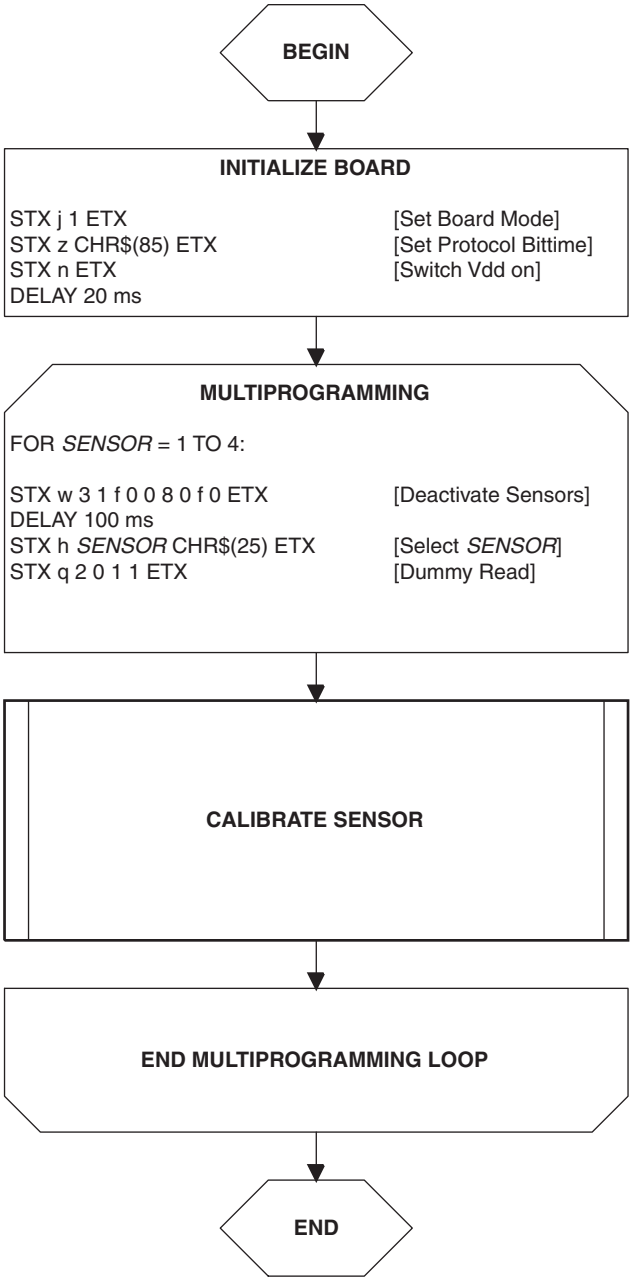

Fig. 5–6: General procedure

5.5.2. The Calibration Procedure

The output characteristic is determined by four parameters:

– Clamp Low specifies the lower clamping voltage of the output curve.

– Clamp High specifies the higher clamping voltage of the output curve.

– V_{OQ} corresponds to the output voltage at zero ADC-Readout.

– Sensitivity corresponds to the increase of the output voltage with magnetic field. A sensitivity of “1” is equivalent to a voltage increase of 5 V at an increase of 2048 of the ADC-Readout.

The following simple formula relates the analog output voltage to the ADC-Readout (V_{DD} is considered to be 5.0 V):

$$V_{OUT} = \frac{Sensitivity \cdot \text{ADC-Readout} \cdot 5\ \text{V}}{2048} + V_{OQ}$$

The calibration procedure is shown in Fig. 5–7. The register values are intended as an example:

Step 1: Programming of the parameters which do not require individual adjustment:

– Clamp Low, Clamp High, Filter Frequency, Magnetic Range, TC, and TCSQ.

Step 2: Get the ADC-Readout value in the first calibration point and assign a nominal output voltage ($V_{OUT,1}$) to that point.

Step 3: Get the ADC-Readout value in the second calibration point and assign a nominal output voltage ($V_{OUT,2}$) to that point.

Step 4: Calculate and program the values of SENSITIVITY and VOQ.

$$Sensitivity = \frac{(V_{OUT2} - V_{OUT1})}{(ADCReadout_2 - ADCReadout_1)} \cdot \frac{2048}{5V}$$

$$SENSITIVITY = Sensitivity \cdot 2048$$

$$V_{OQ} = V_{OUT1} - ADCReadout_1 \cdot Sensitivity \cdot \frac{5V}{2048}$$

$$VOQ = V_{OQ} \cdot \frac{1024}{5V}$$

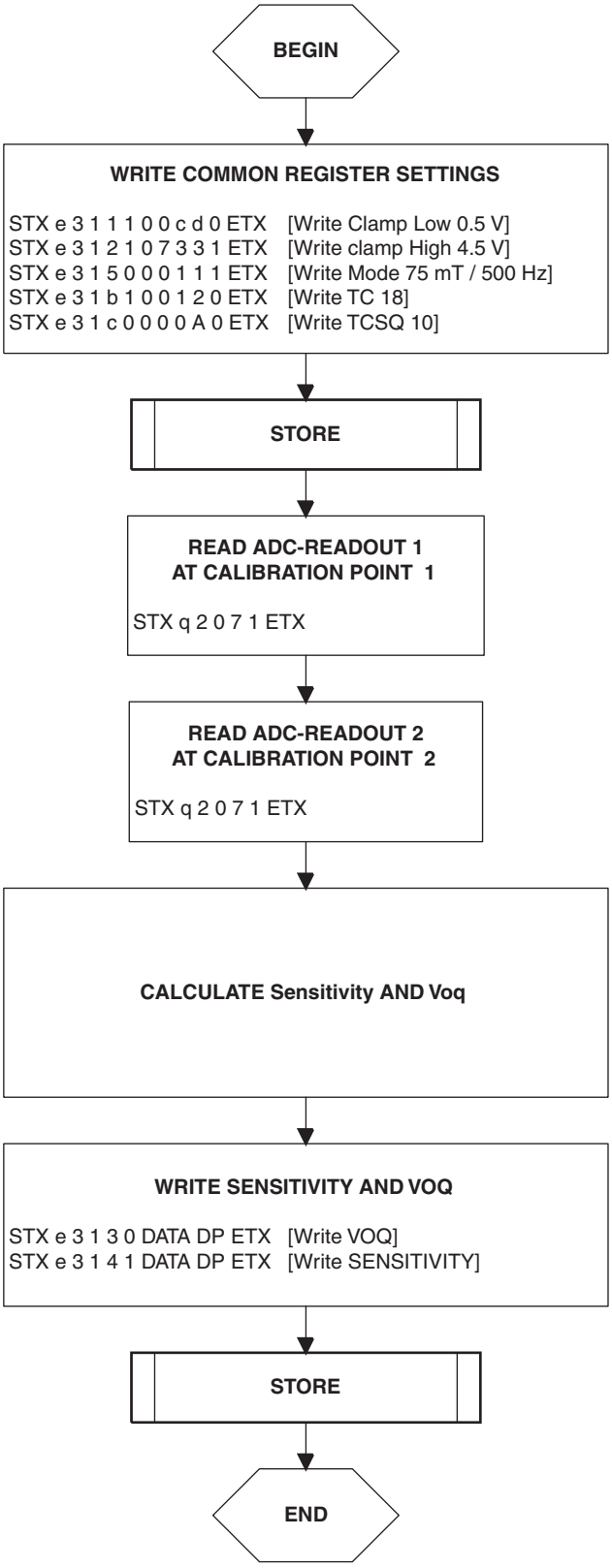

Fig. 5–7: Calibration procedure

Step 5:

Store the registers permanently.

The “Store” sequence is given in Fig. 5–8. The permanent storing of data into the EEPROM memory is a two-step process. First, all zero bits (ERASE), then all one bits (PROM) are written.

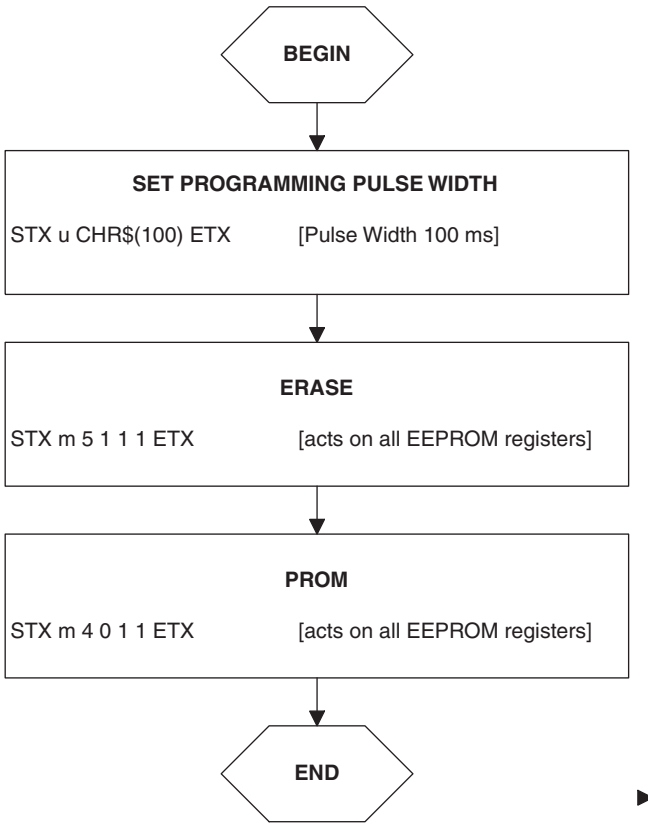

Fig. 5–8: Store sequence

5.5.3. Programming Procedure for HAL82x, HAL83x, HAL880 and HAL1002

The general programming procedure is described in detail for the HAL82x, HAL83x, HAL880 and HAL1002.

An example of a programming procedure is given in Fig. 5–6.

First, the programmer board has to be initialized. If more than one sensor is connected to the same supply line, the sensors have to be addressed sequentially. This is done by the multiprogramming loop: After deactivating all sensors, the sensor to be programmed is activated again by sending a pulse on the corresponding output line, followed by a dummy read command.

Note: The flow chart given in Fig. 5–6 is intended as a simple example. The multiprogramming loop can also be done within the calibration procedure.

Please keep in mind that the board commands “e...”, “q...”, “m...”, and “t” cause the board to send back an eight character string to the PC. This string must be read out of the serial port before sending the next command.

Example:

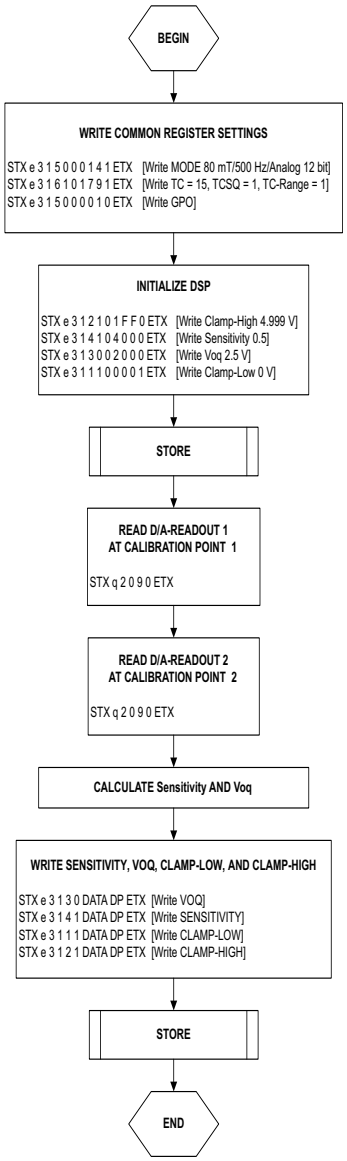

Fig. 5–9: General procedure

5.5.4. The Calibration Procedure

The output characteristic is determined by four parameters:

– Clamp Low specifies the lower clamping voltage of the output curve.

– Clamp High specifies the higher clamping voltage of the output curve.

– V_{OQ} corresponds to the output voltage at zero magnetic field.

– Sensitivity describes the magnetic sensitivity of the sensor.

The following simple formula relates the analog output voltage to the D/A-Readout (V_{DD} is considered to be 5.0 V):

$$\Delta V_{OUT} = \frac{SENSITIVITY}{16384} \cdot 2 \cdot \Delta DA\text{-}Readout \cdot 5$$

The calibration procedure is shown in Fig. 5–7. The register values are intended as an example:

Step 1: Programming of the parameters which do not require individual adjustment:

– Filter Frequency, Magnetic Range, Output Mode, TC (TC, TCSQ and TC-Range) and GP.

Step 2: Initialize DSP

As the D/A-READOUT register value depends on the settings of SENSITIVITY, VOQ and CLAMP LOW/HIGH, these registers have to be initialized with defined values, first:

– $VOQ_{INITIAL}$ = 2.5 V

– $Sensitivity_{INITIAL}$ = 0.5

– Clamp Low = 0 V

– Clamp High = 4.999 V

Step 3: Get the DA-Readout value in the first calibration point and assign a nominal output voltage ($V_{OUT,1}$) to that point.

Step 4: Get the DA-Readout value in the second calibration point and assign a nominal output voltage ($V_{OUT,2}$) to that point.

Step 5: Calculate and program the values of SENSITIVITY and VOQ.

$$Sensitivity = \frac{1}{2} \times \frac{(Vout2 - Vout1)}{(D/A\text{-}Readout2 - D/A\text{-}Readout1)} \times \frac{16384}{5}$$

$$V_{OQ} = \frac{1}{16} \times \left[\frac{Vout2 \times 16384}{5} - [(D/A\text{-}Readout2 - 8192) \times Sensitivity \times 2] \right] \times \frac{5}{1024}$$

Write and program values for Clamp High and Clamp Low

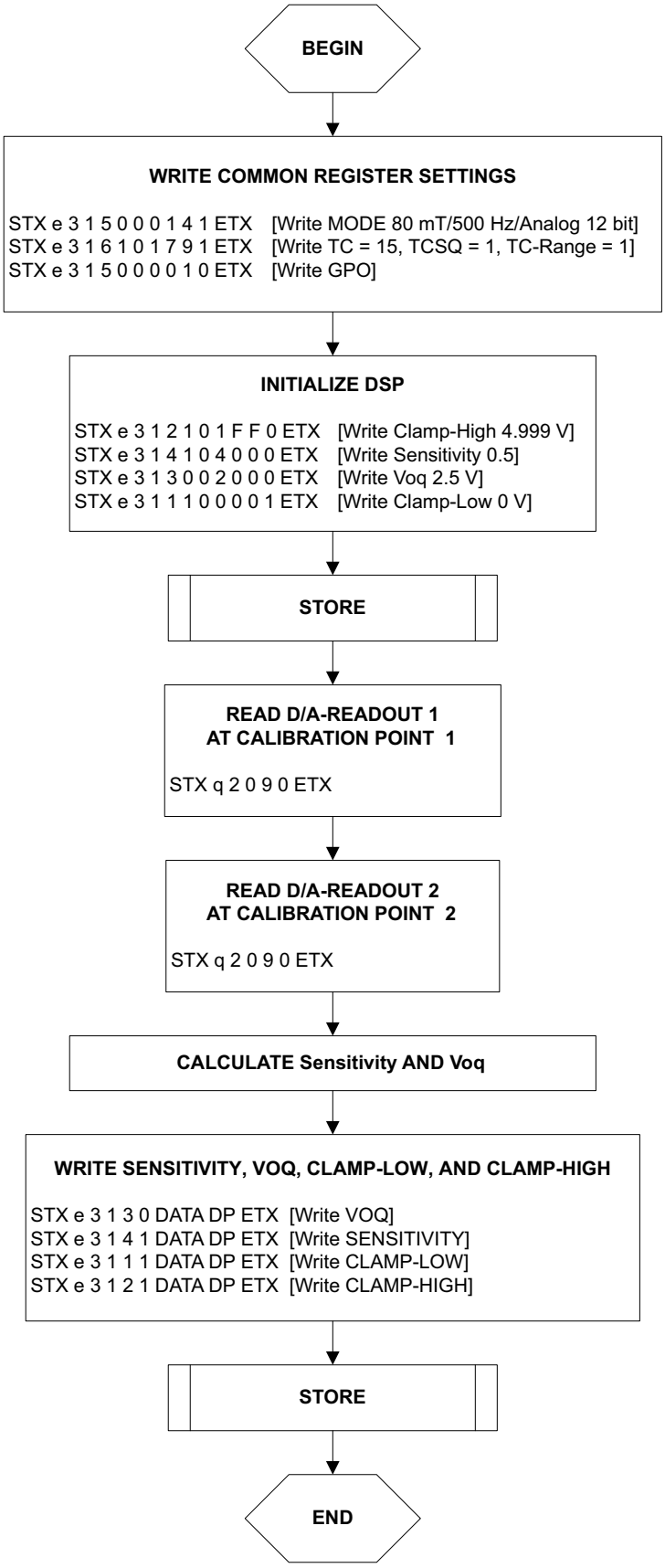

Fig. 5–10: Calibration procedure

Step 6:

Store the registers permanently.

The “Store” sequence is given in Fig. 5–8. The permanent storing of data into the EEPROM memory is a two-step process. First, all zero bits (ERASE), then all one bits (PROM) are written.

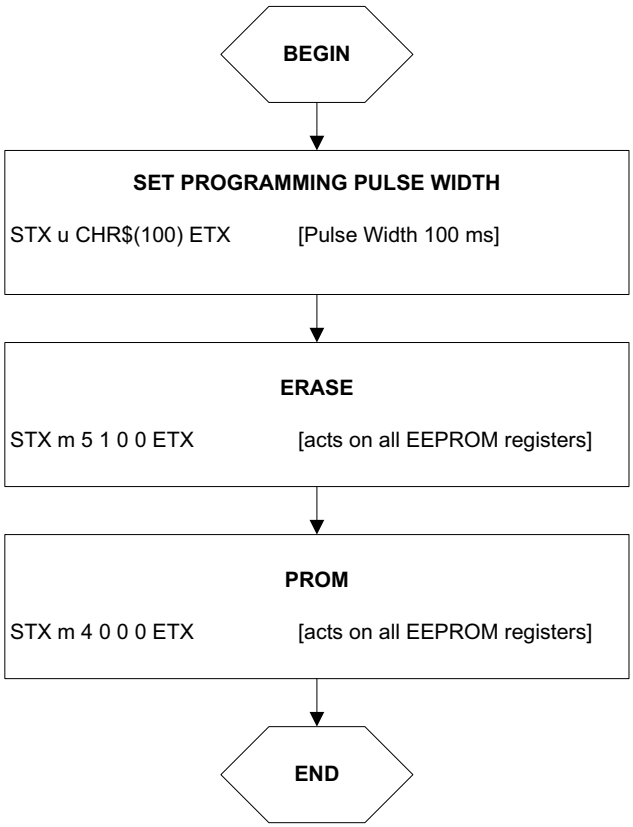

Fig. 5–11: Store sequence

6. Operation Mode 2

HAL855

This operation mode is set by sending the “j2” command.

Communication: PC → prog. board

Section 6.1 and Section 6.2 contain a detailed description of communication between the PC and the programmer board V 5.1. Particularly, all board commands available in the operation mode “j2” are listed.

Communication: prog. board → sensor

The communication between the programmer board and the Hall sensor is described in Section 6.3

Data formats

The data formats for communication between both the PC and the board, and the board and the sensor are summarized in Section 6.4

Programming procedure

Finally, Section 6.5 depicts the programming procedure and gives an example of a complete programming sequence.

6.1. Definition of the Protocol

The general syntax is as follows (blanks are given for better readability only):

PC → Board:
STX BCMD [CMD CP ADR AP] [DAT3 DAT2 DAT1 DAT0 DP] ETX

Board → PC:
STX STATUS DAT3 DAT2 DAT1 DAT0 DP ETX

The characters in brackets [...] are optional, depending on the board command BCOM.

Table 6–1: Description of the characters

Character	Description
STX	ASCII character 2
ETX	ASCII character 3
BCMD	Board command
CMD	Command
CP	Command parity
ADR	Address
AP	Address parity
DAT3 DAT2 DAT1 DAT0	Data characters, each encoding 4 bit in HEX format. Example: 0 1 f 0 (HEX) = 496
DP	Data parity
STATUS	Status/error flag

6.2. The Board Commands BCMD

The board commands activate functions of the firmware. The characters following the BCMD are the parameters of this function. Some functions serve to control the board operation and do not address the Hall device.

The read, write, program, and lock functions communicate with the Hall device. The parameters of these functions are converted into a serial telegram and sent to the connected sensor.

Table 6–2: Description of the board commands

BCMD	Parameters	Description
n	–	Switch V_{DD} on
o	–	Switch V_{DD} off
z[t]	t = 30...255 (as ASCII character) default: t = 85 (HAL855)	Set bit time slow mode bit time = t × 0.02 ms default bit time = 1.7 ms (HAL855)
u[t]	t = 30...255 (as ASCII character) HAL855: set t = 100	Set programming pulse width width = t × 1.00 ms HAL855: set pulse width = 100 ms.
y[t]	t = 10...255 HAL855: set t = 11	Delay Acknowledge width = t × 64.0 µs HAL855 set width = 0.7 ms
h[nt]	n = 1... 4 (as HEX) t = 10...255 (as ASCII character) recommended t = 25	Select sensor n, (default = 1) select pulse width = t × 4.0 µs recommended pulse width = 100 µs
e[p]	p = CMD CP ADR AP DAT3 DAT2 DAT1 DAT0 DP	Write HAL and echo data to PC
q[p]	p = CMD CP ADR AP	Read HAL and echo data to PC
c[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2 CMD3 CP3 ADR3 AP3	Program HAL and echo answer to PC
k[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2	Program LOCK bit and echo answer to PC
i		Read Biphase M sequence

Table 6–2: Description of the board commands, continued

BCMD	Parameters	Description
m[p]	p = 0 ... 7 (as HEX)	Read PWM sequence p = PWM period (2^p in ms).
v	–	Request firmware version
t	–	Request status and echo data to PC
j[p]	p = 0...3 (as HEX)	Switch board operation mode 0 = EMU Board V 4.1 (default) 1 = HAL805, HAL815, HAL817, HAL810, HAL 82x, HAL83x, HAL 880, HAL100x 2 = HAL855 3 = HAL856

Examples:

(Blanks are given for better readability only)

– set board operation mode:
STX j 2 ETX

– power on:
STX n ETX

– set bit time slow CHR$(85) = 'U':
STX z U ETX

– read register no. 2:
STX q 0 1 2 1 ETX

– write register no. 2:
STX e 3 1 2 1 0 0 0 A 1 ETX

– write output characteristic setpoint 0:
STX e 1 0 0 0 0 0 0 0 1 ETX

– write output characteristic setpoint 16:
STX e 2 0 0 0 0 1 0 0 0 ETX

– read output characteristic setpoint 15:
STX e 6 1 f 0 ETX

– read output characteristic setpoint 31:
STX q 7 0 f 0 ETX

– store sequence:
STX u CHR$(100) ETX (set programming time)
STX y CHR$(11) ETX (set delay of Acknowledge)
STX c 5 1 3 0 4 0 3 0 6 1 0 0 ETX
(ERASE-PROM-READ)

– deactivate sensors:
STX e 3 1 F 0 0 8 0 F 0 ETX

– select sensor 1:
STX h 1 CHR$(25) ETX

– lock sensors
STX e 3 1 6 0 0 0 0 1 0 ETX (set Lock bit)
STX u CHR$(100) ETX (set programming time)
STX y CHR$(11) ETX (set delay of Acknowledge)
STX k 4 0 6 0 0 1 3 0 ETX (PROM-READ)

6.2.1. Monitoring of the Supply Voltage

In case of ERASE, PROM, and LOCK, the supply voltage is measured by the board, and a data string is sent back to the PC. The relation between the data and the supply voltage is

$$V_{PROG} = \frac{DAT}{4095} \cdot 6 \cdot 2.485\ \text{V}$$

If the supply voltage is out of the specification limits, the status bit is set to “1”.

Examples:

– V_{PROG} = 4.80 V, Data board → PC:
STX 1 0 5 2 6 0 ETX
(DAT = 0 5 2 6)

– V_{PROG} = 5.00 V, Data board → PC:
STX 0 0 5 5 D 0 ETX
(DAT = 0 5 5 D)

6.3. Programming of the Sensor

6.3.1. Definition of Programming Pulses

The sensor is addressed by modulating a serial telegram on the supply voltage. The sensor answers with a serial telegram on the output pin.

The bits in the serial telegram have a different bit time for the V_{DD}-line and the output. The bit time for the V_{DD}-line is defined through the length of the Sync bit at the beginning of each telegram. The bit time for the output is defined through the Acknowledge bit.

A logical “0” is coded as no voltage change within the bit time. A logical “1” is coded as a voltage change between 50% and 80% of the bit time. After each bit, a voltage change occurs.

6.3.2. Definition of the Telegram

Each telegram starts with the Sync bit (logical 0), 3 bits for the Command (COM), the Command Parity bit (CP), 4 bits for the Address (ADR), and the Address Parity bit (AP).

There are 4 kinds of telegrams:

Write a register

Write a register
After the AP bit, follow 14 Data bits (DAT) and the Data Parity bit (DP). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge bit (logical 0) on the output (see Fig. 6–2).

Read a register

After evaluating this command, the sensor answers with the Acknowledge bit, 14 Data bits, and the Data Parity bit on the output (see Fig. 6–3).

Programming the EEPROM cells

In order to permanently store the written data into the EEPROM cells, an erase and program command have to be sent to the sensor. After the recognition of the erase and program commands, the HAL855 answers with an acknowledge pulse on its output signal. After the acknowledge pulse, a pulse on the V_{DD}-line is created to start the charging of the EEPROM cells. Then, the supply voltage is kept constant during the charging time. To stop the charging, a further command is sent to the HAL855. This stopping command can be a further programming command or a read command (see Fig. 6–5).

Activate a sensor

If more than one sensor is connected to the supply line, selection can be done by first deactivating all sensors. With an Activate pulse on the appropriate output pin, an individual sensor can be selected. All following commands will only be accepted from the activated sensor (see Fig. 6–4).

Lock a sensor

To lock the EEPROM registers, the lock bit has to be programmed. Write the lock bit into the lock register (see Fig. 6–2). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge Bit (logical 0) on the output. In order to store the lock bit permanently, an erase and program command have to be sent to the sensor. This is the same procedure as mentioned above (see Programming the EEPROM cells and Fig. 6–5). The EEPROM registers are locked **after** a power on reset.

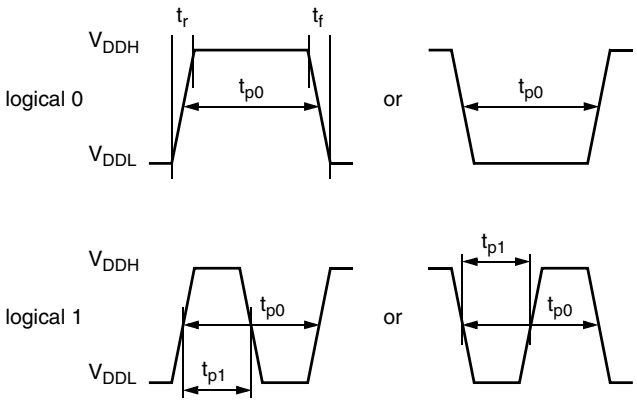

Fig. 6–1: Definition of logical 0 and 1 bit

Table 6–3: Telegram parameters

Symbol	Parameter	Pin	Min.	Typ.	Max.	Unit	Remarks
V_{DDL}	Supply voltage for low level during programming	1	5	5.6	6	V	
V_{DDH}	Supply voltage for high level during programming	1	6.8	8.0	8.5	V	
V_{DDPROG}	Supply voltage for permanent storing of the data	1	4.9	5.0	5.1	V	
t_r	Rise time	1	–	–	0.05	ms	
t_f	Fall time	1	–	–	0.05	ms	
t_{p0}	Bit time on V_{DD}	1	1.7	1.75	1.8	ms	t_{p0} is defined through the Sync bit
t_{pOUT}	Bit time on output pin	3	2	3	4	ms	t_{pOUT} is defined through the Acknowledge bit
t_{p1}	Voltage change for logical 1	1, 3	50	65	80	%	% of t_{p0} or t_{pOUT}
t_{PROG}	Programming time for EEPROM	1	95	100	105	ms	
t_{rp}	Rise time of charging pulse	1	0.2	0.5	1	ms	
t_{fp}	Fall time of charging pulse	1	0	–	1	ms	
t_w	Delay time of charging pulse after Acknowledge	1	0.5	0.7	1	ms	
V_{act}	Voltage for an Activate pulse	3	0	0	0.5	V	
t_{act}	Duration of an Activate pulse	3	0.05	0.1	0.2	ms	

WRITE

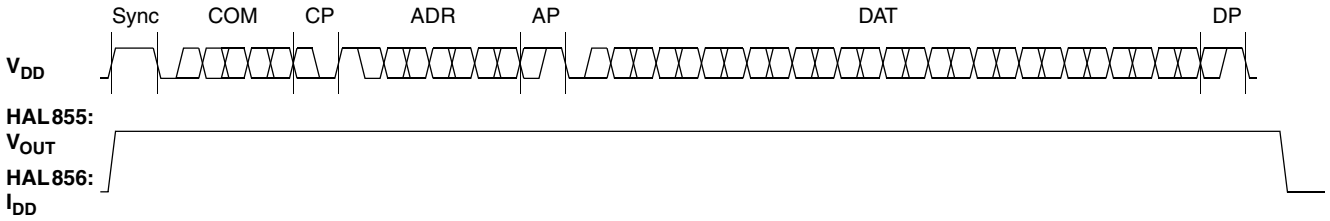

Fig. 6–2: Telegram for coding a Write command

READ

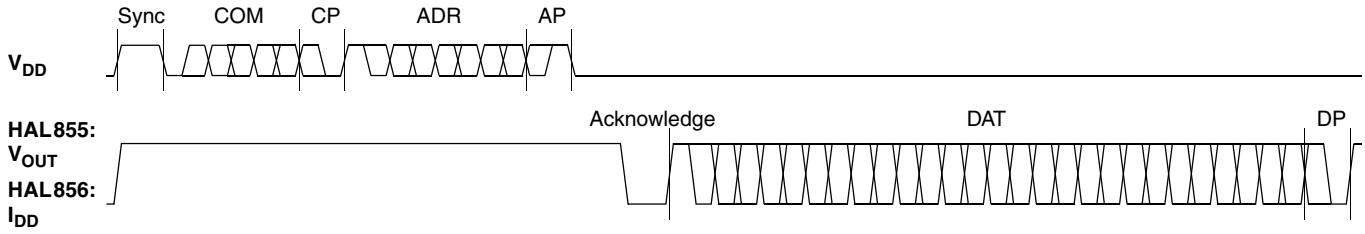

Fig. 6–3: Telegram for coding a Read command

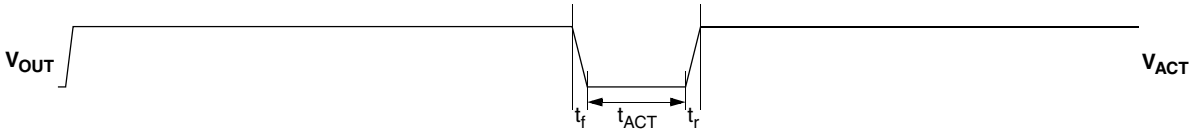

Fig. 6–4: Activate pulse

6.3.3. Telegram Codes

Sync Bit

Each telegram starts with the Sync bit. This logical “0” pulse defines the exact timing for t_{p0}.

Command Bits (COM)

The Command code contains 3 bits and is a binary number. Table 6–4 shows the available commands and the corresponding codes for the HAL855.

Command Parity Bit (CP)

This parity bit is “1” if the number of zeros within the 3 command bits is uneven. The parity bit is “0”, if the number of zeros is even.

Address Bits (ADR)

The Address code contains 4 bits and is a binary number. Table 6–5 shows the available addresses for the HAL855 registers.

Address Parity Bit (AP)

This parity bit is “1” if the number of zeros within the 4 address bits is uneven. The parity bit is “0” if the number of zeros is even.

Data Bits (DAT)

The 14 Data bits contain the register information.

The registers use different number formats for the Data bits. These formats are explained in Section 6.3.4

In the Write command, the last bits are valid. If, for example, the TC register (6 bits) is written, only the last 6 bits are valid.

In the Read command, the first bits are valid. If, for example, the TC register (6 bits) is read, only the first 6 bits are valid.

Data Parity Bit (DP)

This parity bit is “1” if the number of zeros within the binary number is even. The parity bit is “0” if the number of zeros is uneven.

Acknowledge

After each telegram, the output answers with the Acknowledge signal. This logical “0” pulse defines the exact timing for t_{pOUT}.

Table 6–4: Available commands

Command	Code	Explanation
READ	0	read a Setup EEPROM register (like TC, TCSQ, magnetic range, etc.)
READL	6	read a Characteristics EEPROM register (setpoints 0 to 15)
READH	7	read a Characteristics EEPROM register (setpoints 16 to 31)
WRITE	3	write a Setup EEPROM register (like TC, TCSQ, Magnetic range, etc.)
WRITEL	1	write a Characteristics EEPROM register (setpoints 0 to 15)
WRITEH	2	write a Characteristics EEPROM register (setpoints 16 to 31)
PROM	4	program all non-volatile registers
ERASE	5	erase all non-volatile registers
Please note: The Lock bit is set by using the WRITE command followed by a PROM.		

6.3.4. Number Formats

Binary number: The most significant bit is given as first, the least significant bit as last digit.

Example:

101001 represents 41 decimal.

Signed binary number: The first digit represents the sign of the following binary number (1 for negative, 0 for positive sign).

Example:

0101001 represents +41 decimal
1101001 represents –41 decimal

Two's-complementary number: The first digit of positive numbers is "0", the rest of the number is a binary number. Negative numbers start with "1". In order to calculate the absolute value of the number, calculate the complement of the remaining digits and add "1".

Example:

0101001 represents +41 decimal
1010111 represents –41 decimal

Table 6–5: Available register addresses

Register	Code	Data Bits	Format	Customer	Remark
CURRENT-SOURCE	1	10	binary	read/write/program	In case of HAL855: used to define output slew rate In case of HAL856: used to define output current levels (I_{DD_HIGH} and I_{DD_LOW})
PARTNUMBER	2	11	binary	read/write/program	only with Biphase-M mode
SHIFT	3	11	two's compl.	read/write/program	
SLOPE	4	14	signed binary	read/write/program	
MODE	5	14	binary	read/write/program	range, filter, and output format settings
LOCK	6	1	binary	write/program	lock bit
DIGITAL READOUT	7	12	binary	read	digital value after signal processing
OFFSET CORRECTION	8	5	two's compl. (4 LSBs)	read/write/program	compensation of system offsets
SPECIALCUST.	9	6	binary	read/write/program	special customer register to define Biphase-M bittime and Partnumber Enable
TC	11	6	signed binary (6 LSBs)	read/write/program	linear temperature coefficient
TCSQ	12	5	binary	read/write/program	quadratic temperature coefficient
DEACTIVATE	15	11	binary	write	deactivate the sensor
CURVE LOW	0...15	9	binary	write/read/program	setpoints 0 to 15
CURVE HIGH	0...15	9	binary	write/read/program	setpoints 16 to 31

6.3.5. Register Information

Terminology

REGISTER: Name of the register or register value

Register: Name of the parameter

PARTNUMBER

– The register range is from 0 up to 2047.

SHIFT

– The register range is from –1024 up to 1023.

– The register value is calculated by:

$$SHIFT = \frac{Shift}{100\%} \cdot 1024$$

SLOPE

– The register range is from –8192 up to 8191.

– The register value is calculated by:

$$SLOPE = Slope \cdot 2048$$

TC and TCSQ

– The TC register range is from –31 up to 31.

– The TCSQ register range is from 0 up to 31.

Please refer to the data sheet for the recommended values.

Note: The word length of the TC register is 7 bit. The six LSBs represent a signed binary number. The MSB has to be ignored.

MODE

– The register range is from 0 up to 16383 and contains the settings for PERIOD, FORMAT, FILTER, and RANGE:

$$MODE = PERIOD \cdot 512 + FORMAT \cdot 64 + FILTER \cdot 8 + RANGE$$

Please refer to the data sheet for the available PERIOD, FORMAT, FILTER, and RANGE values.

DIGITAL-READOUT

– This register is read only.

– The register range is from 0 up to 4095.

OFFSET CORRECTION

– The register range is from 0 to 31

– The MSB is set to activate the offset correction.

CURRENTSOURCE

– The register range is from 0 to 1023 and contains the settings for LOW CURRENT, HIGH CURRENT, and SLEW RATE:

$$CURRENTSOURCE = SLEW\ RATE \times 256 + LOW\ CURRENT \times 8 + HIGH\ CURRENT$$

SPECIAL CUSTOMER

– The register range is from 0 to 63 and contains the settings for OUTPUT BITTIME and PARTNUMBER ENABLE:

$$SPECIAL\ CUSTOMER = PARTNUMBER\ ENABLE \times 16 + OUTPUT\ BITTIME$$

DEACTIVATE

– This register can only be written.

– The register has to be written with 2063 decimal (80F hexadecimal) for the deactivation.

– The sensor can be reset with an Activate pulse on the output pin or by switching off and on the supply voltage.

6.4. Data Formats

The protocol between the programmer board and the Hall sensor defines fixed lengths for the data, command, and address words to be transmitted:

– Command (CMD): 3 bit
– Address (ADR): 4 bit
– Data (DAT): 14 bit
– Command Parity (CP): 1 bit
– Address Parity (AP): 1 bit
– Data Parity (DP): 1 bit

Reminder: For communication with the programmer board, the 14 data bits are encoded in four ASCII characters: DAT3, DAT2, DAT1, and DAT0.

As every register has a different length, not all of the 14 bits are used. Furthermore, there is a difference between the write format (transfer Board → IC) and the read format (transfer IC → Board). Table 6–6 shows which of the 14 bits are valid for each register (write format and read format).

Table 6–6: Data formats

Register	Char	DAT3				DAT2				DAT1				DAT0			
	Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PART NUMBER	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SHIFT	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SLOPE	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
MODE	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
DIGITAL OUTPUT	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
SPECIAL CUSTOMER	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
TC	Write	–	–	–	–	–	–	–	–	–	(V)	V	V	V	V	V	V
	Read	–	–	(V)	V	V	V	V	V	V	–	–	–	–	–	–	–
TCSQ	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
DEACTIVATE	Write	–	–	–	–	1	0	0	0	0	0	0	0	1	1	1	1
CURVE LOW	Write	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	–	–	–	–	–
CURVE HIGH	Write	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	–	–	–	–	–
OFFSET CORRECTION	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
CURRENT SOURCE	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–

V: valid, (V): This bit must be ignored, –: ignore, bit order: MSB first.

6.5. The Programming Procedure

The general programming procedure is described in detail for the HAL855.

An example for a programming procedure is given in Fig. 6–6.

First, the programmer board has to be initialized. If more than one sensor is connected to the same supply line, the sensors have to be addressed sequentially. This is done by the multiprogramming loop: After deactivating all sensors, the sensor to be programmed is activated again by sending a pulse on the corresponding output line, followed by a dummy read command.

Note: The flow chart given in Fig. 6–6 is intended as a simple example. The multiprogramming loop can also be done within the calibration procedure.

Example:

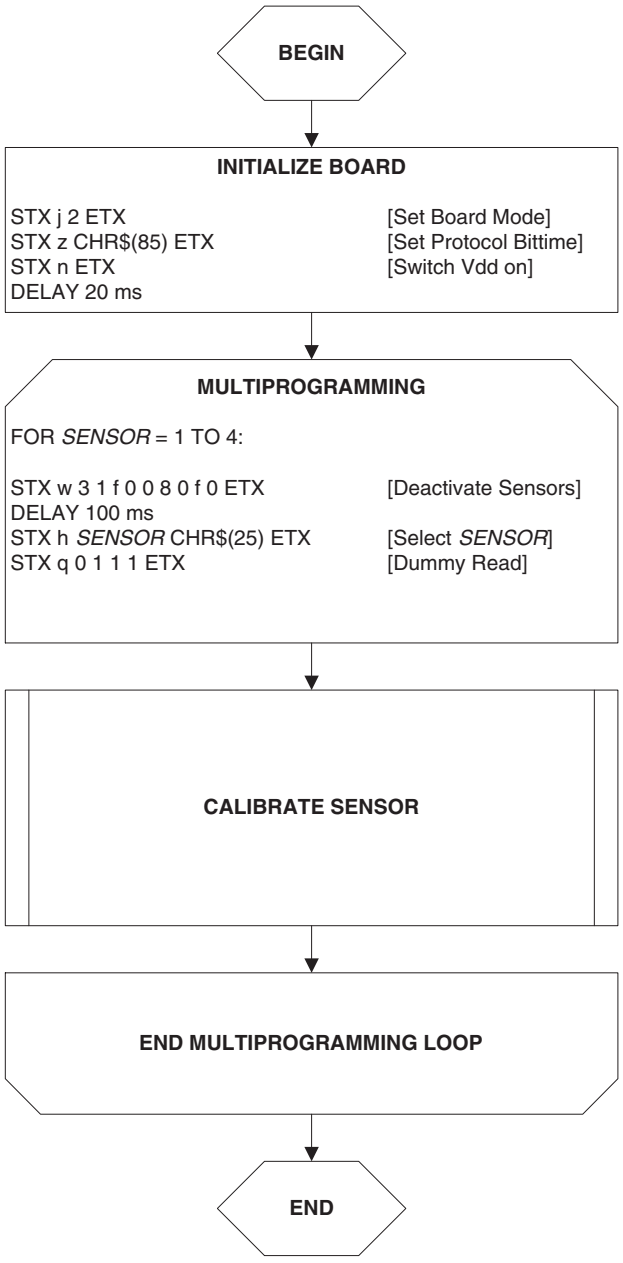

Fig. 6–6: General procedure

6.5.1. The Programmable Output Characteristic

The output characteristic defines the shape of the sensor output signal. It consists of 32 setpoints. Each setpoint can be set to values between 0 and 511 LSB. The output characteristic has to be monotonic increasing (Setpoint0 < Setpoint1 < ... < Setpoint31)

An example of how the setpoints are programmed is given in Fig. 6–7.

Note: In case of a non-linear output characteristic, the parameters SHIFT and SLOPE are related to the (virtual) straight line connecting the first and the last setpoints (setpoint 0 and setpoint 31, respectively).

Example:

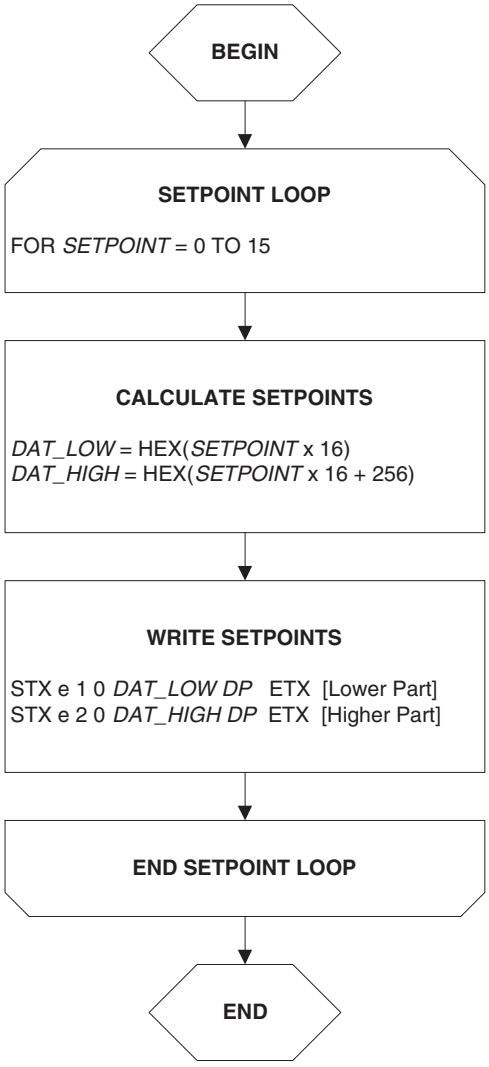

Fig. 6–7: Definition of the Output Characteristic “Linear (Standard)”

6.5.2. The Calibration Procedure

The output characteristic is determined by the following parameters:

– Shift corresponds to the output voltage at an ADC value of zero.

– Slope corresponds to the increase of the output signal with magnetic field. A Slope of “1” is equivalent to an output signal increase of 100% at an increase of 2048 of the ADC value.

The HAL855 provides no ADC-Readout register. Therefore, the ADC value cannot be read out. Instead, a digital readout DOUT is provided. The digital readout is captured at the end of the signal processing.

The DOUT value can be used in the calibration process provided the register settings are programmed to known initial settings. Hence, the calibration process must start with a initialization of all relevant registers.

The calibration procedure is shown in Fig. 6–8. The register values are intended as an example:

Step 1:

Programming of the parameters which do not require individual adjustment:

– Filter Frequency, Magnetic Range, Output Format, Output Period, TC, and TCSQ.

Step 2:

Initialize the registers SLOPE and SHIFT:

– The SHIFT register should be programmed to 50%.

– As the ADC value depends on the low pass filter frequency, it is recommended to program the SLOPE register to the values listed in Table 6–7.

Table 6–7: Initial SLOPE values

3 dB Filter Frequency	$SLOPE_{INITIAL}$
80	0.2578
160	0.2578
500	0.1938
1000	0.1938
2000	0.3398

Step 3:

Get the digital readout value DOUT1 in the first calibration point and assign a nominal output voltage ($V_{OUT,1}$) to that point.

Step 4:

Get the digital readout value DOUT2 in the second calibration point and assign a nominal output voltage ($V_{OUT,2}$) to that point.

Step 5:

Calculate and program the new values of Shift and Slope.

– The SHIFT register is calculated as:

$$Shift = \frac{100\%}{4096} \cdot \left(\frac{3968 - ((DOUT2 - 2048) \cdot 3968)}{(DOUT2 - DOUT1)} \right)$$

$$SHIFT = \frac{Shift}{100\%} \cdot 1024$$

– The SLOPE register is calculated as:

$$Slope = \frac{Slope_{INITIAL}}{(DOUT2 - DOUT1)} \cdot 3968$$

$$SLOPE = Slope \cdot 2048$$

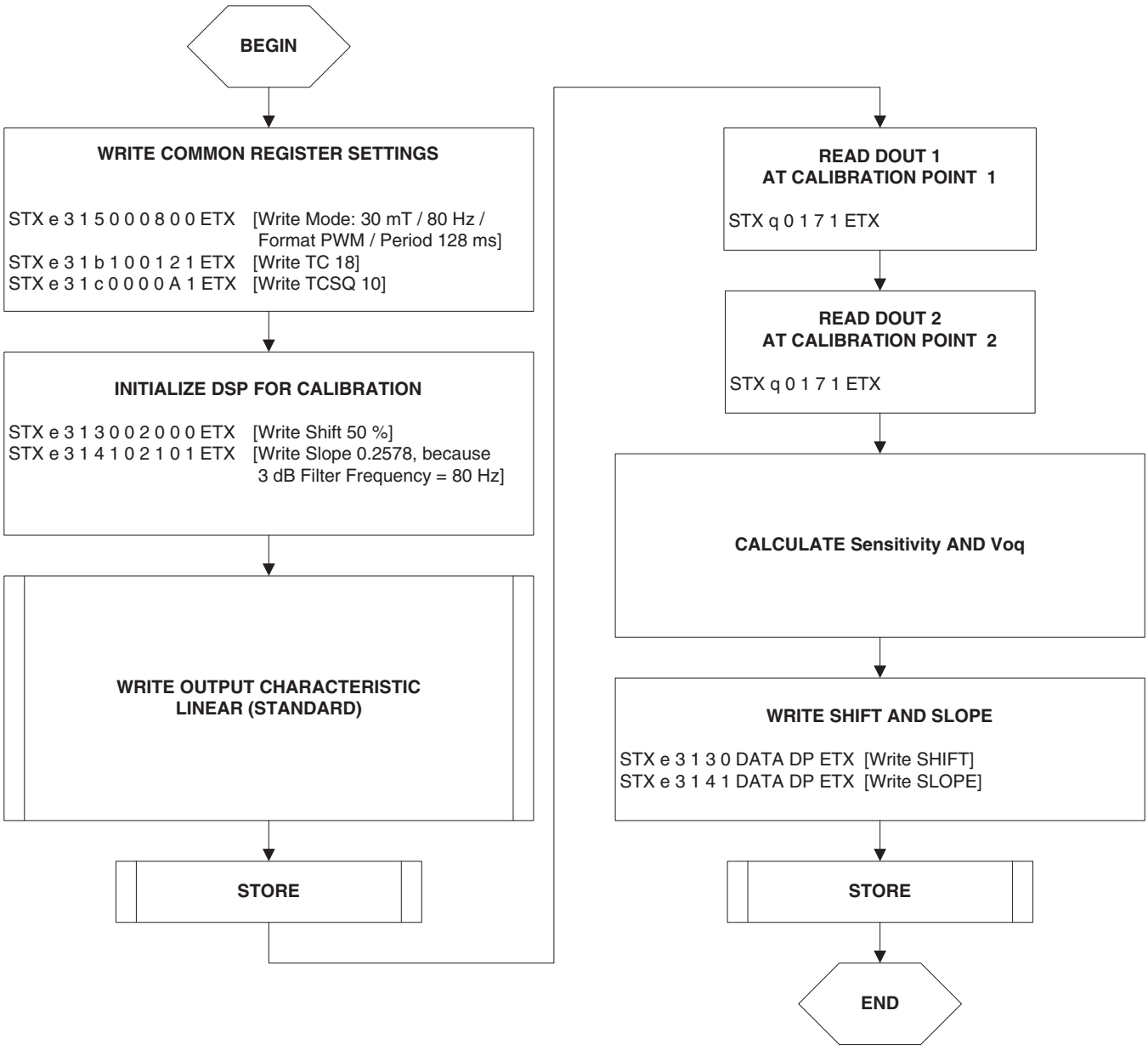

Fig. 6–8: Calibration procedure

Step 6:

The “Store” sequence is given in Fig. 6–9. The permanent storing of data into the EEPROM memory is a two-step process. First, all zero bits (ERASE), then all one bits (PROM) are written.

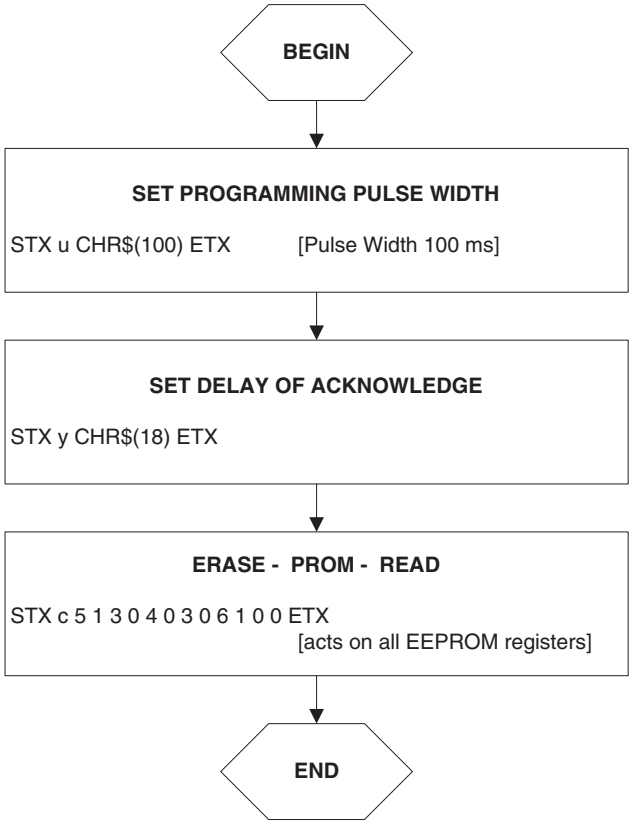

Fig. 6–9: Store sequence

7. Operation Mode 3

HAL 856

This operation mode is set by sending the “j3” command.

Note: The HAL 856 is a two-wire device. It is programmed by modulating the supply voltage, and it responds by modulating its current consumption. In order to detect the current levels of the sensor, the Hall Programmer Board V 5.1 measures the sensor’s supply current. Hence, only one HAL 856 can be connected to the board.

Communication: PC → prog. board

Section 7.1 and Section 7.2 contain a detailed description of communication between the PC and the programmer board V 5.1. Particularly, all board commands available in the operation mode “j3” are listed.

Communication: prog. board → sensor

The communication between the programmer board and the Hall sensor is described in Section 7.3

Data formats

The data formats for communication between both the PC and the board, and the board and the sensor are summarized in Section 7.4

Programming procedure

Finally, Section 7.5 depicts the programming procedure and gives an example of a complete programming sequence.

7.1. Definition of the Protocol

The general syntax is as follows (blanks are given for better readability only):

PC → Board:
STX BCMD [CMD CP ADR AP] [DAT3 DAT2 DAT1 DAT0 DP] ETX

Board → PC:
STX STATUS DAT3 DAT2 DAT1 DAT0 DP ETX

The characters in brackets [...] are optional, depending on the board command BCOM.

Table 7–1: Description of the characters

Character	Description
STX	ASCII character 2
ETX	ASCII character 3
BCMD	Board command
CMD	Command
CP	Command parity
ADR	Address
AP	Address parity
DAT3 DAT2 DAT1 DAT0	Data characters, each encoding 4 bit in HEX format. Example: 0 1 f 0 (HEX) = 496
DP	Data parity
STATUS	Status/error flag

7.2. The Board Commands BCMD

The board commands activate functions of the firmware. The characters following the BCMD are the parameters of this function. Some functions serve to control the board operation and do not address the Hall device.

The read, write, program, and lock functions communicate with the Hall device. The parameters of these functions are converted into a serial telegram and sent to the connected sensor.

Table 7–2: Description of the board commands

BCMD	Parameters	Description
n	–	Switch V_{DD} on
o	–	Switch V_{DD} off
z[t]	t = 30...255 (as ASCII character) default: t = 85 (HAL856)	Set bit time slow mode bit time = t × 0.02 ms default bit time = 1.7 ms (HAL856)
u[t]	t = 30...255 (as ASCII character) HAL856: set t = 100	Set programming pulse width width = t × 1.00 ms HAL856: set pulse width = 100 ms
y[t]	t = 10...255 HAL856: set t = 11	Delay Acknowledge width = t × 64.0 µs HAL856 set width = 0.7 ms.
e[p]	p = CMD CP ADR AP DAT3 DAT2 DAT1 DAT0 DP	Write HAL and echo data to PC
q[p]	p = CMD CP ADR AP	Read HAL and echo data to PC
c[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2 CMD3 CP3 ADR3 AP3	Program HAL and echo answer to PC
k[p]	p = CMD1 CP1 ADR1 AP1 CMD2 CP2 ADR2 AP2	Program LOCK bit and echo answer to PC
i		Read Biphase M sequence
m[p]	p = 0 ... 7 (as HEX)	Read PWM sequence p = PWM period (2^p in ms).
v	–	Request firmware version
t	–	Request status and echo data to PC
j[p]	p = 0...3 (as HEX)	Switch board operation mode 0 = EMU Board V 4.1 (default) 1 = HAL805, HAL815, HAL817, HAL810, HAL 82x, HAL83x, HAL 880, HAL1000 and HAL1002 2 = HAL855 3 = HAL856

Examples:

(Blanks are given for better readability only)

– set board operation mode:
STX j 3 ETX

– power on:
STX n ETX

– set bit time slow CHR$(85) = 'U':
STX z U ETX

– read register no. 2:
STX q 0 1 2 1 ETX

– write register no. 2:
STX e 3 1 2 1 0 0 0 A 1 ETX

– write output characteristic setpoint 0:
STX e 1 0 0 0 0 0 0 0 1 ETX

– write output characteristic setpoint 16:
STX e 2 0 0 0 0 1 0 0 0 ETX

– read output characteristic setpoint 15:
STX e 6 1 f 0 ETX

– read output characteristic setpoint 31:
STX q 7 0 f 0 ETX

– store sequence:
STX u CHR$(100) ETX (set programming time)
STX y CHR$(11) ETX (set delay of Acknowledge)
STX c 5 1 3 0 4 0 3 0 6 1 0 0 ETX
(ERASE-PROM-READ)

– lock sensors
STX e 3 1 6 0 0 0 0 1 0 ETX (set Lock bit)
STX u CHR$(100) ETX (set programming time)
STX y CHR$(11) ETX (set delay of Acknowledge)
STX k 4 0 6 0 0 1 3 0 ETX (PROM-READ)

7.2.1. Monitoring of the Supply Voltage

In case of ERASE, PROM, and LOCK, the supply voltage is measured by the board, and a data string is sent back to the PC. The relation between the data and the supply voltage is

$$V_{PROG} = \frac{DAT}{4095} \cdot 6 \cdot 2.485\ \mathrm{V}$$

If the supply voltage is out of the specification limits, the status bit is set to "1".

Examples:

– V_{PROG} = 4.80 V, Data board → PC:
STX 1 0 5 2 6 0 ETX
(DAT = 0 5 2 6)

– V_{PROG} = 5.00 V, Data board → PC:
STX 0 0 5 5 D 0 ETX
(DAT = 0 5 5 D)

7.3. Programming of the Sensor

7.3.1. Definition of Programming Pulses

The sensor is addressed by modulating a serial telegram on the supply voltage. The sensor answers with a serial telegram on the output pin.

The bits in the serial telegram have a different bit time for the V_{DD}-line and the output. The bit time for the V_{DD}-line is defined through the length of the Sync bit at the beginning of each telegram. The bit time for the output is defined through the Acknowledge bit.

A logical “0” is coded as no voltage change within the bit time. A logical “1” is coded as a voltage change between 50% and 80% of the bit time. After each bit, a voltage change occurs.

7.3.2. Definition of the Telegram

Each telegram starts with the Sync bit (logical 0), 3 bits for the Command (COM), the Command Parity bit (CP), 4 bits for the Address (ADR), and the Address Parity bit (AP).

There are 4 kinds of telegrams:

Write a register

After the AP bit, follow 14 Data Bits (DAT) and the Data Parity bit (DP). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge bit (logical 0) on the output (see Fig. 7–2).

Read a register

After evaluating this command, the sensor answers with the Acknowledge bit, 14 Data bits, and the Data Parity bit on the output (see Fig. 7–3).

Programming the EEPROM cells

In order to permanently store the written data into the EEPROM cells, an erase and program command have to be sent to the sensor. After the recognition of the erase and program commands, the HAL856 answers with an acknowledge pulse on its output signal. After the acknowledge pulse, a pulse on the V_{DD}-line is created to start the charging of the EEPROM cells. Then the supply voltage is kept constant during the charging time. To stop the charging, a further command is sent to the HAL856. This stopping command can be a further programming command or a read command (see Fig. 7–4).

Lock a sensor

To lock the EEPROM registers, the lock bit has to be programmed. Write the lock bit into the lock register (see Fig. 7–2). If the telegram is valid and the command has been processed, the sensor answers with an Acknowledge Bit (logical 0) on the output. In order to store the lock bit permanently, an erase and program command have to be sent to the sensor. This is the same procedure as mentioned above (see Programming the EEPROM cells and Fig. 7–4. The EEPROM registers are locked **after** a power on reset.

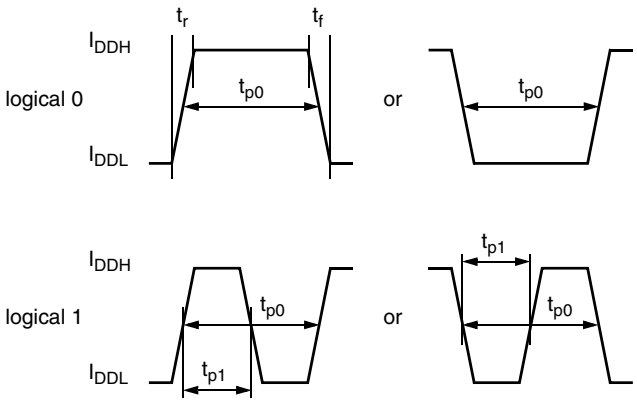

Fig. 7–1: Definition of logical 0 and 1 bit

Table 7–3: Telegram parameters

Symbol	Parameter	Pin	Min.	Typ.	Max.	Unit	Remarks
V_{DDL}	Supply voltage for low level during programming	1	5	5.6	6	V	
V_{DDH}	Supply voltage for high level during programming	1	6.8	8.0	8.5	V	
V_{DDPROG}	Supply voltage for permanent storing of the data	1	4.9	5.0	5.1	V	
t_r	Rise time	1			0.05	ms	
t_f	Fall time	1			0.05	ms	
t_{p0}	Bit time on V_{DD}	1	1.7	1.75	1.8	ms	t_{p0} is defined through the Sync bit
t_{pOUT}	Bit time on output pin	3	2	3	4	ms	t_{pOUT} is defined through the Acknowledge bit
t_{p1}	Voltage change for logical 1	1, 3	50	65	80	%	% of t_{p0} or t_{pOUT}
t_{PROG}	Programming time for EEPROM	1	95	100	105	ms	
t_{rp}	Rise time of charging pulse	1	0.2	0.5	1	ms	
t_{fp}	Fall time of charging pulse	1	0		1	ms	
t_w	Delay time of charging pulse after Acknowledge	1	0.5	0.7	1	ms	

WRITE

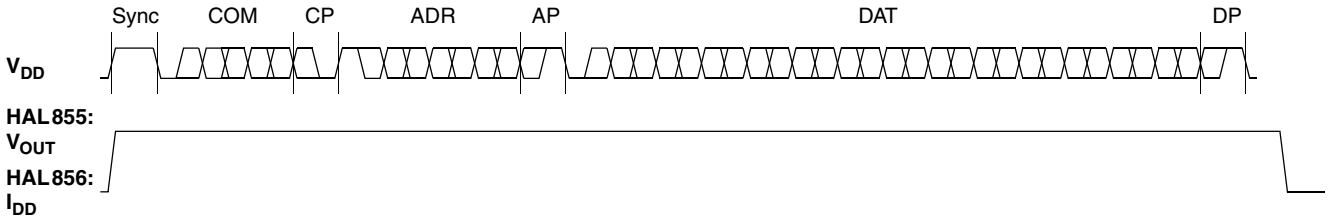

Fig. 7–2: Telegram for coding a Write command

READ

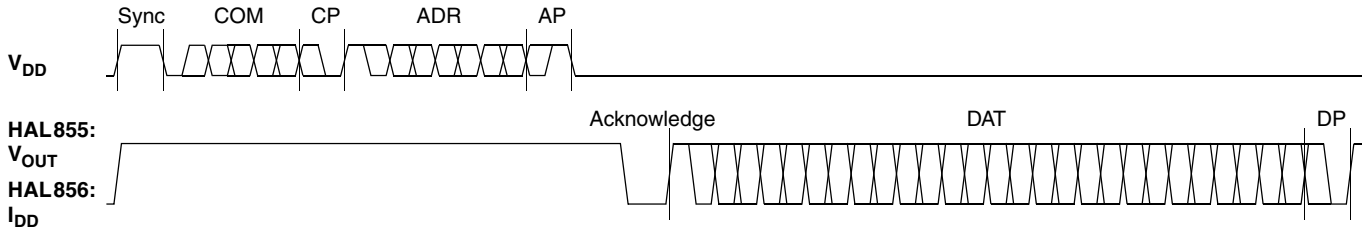

Fig. 7–3: Telegram for coding a Read command

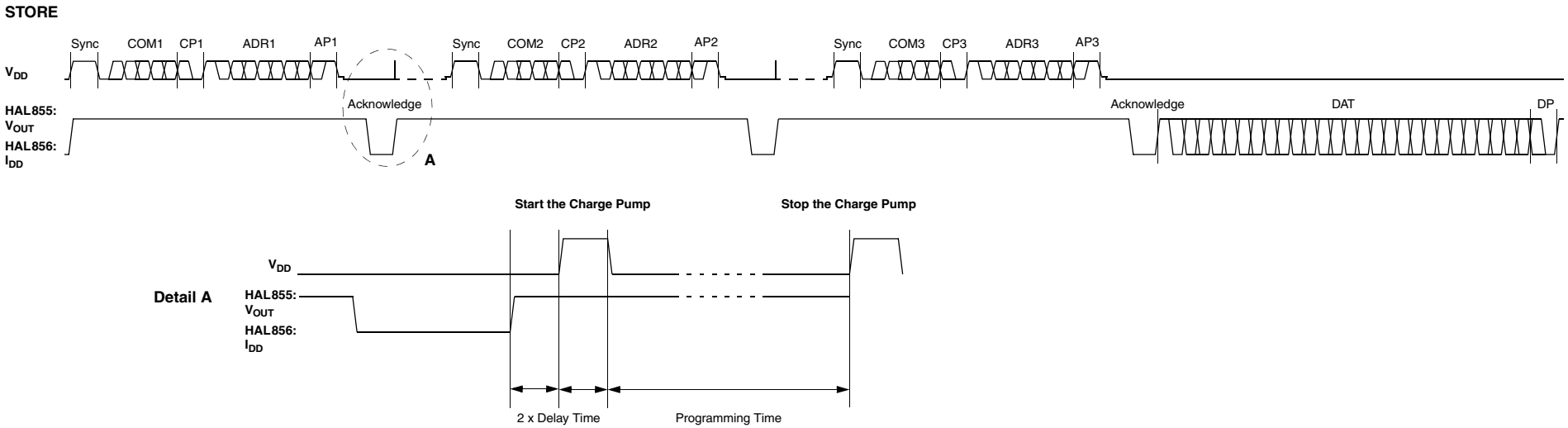

Fig. 7–4: Telegram for programming the EEPROM

7.3.3. Telegram Codes

Sync Bit

Each telegram starts with the Sync bit. This logical “0” pulse defines the exact timing for t_{p0}.

Command Bits (COM)

The Command code contains 3 bits and is a binary number. Table 7–4 shows the available commands and the corresponding codes for the HAL856.

Command Parity Bit (CP)

This parity bit is “1” if the number of zeros within the 3 command bits is uneven. The parity bit is “0”, if the number of zeros is even.

Address Bits (ADR)

The Address code contains 4 bits and is a binary number. Table 7–5 shows the available addresses for the HAL856 registers.

Address Parity Bit (AP)

This parity bit is “1” if the number of zeros within the 4 address bits is uneven. The parity bit is “0” if the number of zeros is even.

Data Bits (DAT)

The 14 Data bits contain the register information.

The registers use different number formats for the Data bits. These formats are explained in Section 7.3.4

In the Write command, the last bits are valid. If, for example, the TC register (6 bits) is written, only the last 6 bits are valid.

In the Read command, the first bits are valid. If, for example, the TC register (6 bits) is read, only the first 6 bits are valid.

Data Parity Bit (DP)

This parity bit is “1” if the number of zeros within the binary number is even. The parity bit is “0” if the number of zeros is uneven.

Acknowledge

After each telegram, the output answers with the Acknowledge signal. This logical “0” pulse defines the exact timing for t_{pOUT}.

Table 7–4: Available commands

Command	Code	Explanation
READ	0	read a Setup EEPROM register (like TC, TCSQ, magnetic range, etc.)
READL	6	read a Characteristics EEPROM register (setpoints 0 to 15)
READH	7	read a Characteristics EEPROM register (setpoints 16 to 31)
WRITE	3	write a Setup EEPROM register (like TC, TCSQ, Magnetic range, etc.)
WRITEL	1	write a Characteristics EEPROM register (setpoints 0 to 15)
WRITEH	2	write a Characteristics EEPROM register (setpoints 16 to 31)
PROM	4	program all non-volatile registers
ERASE	5	erase all non-volatile registers
Please note: The Lock bit is set by using the WRITE command followed by a PROM.		

7.3.4. Number Formats

Binary number: The most significant bit is given as first, the least significant bit as last digit.

Example:

101001 represents 41 decimal.

Signed binary number: The first digit represents the sign of the following binary number (1 for negative, 0 for positive sign).

Example:

0101001 represents +41 decimal
1101001 represents –41 decimal

Two's-complementary number: The first digit of positive numbers is "0", the rest of the number is a binary number. Negative numbers start with "1". In order to calculate the absolute value of the number, calculate the complement of the remaining digits and add "1".

Example:

0101001 represents +41 decimal
1010111 represents –41 decimal

Table 7–5: Available register addresses

Register	Code	Data Bits	Format	Customer	Remark
CURRENT-SOURCE	1	10	binary	read/write/program	Used to define output slew rate and output current levels (I_{DD_HIGH} and I_{DD_LOW})
PARTNUMBER	2	11	binary	read/write/program	Only with BiPhase-M mode
SHIFT	3	11	two's compl.	read/write/program	
SLOPE	4	14	signed binary	read/write/program	
MODE	5	14	binary	read/write/program	range, filter and output format settings
LOCK	6	1	binary	write/program	lock bit
DIGITAL READOUT	7	12	binary	read	digital value after signal processing
OFFSET CORRECTION	8	5	two's compl. (4 LSBs)	read/write/program	compensation of system offsets
SPECIALCUST.	9	6	binary	read/write/program	special customer register to define Biphase-M bittime and Partnumber Enable
TC	11	6	signed binary (6 LSBs)	read/write/program	linear temperature coefficient
TCSQ	12	5	binary	read/write/program	quadratic temperature coefficient
CURVE LOW	0...15	9	binary	write/read/program	setpoints 0 to 15
CURVE HIGH	0...15	9	binary	write/read/program	setpoints 16 to 31

7.3.5. Register Information

Terminology

REGISTER: Name of the register or register value

Register: Name of the parameter

PARTNUMBER

– The register range is from 0 up to 2047.

SHIFT

– The register range is from –1024 up to 1023.

– The register value is calculated by:

$$SHIFT = \frac{Shift}{100\%} \cdot 1024$$

SLOPE

– The register range is from –8192 up to 8191.

– The register value is calculated by:

$$SLOPE = Slope \cdot 2048$$

TC and TCSQ

– The TC register range is from –31 up to 31.

– The TCSQ register range is from 0 up to 31.

Please refer to the data sheet for the recommended values.

Note: The word length of the TC register is 7 bit. The six LSBs represent a signed binary number. The MSB has to be ignored.

MODE

– The register range is from 0 up to 16383 and contains the settings for PERIOD, FORMAT, FILTER and RANGE:

$$MODE = PERIOD \cdot 512 + FORMAT \cdot 64 + FILTER \cdot 8 + RANGE$$

Please refer to the data sheet for the available PERIOD, FORMAT, FILTER and RANGE values.

DIGITAL-READOUT

– This register is read only.

– The register range is from 0 up to 4095.

OFFSET CORRECTION

– The register range is from 0 to 31

– The MSB is set to activate the offset correction.

CURRENTSOURCE

– The register range is from 0 to 1023 and contains the settings for LOW CURRENT, HIGH CURRENT, and SLEW RATE:

$$CURRENTSOURCE = SLEW\ RATE \times 256 + LOW\ CURRENT \times 8 + HIGH\ CURRENT$$

SPECIAL CUSTOMER

– The register range is from 0 to 63 and contains the settings for OUTPUT BITTIME and PARTNUMBER ENABLE:

$$SPECIAL\ CUSTOMER = PARTNUMBER\ ENABLE \times 16 + OUTPUT\ BITTIME$$

7.3.6. Programming Information

If the content of any register is to be changed, the desired value must first be written into the corresponding RAM register. Before reading out the RAM register again, the register value must be permanently stored in the EEPROM.

Permanently storing a value in the EEPROM is done by first sending an ERASE command followed by sending a PROM command and a read command. The address within the ERASE and PROM commands is not important. ERASE and PROM act on all registers in parallel.

If all HAL856 registers are to be changed, all writing commands can be sent one after the other, followed by sending one ERASE and PROM command at the end.

During all communication sequences, the customer has to check if the communication with the sensor was successful. This means that the acknowledge and the parity bits sent by the sensor have to be checked by the customer. If the Micronas programmer board is used, the customer has to check the error flags sent from the programmer board. It is recommended to use the programmer board version 5.1.

Note: For production and qualification tests, it is mandatory to set the Lock bit after final adjustment and programming of HAL856. The Lock function is active after the next power-up of the sensor.
The success of the Lock process should be checked by reading at least one sensor register after locking and/or by an analog check of the sensor's output signal.
Electrostatic Discharges (ESD) may disturb the programming pulses. Please take precautions against ESD.

7.4. Data Formats

The protocol between the programmer board and the Hall sensor defines fixed lengths for the data, command, and address words to be transmitted:

– Command (CMD): 3 bit

– Address (ADR): 4 bit

– Data (DAT): 14 bit

– Command Parity (CP): 1 bit

– Address Parity (AP): 1 bit

– Data Parity (DP): 1 bit

Reminder: For communication with the programmer board, the 14 data bits are encoded in four ASCII characters: DAT3, DAT2, DAT1, and DAT0.

As every register has a different length, not all of the 14 bits are used. Furthermore, there is a difference between the write format (transfer board → IC) and the read format (transfer IC → Board). Table 7–6 shows which of the 14 bits are valid for each register (write format and read format).

Table 7–6: Data formats

Register	Char	DAT3				DAT2				DAT1				DAT0			
	Bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PART NUMBER	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SHIFT	Write	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	–	–	–
SLOPE	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
MODE	Write	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
DIGITAL OUTPUT	Read	–	–	V	V	V	V	V	V	V	V	V	V	V	V	V	V
SPECIAL CUSTOMER	Write	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	–	–	–	–	–	–	–	–
TC	Write	–	–	–	–	–	–	–	–	–	(V)	V	V	V	V	V	V
	Read	–	–	(V)	V	V	V	V	V	V	–	–	–	–	–	–	–
TCSQ	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
CURVE LOW	Write	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	–	–	–	–	–
CURVE HIGH	Write	–	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	–	–	–	–	–
OFFSET CORRECTION	Write	–	–	–	–	–	–	–	–	–	–	–	V	V	V	V	V
	Read	–	–	V	V	V	V	V	–	–	–	–	–	–	–	–	–
CURRENT SOURCE	Write	–	–	–	–	–	–	V	V	V	V	V	V	V	V	V	V
	Read	–	–	V	V	V	V	V	V	V	V	V	V	–	–	–	–

V: valid, –: ignore, bit order: MSB first, (V): This bit must be ignored.

7.5. The Programming Procedure

The general programming procedure is described in detail for the HAL856.

An example for a programming procedure is given in Fig. 7–5.

First, the programmer board has to be initialized. Then, the sensor can be calibrated. Please keep in mind that only one HAL856 can be connected to the board!

Example:

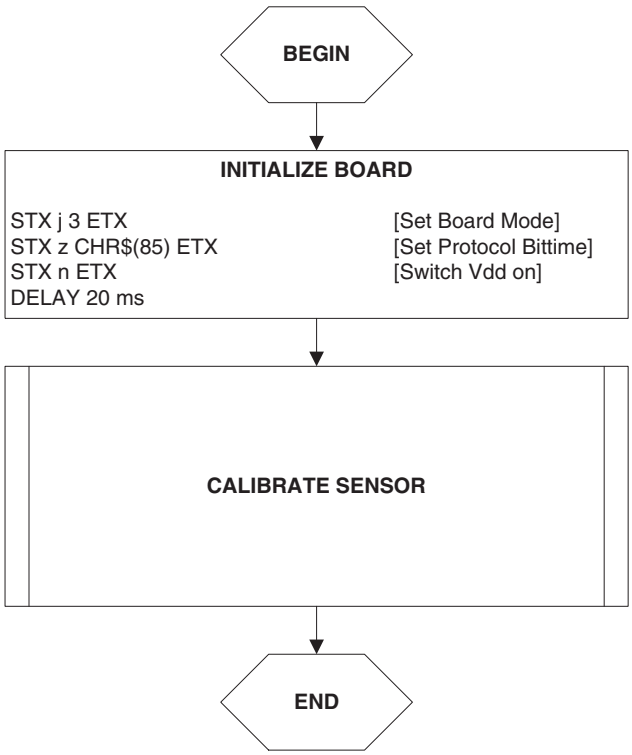

Fig. 7–5: General procedure

7.5.1. The Programmable Output Characteristic

The HAL856 allows programming of a user-defined output characteristic by 32 setpoints.

Each setpoint corresponds to a defined output signal between 0% and 99.8%

The sequence of setpoints must be defined monotonously.

An example of how the setpoints are programmed is given in Fig. 7–6.

Note: In case of a non-linear output characteristic, the parameters SHIFT and SLOPE are related to the (virtual) straight line connecting the first and the last setpoints (setpoint 0 and setpoint 31, respectively).

Example:

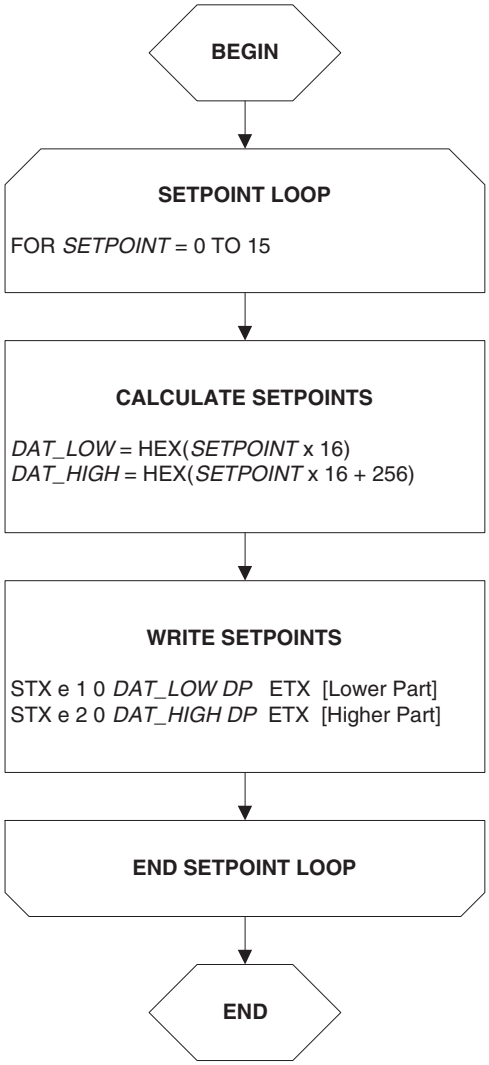

Fig. 7–6: Definition of the Output Characteristic “Linear (Standard)”

7.5.2. The Calibration Procedure

The output characteristic is determined by the following parameters:

– Shift corresponds to the output voltage at an ADC value of zero.

– Slope corresponds to the increase of the output signal with magnetic field. A Slope of “1” is equivalent to a output signal increase of 100% at an increase of 2048 of the ADC value.

The HAL856 provides no ADC-Readout register. Therefore, the ADC value cannot be read out. Instead, a digital readout DOUT is provided. The digital readout is captured at the end of the signal processing.

The DOUT value can be used in the calibration process provided the register settings are programmed to known initial settings. Hence, the calibration process must start with a initialization of all relevant registers.

The calibration procedure is shown in Fig. 7–7. The register values are intended as example:

Step 1:

Programming of the parameters which do not require individual adjustment:

– Filter Frequency, Magnetic Range, Output Format, Output Period, TC, and TCSQ.

Step 2:

Initialize the registers SLOPE and SHIFT:

– The SHIFT register should be programmed to 50%.

– As the ADC value depends on the low pass filter frequency, it is recommended to program the SLOPE register to the values listed in Table 7–7.

Table 7–7: Initial SLOPE values

3 dB Filter Frequency	$SLOPE_{INITIAL}$
80	0.2578
160	0.2578
500	0.1938
1000	0.1938
2000	0.3398

Step 3:

Get the digital readout value DOUT1 in the first calibration point and assign a nominal output voltage ($V_{OUT,1}$) to that point.

Step 4:

Get the digital readout value DOUT2 in the second calibration point and assign a nominal output voltage ($V_{OUT,2}$) to that point.

Step 5:

Calculate and program the new values of Shift and Slope.

– The SHIFT register is calculated as:

$$Shift = \frac{100\%}{4096} \cdot \left(\frac{3968 - ((DOUT2 - 2048) \cdot 3968)}{(DOUT2 - DOUT1)} \right)$$

$$SHIFT = \frac{Shift}{100\%} \cdot 1024$$

– The SLOPE register is calculated as:

$$Slope = \frac{Slope_{INITIAL}}{(DOUT2 - DOUT1)} \cdot 3968$$

$$SLOPE = Slope \cdot 2048$$

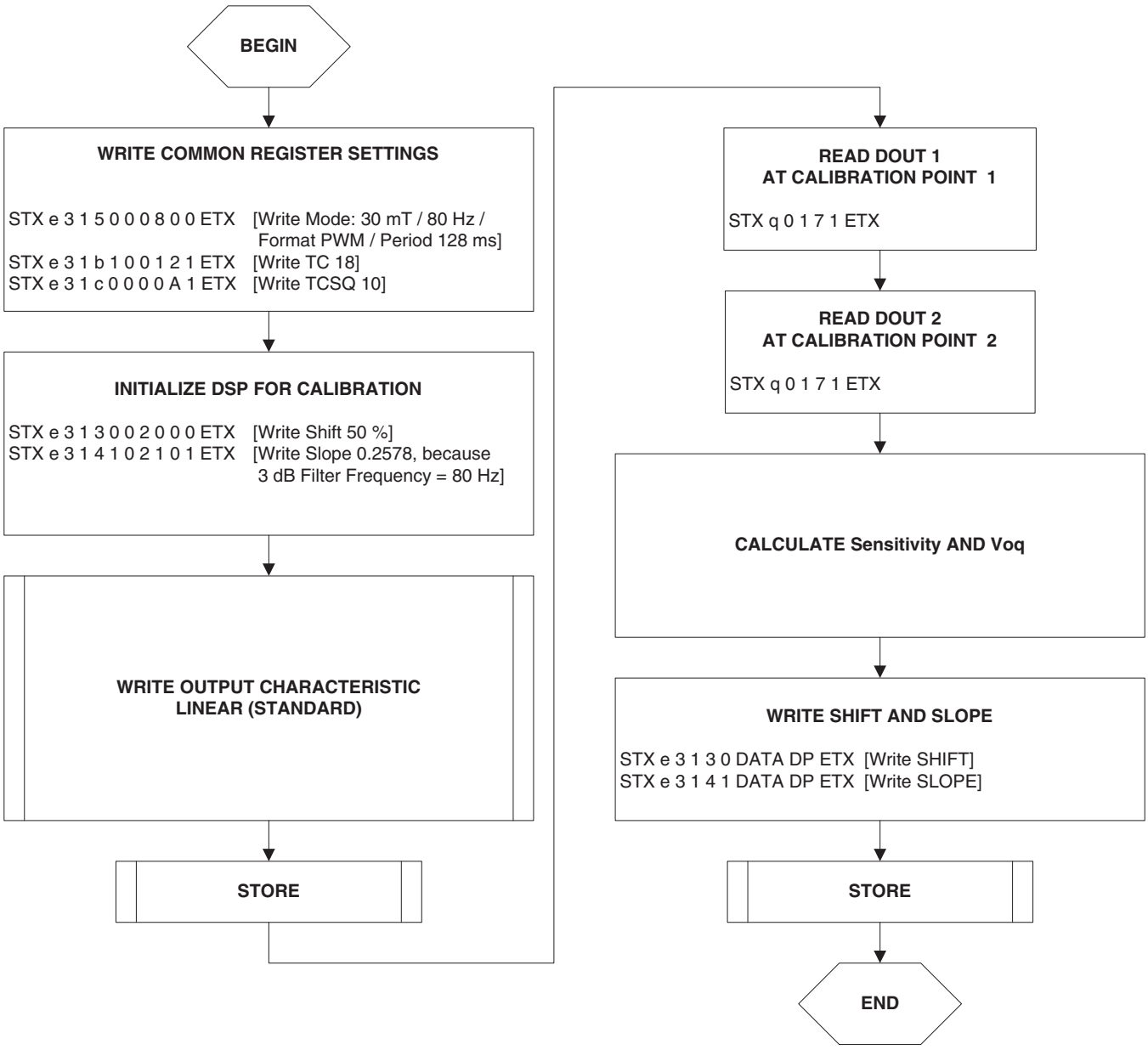

Fig. 7–7: Calibration procedure

Step 6:

The “Store” sequence is given in Fig. 7–8. The permanent storing of data into the EEPROM memory is a two-step process. First, all zero bits (ERASE), then all one bits (PROM) are written.

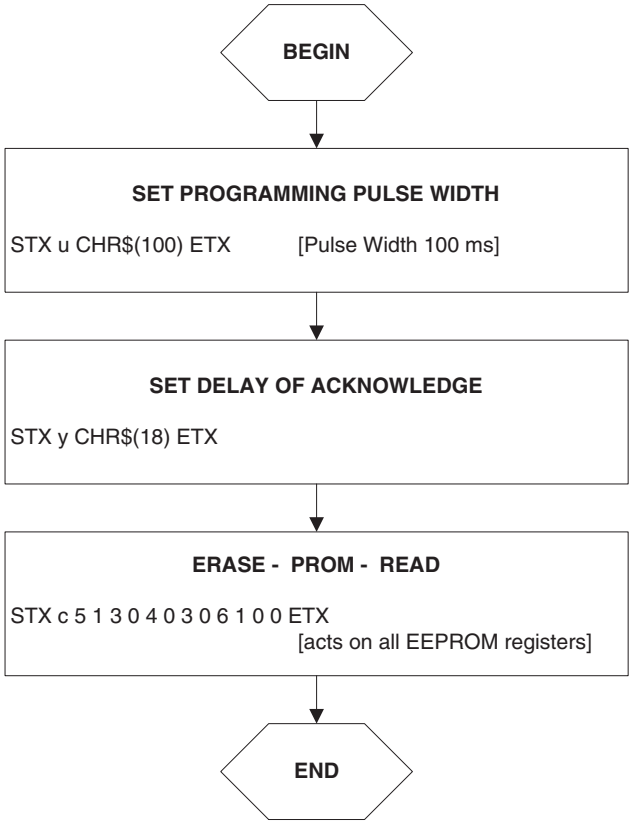

Fig. 7–8: Store sequence

8. Application Note History

1. Application Note Kits/Boards: “HAL8xx, HAL1000, HAL1500 Programmer Board”, Aug. 12, 2004, 6251-632-1-1AK. First release of the application note kits/boards.
2. HAL8xy, HAL100x Programmer Board, May 14, 2009; APN000046_001EN. First release of the application note. Major changes:
   - Specification for HAL1500 removed
   - Specification for HAL824, HAL825, and HAL880 added
   - Section 3.4.1. Firmware Update added
   - Section 4.3.6. Programming Information updated
   - Section 5.3.7. Programming Information updated
   - Section 6.3. Programming of the Sensor updated
   - Section 6.4. Data Formats updated
   - Section 7.3. Programming of the Sensor updated
   - Section 7.4. Data Formats updated
   - Section 8. Operation Mode 5 added
3. HAL8xy, HAL100x Programmer Board, April 21, 2010; APN000046_02EN. Second release of the application note.
   Major changes: Section 8. Operation Mode 5 removed
4. HAL8xy, HAL100x Programmer Board, Sept. 19, 2013; APN000046_03EN. Third release of the application note.
   Major changes:
   - Sensor types HAL817, HAL83x and HAL1002 added

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