Technical Description
- OEM board for pulse wave analysis -

Hardware-Version: **1.0**

Firmware-Version: **1.0**

Issued:	T. Fischer	Reviewed:	A. Gürbüz	Approved:	L. Engel
Date:	Mar-21-2018	Date:	Mar-21-2018	Date:	Mar-21-2018
Signature:		Signature:		Signature:	

Table of contents

Revision History

Doc.-Rev.	Date	Author	Comment
A	March 21st, 2018	T. Fischer	Creation

1 General information

1.1 Intended use

The OEM board for pulse wave analysis (PWA module) is intended for the non-invasive estimation of central blood pressure (systolic and diastolic value) and vascular conditions (augmentation pressure, augmentation index, and pulse wave velocity) in combination with a blood pressure measurement device that determines the blood pressure on the upper arm and fulfils the accuracy requirements of the ISO 81060-2.

The PWA module is an electronic board developed for health care medicine and designed to be integrated into a host system that have to be a medical device. In this case the PWA module is a subsystem of the superordinate medical device and is connected to power supply, control lines, serial interface and the analog output of a pressure sensor. The control of the board is done by commands via the serial interface. The results of the measurement and other information are transmitted via this serial interface.

The PWA module has to be built in the host system that has to fulfil the general requirements for basic safety and essential performance of medical electrical equipment (IEC 60601-1) and the particular requirements for the basic safety and essential performance of automated non-invasive sphygmomanometers (IEC 80601-2-30).

The intended patient populations are adults and children with an age of at least 12 years. The PWA module is **not** intended to be used on neonates. The arm circumference of the patient has to be in the range of the employed cuffs.

The medical device with the integrated PWA module has to be intended for use following the consultation and instruction by a physician (family doctor, specialist or hospital). It can be used, if the physical condition of the patient allows an automatic, non-invasive blood pressure measurement. The PWA module is **not** intended to be used in intensive care medicine or for alarming of life-threatening conditions. Contraindications for the usage of the PWA module are arrhythmias, femoral artery stenosis, and pregnancy.

A measurement with PWA module can be combined with other measurements and medical examinations at the patient that do not affect the blood flow in the upper arm during the pulse wave analysis.

1.2 Module features

Measurement:

- Non-invasive PWA on mean arterial pressure level in the blood pressure cuff
- Recording and storage of an analog pressure signal derived from the upper arm
- Preprocessing with artifact elimination of the analog pressure signal
- Reconstruction of the central pulse contour
- Estimation of central blood pressure (systolic, and diastolic value)
- Decomposition of the central pulse wave into antegrade and retrograde pulse wave
- Estimation of vascular conditions (augmentation pressure, augmentation index, pulse wave velocity, and arterial age)
- Support of spot and long-term pulse wave analysis up to 24 hours

Technical:

- Multiple communication interfaces like TWI and UART
- Optional Bluetooth communication (Bluetooth 2.1 + EDR)
- Storage capacity for up to 100 measurements
- Bootloader for system updates in the field
- Operating voltage of 3.3 V with an maximal operating current of 60.0 mA (without Bluetooth) and 160.0 mA (with active Bluetooth)
- Low power consumption of under 8.0 mA during sleep mode

1.3 Standards and guidelines

The PWA module is an accessory to a medical device in form of an electronic board and is designed, manufactured, tested, and distributed like a medical device. It is a subsystem, which has to be built in a host system. Therefore, it is only possible for the module to fulfil the relevant and applicable requirements of the following standards and guidelines:

- Design, manufacture, final inspection and distribution according ISO 13485
- Risk management according ISO 14971
- ARTERY guideline: Validation of non-invasive central blood pressure devices
- ARTERY guideline: validation of non-invasive haemodynamic measurement devices: Part 1, arterial pulse wave velocity

The host system has to fulfill further standards to become a medical device.

2 Measurement method

2.1 Measurement procedure

The inflation of an external blood pressure cuff up to a defined pressure near the mean arterial pressure level of the patient is the basis of the measurement procedure. The module is a slave device that is controlled by the host system. The pressure control for the cuff and the time management has to be done by the host system. The PWA module merely controls the timing of his internal processes.

The host system has to provide the analog pressure signal without any preprocessing (filtering and amplification) to the PWA pin of the module (see chapter 4.5). The complete analog preprocessing is done by the module. The pressure level for the record of the pulse waves should be constant and the leakage should be less than 3 mmHg/min during the measurement procedure.

The PWA module records the pulse waves over a period of 15 seconds after a received start command. The recorded pulses are provided online for the host system and are stored internally on the PWA module for a later evaluation (see Figure 1).

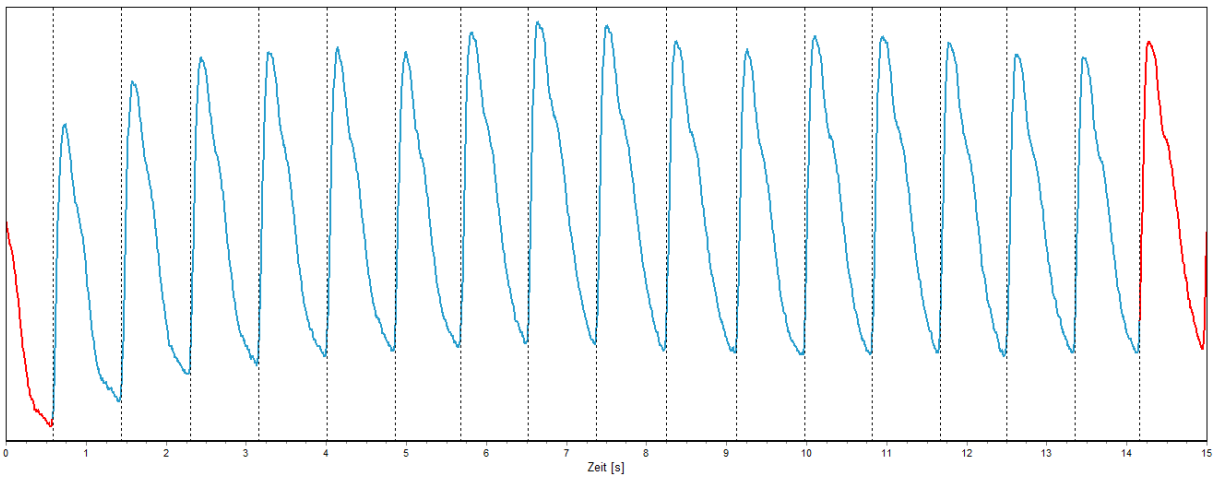


Figure 1: Digitalized analog pressure signal provided by the PWA module

The raw data will be preprocessed by the module after the record phase. Therefore the continuous signal is divided into single pulses and an artifact rejection is done to avoid inclusion of distorted pulses into the pulse wave analysis. All valid pulses will be timely scaled and averaged to one peripheral pulse wave. This peripheral pulse wave is stored (see Figure 2).

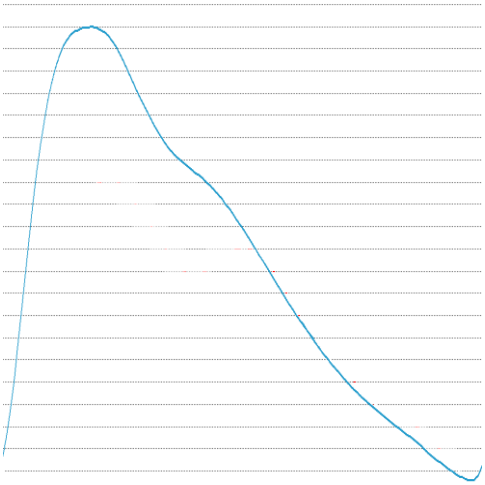


Figure 2: Preprocessed peripheral pulse wave

The recorded and preprocessed data that was stored on the module have to be transmitted to an analysis program like PhysioPortWin to reconstruct the central pulse wave by using a general transfer function and calculate the parameters for the assessment of arterial stiffness.

⚠ **Note:** Coming firmware versions will perform the complete pulse wave analysis and only need software support for the visualization of the results.

2.2 Measurement modes

The PWA module supports two types of measurement. Basically the module performs always a single or spot measurement, but it is possible to perform several sport measurements consecutively.

1. Manual mode (spot measurement):
   The user decides when he would like to trigger the measurement by sending a start command. The module records and stores the signal that was preprocessed and digitized by him. Additionally the module provides the current signal trace to the host module during the measurement to display the graph during the measurement.
2. Cycle mode (long-term measurement):
   The user can start a series of measurements by repeatedly sending start commands. After a completed measurement the next measurement can be started.

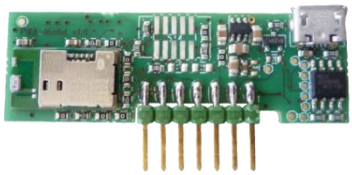
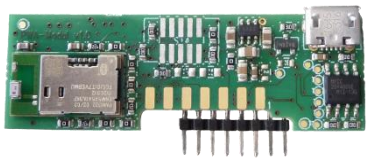
⚠ **Note:** If a running measurement is not finished yet, further start commands are ignored.

⚠ **Note:** The module stores every measurement. It is recommended to delete the storage of the module, before the patient is changed. That avoids the danger of data confusion.

3 Technical specifications

3.1 Mechanical specification

Table 1: Dimensions and connectors

	Mounted PCB with UART connector (ST1)	Mounted PCB with TWI connector (ST2)
Picture		
Height	6.40 mm	6.40 mm
Width	50.00 mm	50.00 mm
Depth (with pin header)	16.75 mm (23.50 mm)	16.75 mm (20.25 mm)
Weight	< 5.0 g	< 5.0 g
Connector	single row pin header (7 pins with 2.54 mm pitch)	single row pin header (8 pins with 2.00 mm pitch)

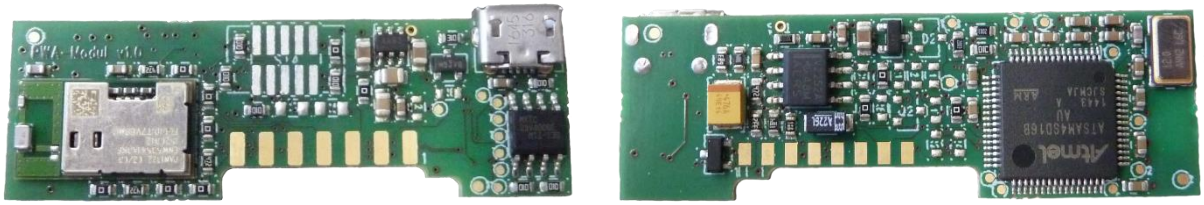


Figure 3: PWA module without pin header (bottom layer - left side / top layer - right side)

3.2 Ambient specification

Table 2: Operation conditions

Temperature	+5 and 40°C
Relative humidity	15 and 93 %, no condensation
Atmospheric pressure	700 and 1,060 hPa

Table 3: Transport and storage conditions

Temperature	-25 and 70°C
Relative humidity	10 and 93 %, no condensation
Atmospheric pressure	500 and 1,060 hPa

3.3 Electrical specification

Table 4: Signal description and electrical characteristics

Pin	Parameter	Min	Max	Unit
Supply voltage				
+3V3	Supply voltage	2.90	3.60	V
	Power Consumption (active)	60.0	160.0	mA
	Power Consumption (sleep)	4.8	8.0	mA
Ground potential				
GND	Ground	-	0.00	V
Power on (low active)				
$\overline{\text{PWR\_ON}}$	Low level	-	2.30	V
	High level	3.00	3.60	V
Reset (low active input)				
$\overline{\text{RST}}$	Reset pulse duration	5.0	-	µs
	Low level	-0.30	0.80	V
	High level	2.00	3.60	V
Handshake (output)				
SLAVE_TRIGGER	Low level	-	0.15	V
	High level	3.15	-	V
Two wire interface (TWI / I²C)				
SDA	Low level	-0.30	1.00	V
	High level	2.30	3.60	V
SCL	TWI clock frequency	0	400	kHz
Universal asynchronous receiver/transmitter (UART)				
URXD1	Low level	-0.30	0.80	V
	High level	2.00	3.60	V
UTXD1	Low level	-	0.15	V
	High level	3.15	-	V
Analog pressure signal with DC offset compensation (input)				
PWA	Signal AC range	-	76.00	mV
	ADC resolution	10	12	Bit
	ADC sample frequency	50	160	Hz

Table 5: Communication interfaces

UART	– Configuration:	115,200 bits per second / 8N1
TWI	– Module mode:	Slave mode
	– Address:	89 (0x59)
Bluetooth	– Type:	Bluetooth 2.1 + EDR
	– Class:	B
	– Profile:	SPP
	– Configuration:	115,200 bits per second / 8N1

3.4 Accuracy specification

The following chapter presents the results of the device validation regarding the measurement accuracy of the central blood pressure (cSYS/cDIA) and the pulse wave velocity (PWV).

Table 6: Measurement range of central blood pressure and pulse wave velocity

Parameter	Measurement range	Required mean deviation according the ARTERY guidelines	Required standard deviation according the ARTERY guidelines
cSYS	80 - 200 mmHg	≤ ± 5 mmHg	≤ 8 mmHg
cDIA	50 - 120 mmHg	≤ ± 5 mmHg	≤ 8 mmHg
PWV	4.5 - 16.0 m/s	≤ ± 1.0 m/s	≤ 1.5 m/s
HR	45 - 120 bpm	-	-

The accuracy determination was conducted with the PWA module integrated into the host system PHYSIO-PORT AS from PAR Medizintechnik GmbH & Co. KG (Germany). Central blood pressure and pulse wave velocity was non-invasive compared against SphygmoCor from AtCor Medical, Inc. (USA). The measurement accuracy is presented in Table 7.

Table 7: Accuracy of central blood pressure and pulse wave velocity

Parameter	Mean deviation	Standard deviation
cSYS	3.19 mmHg	2.8 mmHg
cDIA	0.6 mmHg	0.5 mmHg
PWV	0.9 m/s	1.4 m/s

4 Hardware interface

The system integrator is responsible for the correct installation of the PWA module into the host system. It should be performed by trained staff of the system integrator. The host system needs space and a mounting option for the module. Furthermore the host system has to provide a proper voltage and current supply (see Chapter 3 "Technical specifications"). For the control of the module the host needs an UART or TWI interface. After installation of PWA module the system integrator is responsible for the basic safety according to the standard EN 60601-1 and the EMC compatibility according to the standard EN 60601-1-2. The power supply of the host system should be a medical power supply.

4.1 UART communication

The first communication interface to the board is done via universal asynchronous receiver/transmitter (UART) with a baud rate of 115,200 bits per second. The interface lines operate with low voltage TTL (0.0 and 3.3 volts). A bidirectional connection is necessary, because the module receives data from and sends data to the host system.

⚠ **Note:** The system integrator have to decide between the two provided communication interfaces UART and TWI, because only one of them can be used at the same time.

4.2 TWI communication

The second communication interface to the board is done via two wire interface (TWI) with a maximum clock frequency of 400 kHz. The PWA module is configured as slave device and the complete communication control is done by the master. The interface lines operate with low voltage TTL (0.0 and 3.3 volts). An additional signal line (SLVAE-TRIGGER) is necessary, because the module has to indicate different states like "data ready" or "action complete" to the host system. See Chapter 6 ("Timing diagrams for communication") for detailed information.

⚠ **Note:** The system integrator have to decide between the two provided communication interfaces UART and TWI, because only one of them can be used at the same time.

4.3 Bluetooth

A third communication interface to the board is done via Bluetooth 2.1 + EDR with a baud rate of 115,200 bits per second. The used communication protocol is the serial port profile (SPP). You will find the module with the name consisting of PWA module and serial number e.g. “PWA module 12345”.

⚠ **Note:** This feature is not available for firmware version 1.0, but already prepared in hardware for later product versions.

4.4 Bootloader

The module contains a bootloader for update purposes in the field. The connector for the bootloader is a micro USB connector. You need for an update the new version of the firmware, the software BootloaderGUI, and a bootloader communication box for field updates. All firmware updates will be performed by PAR Medizintechnik GmbH & Co. KG.

⚠ **Note:** The module does not contain a low voltage TTL converter. Do not connect the micro USB connector directly to your PC to avoid damage to the module. You have to use the bootloader communication box.

4.5 Connector configuration

Table 8: Pin interface of the two closeable pin header

Pin number	UART connector (ST1) 1 2 3 4 5 6 7	TWI connector (ST2) 1 2 3 4 5 6 7 8
1	+3V3	+3V3
2	GND	GND
3	URXD1	PWR_ON
4	UTXD1	SLAVE_TRIGGER
5	RST	SDA
6	PWA	SCL
7	PWR_ON	PWA
8		RST

⚠ **Note:** The pins from ST1 are numbered from the right to the left side, if you look from the top layer. The pins from ST2 are numbered from the left to the right side, if you look from the top layer (see Chapter 3.1 Figure 3).

5 Software interface

The following description is valid for firmware version 1.0 and higher. In this chapter is the standard communication protocol with the PWA module described.

5.1 General conventions

All commands and messages from host system to PWA module and vice versa begin with the "Start of Text" character (ASCII: 0x02) and end with the "End of Text" character (ASCII: 0x03). In this document the designation for "Start of Text" is STX and "End of Text" ETX.

Different content of a message is separated by a semicolon (ASCII: 0x3B). In this document the designation for a semicolon is ";".

⚠ **Note:** The answers from the module are not always embraced with STX and ETX.

5.2 Command list

The commands do not have the same structure and length. The signs of the commands are combination of ASCII characters and integer values, because of performance purposes. The module checks the structure and length of all received commands. The module ignores unknown commands. The function and structure of all commands are described in Table 9.

Table 9: List of commands for the control of the module

Command	Length	Function
0x58 ("X") **or** 0x78 ("x")	1 Byte	The receipt of a small or capital x character aborts the current action.
STX TS ; BP ; PS ETX	35 Byte	The receipt of timestamp (TS) blood pressure (BP), and patient size (PS) unequal zero starts a measurement.[1]
STX 0x44 0x50 ETX	4 Byte	The receipt of "D" and "P" (**d**evice **p**rogramming) initiates the storage erase.
STX 0x52 0x4F ETX	4 Byte	The receipt of "R" and "O" (**r**ead **o**ut) initiates the read out of all stored data.
STX 0x47 0x56 ETX	4 Byte	The receipt of "G" and "V" (**g**et **v**ersion) initiates the read out of the firmware version.
STX 0x47 0x53 ETX	4 Byte	The receipt of "G" and "S" (**g**et **s**tatus) initiates the read out of the status or error code.

[1] The following explanation describes the expected structure of timestamp (see

Table 10), blood pressure results (see Table 11), and patient size (see Table 12) for the start of a measurement.

⚠ **Note:** All signs are ASCII characters.

Table 10: Structure of the timestamp

Command part	Length	Example	Meaning
Seconds	2 Byte	0x35 0x36	12 o'clock 34 minutes and 56 seconds
Minutes	2 Byte	0x33 0x34	
Hours	2 Byte	0x31 0x32	
Leap year	1 Byte	0xFF	Unused
Day	2 Byte	0x31 0x32	April 12th, 2018
Month	2 Byte	0x30 0x34	
Year	2 Byte	0x31 0x38	

Table 11: Structure of the blood pressure result

Command part	Length	Example	Meaning
Systolic pressure	3 Byte	0x31 0x32 0x30	120 mmHg
Semicolon	1 Byte	;	Separator
Diastolic pressure	3 Byte	0x30 0x38 0x30	80 mmHg
Semicolon	1 Byte	;	Separator
Mean arterial pressure	3 Byte	0x30 0x39 0x33	93 mmHg
Semicolon	1 Byte	;	Separator
Heart rate	3 Byte	0x30 0x36 0x33	63 Bpm

Table 12: Structure of the patient size

Command part	Length	Example	Meaning
Patient size	3 Byte	0x31 0x37 0x38	178 cm

⚠ **Note:** The PWA module ignores all commands during the execution of an action. The sole exception is the abort command.

6 Timing diagrams for communication

The Figure 4 explains how to read the timing diagrams. The thin dotted lines are the TTL voltage levels. The name of the signal line is shown on the left side of each timing diagram and in brackets is the test point given, if this signal line is not a pin of the hardware interface. Bold lines are the signal level these can be high, undefined or low. Dashed lines denote a time skip.

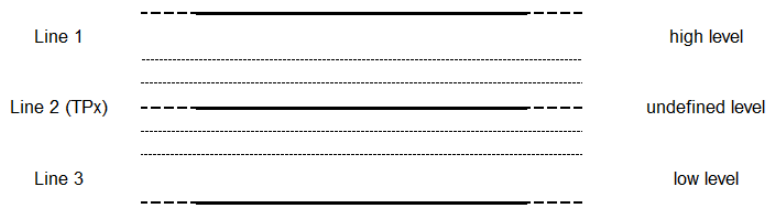


Figure 4: Legend of the timing diagrams

Startup and power down

The lines PWR_ON, RST, and +3V3 have to be used to activate the module. The SLAVE-TRIGGER line is in an undefined state in front of the activation, if you are using the TWI interface. The lines RST and +3V3 have to be on high level. If you pull afterwards the PWR_ON line to low level, the switchable power supply (+3V3A) will rise to high level and the module will start. You can check this with test point 2 on the PWA module.

If you are using the TWI interface, the startup procedure is shown by a high level of the SLAVE_TRIGGER. If the module is initialized, the SLAVE_TRIGGER line will switch to low level. If you are using UART interface, the initialized module will send you a status message consisting of four bytes (see Chapter 7). Afterwards the module is ready to process host commands.

You have to pull the PWR_ON line to a high level to power down the module. In this case the switchable power supply (+3V3A) will fall to low level and the module will be powered down. The SLAVE_TRIGGER will get back into an undefined state (if you are using the TWI interface).

The described startup and power down procedure is visualized by Figure 5 for the TWI interface and in Figure 6 for the UART interface.

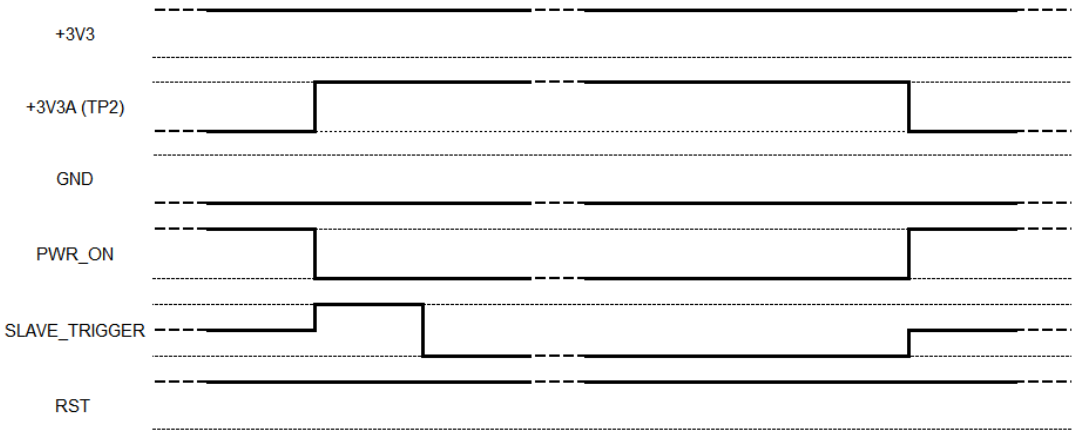


Figure 5: Timing diagram for the startup and power down of the module (TWI interface)

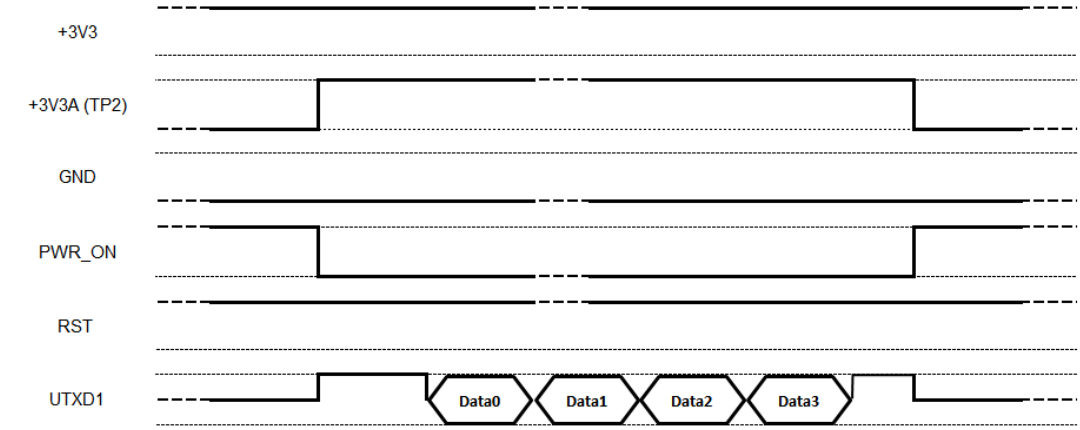


Figure 6: Timing diagram for the startup and power down of the module (UART interface)

⚠ **Note:** It is possible to power down the PWA module after each action to reduce the power consumption and safe a maximum of the battery life. All measurements will be stored on the flash storage of the module.

Reset

You have to pull the RST line to a low level to initiate a reset. All components of the module will restart, if you release the RST line back to high level. During a reset the SLAVE-TRIGGER is in an undefined state, if you are using the TWI interface. The described reset procedure is visualized by Figure 7.

The timing diagram without the SLAVE-TRIGGER line is valid for the UART interface, too.

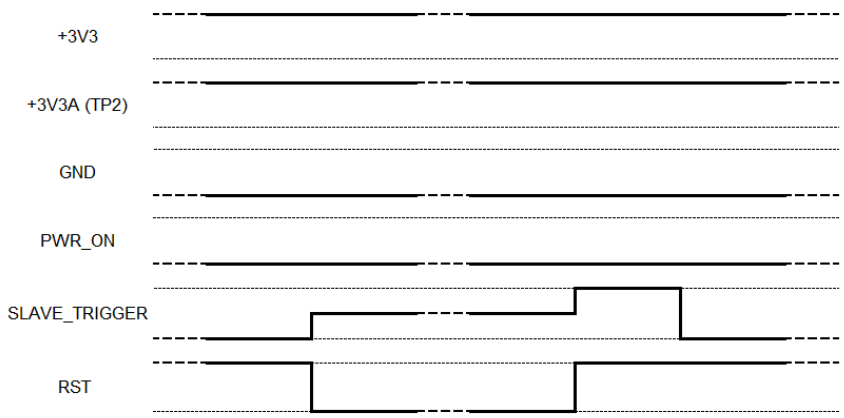


Figure 7: Timing diagram for the reset procedure

⚠ **Note:** You can reset the module at any time. If you reset the module during actions, data corruption is possible.

Start measurement

The precondition for the start of a measurement is an initialized module. That condition is shown by a low level on the SLAVE_TRIGGER line or the respective status message (see Chapter 7). The host system has to send timestamp, blood pressure values, and patient size in a frame of 35 Bytes to the module. If the structure corresponds to the content of Table 10, Table 11, and Table 12, the module will start the record of pulse wave data for 15 seconds. During these 15 seconds the module stores data into the internal flash for a later evaluation and sends data to the host module for a data display during the measurement. A transition from low to high level on the SLAVE_TRIGGER line indicates data that can be requested via TWI by the host system. If you are using the UART interface, the module will send the data automatically. The default data rate of the ADC is 160 Hz (2.400 values in 15 seconds). Each value consists out of two bytes that can be transmitted to the host system. The described measurement procedure is visualized by Figure 8 for the TWI interface. Figure 9 is valid for the UART interface.

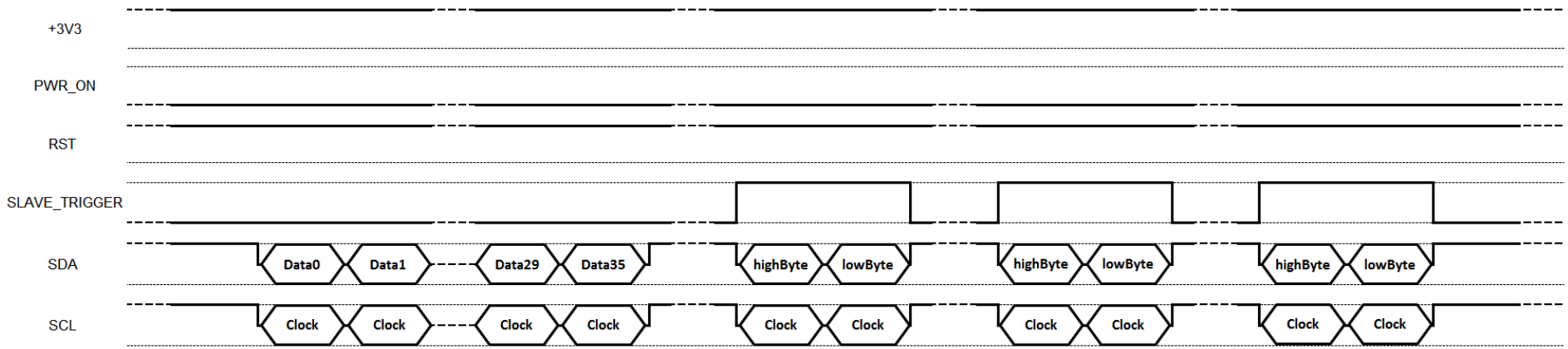


Figure 8: Timing diagram for the measurement procedure (TWI interface)

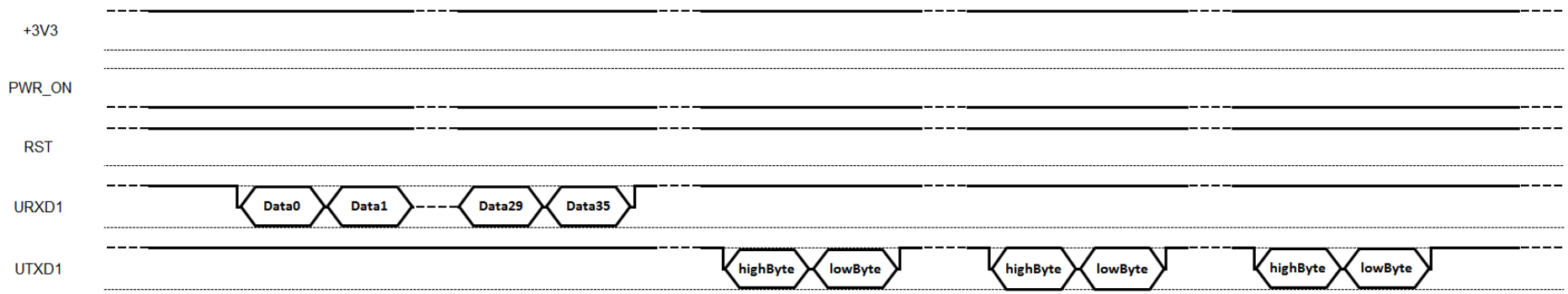


Figure 9: Timing diagram for the measurement procedure (UART interface)

⚠ **Note:** The pressure level in the cuff should be constant before the start command is sent to the module. The reason is to maximize the number of valid pulses for the following pulse wave analysis.

⚠ **Note:** It is not necessary to request the data from the module, if the host system wants to evaluate the results later.

Read out storage

The precondition for the read out of all stored measurements is an initialized module. That condition is shown by a low level on the SLAVE_TRIGGER line or the respective status message (see Chapter 7). The host system has to send the **r**ead **o**ut command (see Table 9). If the module receives that command, the module will start to transfer the stored data to the host system. The transition from low to high level on the SLAVE_TRIGGER line indicates that the data can be requested by the host system, if you are using the TWI interface. In case of the UART interface the module will send all stored data to the host system. The first three bytes contain a massage frame with STX and ETX and the number of stored measurements. This number is necessary, if the host system uses TWI to indicate the number of bytes that have to be requested. Each measurement consists out of 5,137 bytes (see Table 13). A transition from high to low level on the SLAVE_TRIGGER line indicates that all data is read out. The described read out data procedure is visualized by Figure 10 for the TWI interface. Figure 11 is valid for the UART interface.

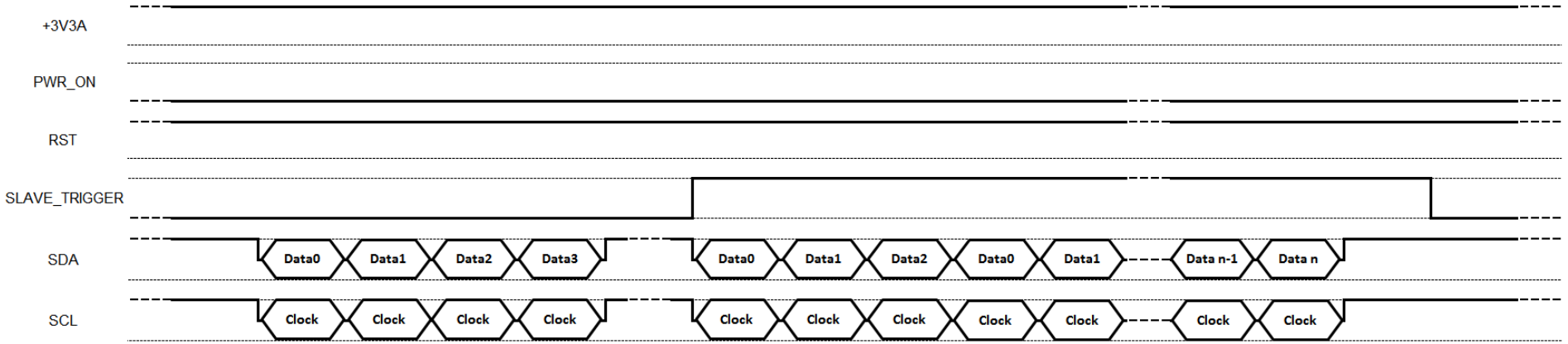


Figure 10: Timing diagram for the read out of the flash storage (TWI interface)

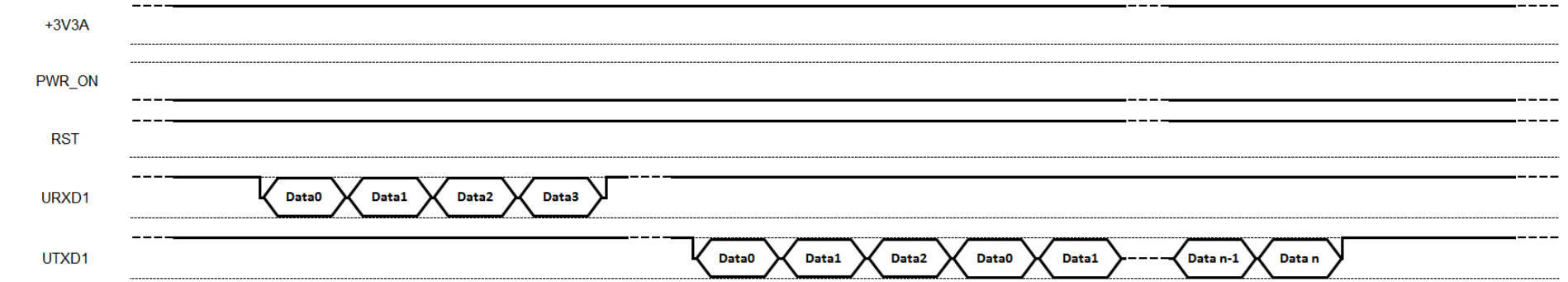
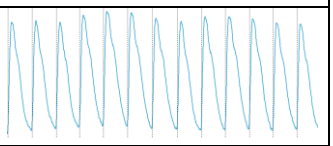
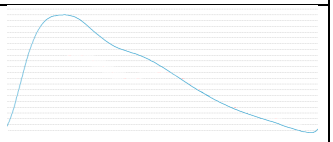


Figure 11: Timing diagram for the read out of the flash storage (UART interface)

⚠ **Note:** Firmware version 1.0 stores number of measurement, timestamp, raw data, and the peripheral pulse wave. The fields for the pulse wave analysis are filled up with dummy data. Coming firmware versions will perform the complete pulse wave analysis, store all parameters, and only need software support for the visualization of the results.

⚠ **Note:** The host system should count the number of received bytes to recognize the end of the read out process.

Table 13: Structure of the stored data on the flash of the module

Command part	Length	Example		Meaning
Start sign	1 Byte	0x02		STX
Number of measurement	1 Byte	0x00		First measurement
Separator	1 Byte	0x3D		Semicolon
Seconds	2 Byte	0x35 0x36		12 o'clock 34 minutes and 56 seconds
Minutes	2 Byte	0x33 0x34		
Hours	2 Byte	0x31 0x32		
Leap year	1 Byte	0xFF		Unused
Day	2 Byte	0x31 0x32		April 12th, 2018
Month	2 Byte	0x30 0x34		
Year	2 Byte	0x31 0x38		
Separator	1 Byte	0x3D		Semicolon
Raw data	4,800 Byte	High byte	Low byte	
		0x01	0x5B	
		0x015B		
Separator	1 Byte	0x3D		Semicolon
Peripheral pulse wave	256 Byte	High byte	Low byte	
		0x00	0x5D	
		0x005D		
Separator	1 Byte	0x3D		Semicolon
Central systolic pressure	2 Byte	0x00 0x6C 0x3D		108 mmHg+Semicolon
Central diastolic pressure	2 Byte	0x00 0x51 0x3D		81 mmHg + Semicolon
Central pulse pressure	1 Byte	0x1B 0x3D		27 mmHg + Semicolon
Augmentation pressure	2 Byte	0x04 0x3D		4 mmHg + Semicolon
Augmentation index	2 Byte	0x0E 0x3D		14 % + Semicolon
Pulse transit time	3 Byte	0x00 0x7F 0x3D		127 ms + Semicolon
Pulse wave velocity	2 Byte	0x3F 0x3D		6.3 m/s + Semicolon
Vascular age	2 Byte	0x16 0x3D		22 years + Semicolon
Cardiovascular risk	2 Byte	0x00 0x3D		0 % + Semicolon
10 Dummy (1 Byte)	20 Byte	0xDD 0x3D		221 + Semicolon
6 Dummy (2 Byte)	18 Byte	0xDD 0xDD 0x3D		56.797 + Semicolon
1 Dummy (2 Byte)	2 Byte	0xDD 0xDD		56.797
End sign	1 Byte	0x03		ETX

Erase storage

The precondition for the erase the flash storage is an initialized module. That condition is shown by a low level on the SLAVE_TRIGGER line or the respective status message (see Chapter 7). The host system has to send the **d**evice **p**rogramming command (see Table 9). If the module receives that command, the module will start to erase the flash storage that contains the taken measurements. The transition from low to high level on the SLAVE_TRIGGER line indicates that the module executes the command and the transition from high to low level on the SLAVE_TRIGGER line indicates that the storage is erased, if you are using the TWI interface. In case of the UART interface the module will send the respective status message (see Chapter 7), if the storage is erased.

The described erase procedure is visualized by Figure 12 for the TWI interface. Figure 13 is valid for the UART interface.

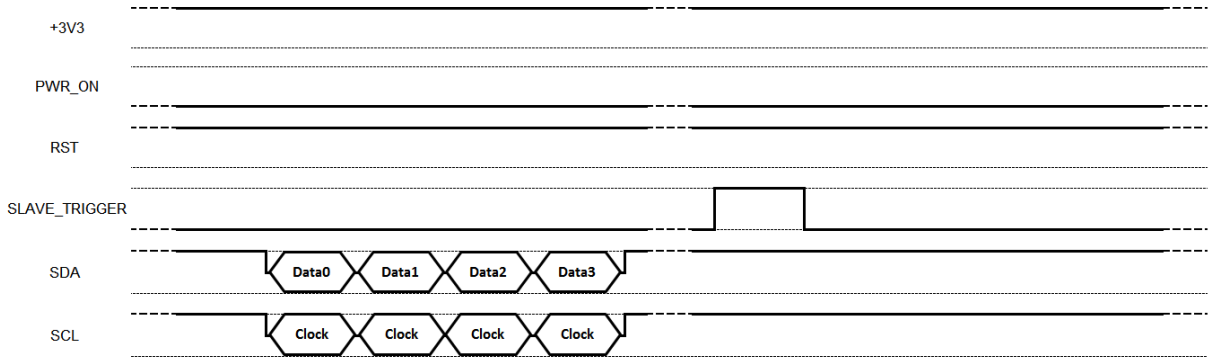


Figure 12: Timing diagram for the erase of the flash storage (TWI interface)

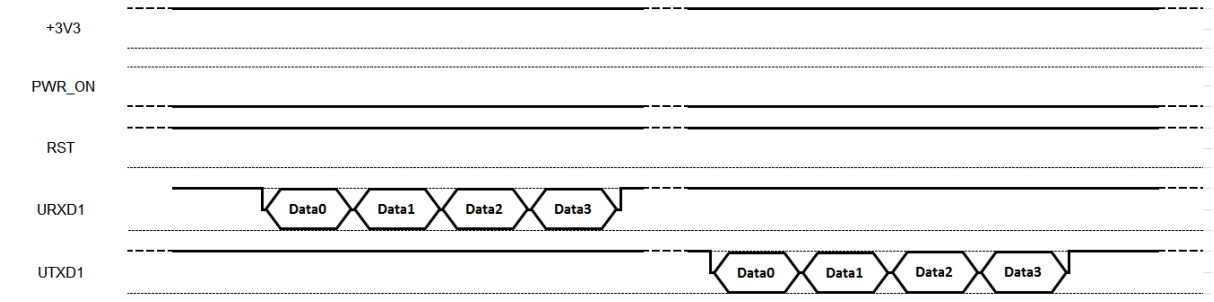


Figure 13: Timing diagram for the erase of the flash storage (UART interface)

⚠ **Note:** The host module has to read out the stored data before the erase command is sent. Otherwise the data will be lost.

Abort action

An abort command can always be sent to the module. If the module receives that command, it will abort the current process and store the received command, in case of a measurement, three times in the storage. The transition from low to high level on the SLAVE_TRIGGER line indicates that all internal processes are aborted. The SLAVE_TRIGGER line stays high until the module is reset or shut down, if you are using the TWI interface. In case of the UART interface the module will always send the respective status message (see Chapter 7).

The described abort procedure is visualized by Figure 14 for the TWI interface. Figure 15 is valid for the UART interface.

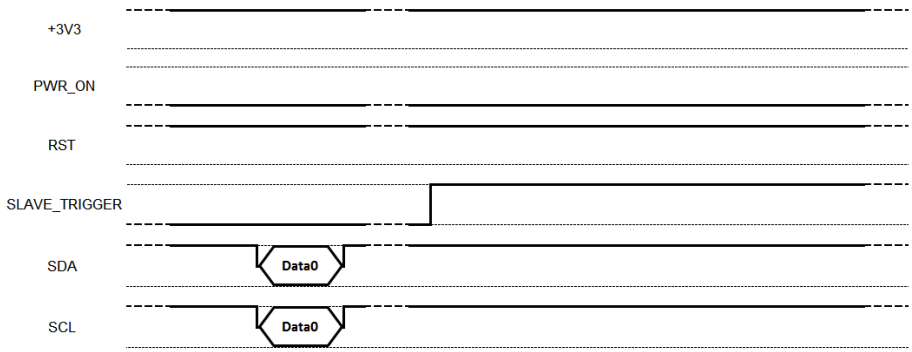


Figure 14: Timing diagram for the abort procedure by software (TWI interface)

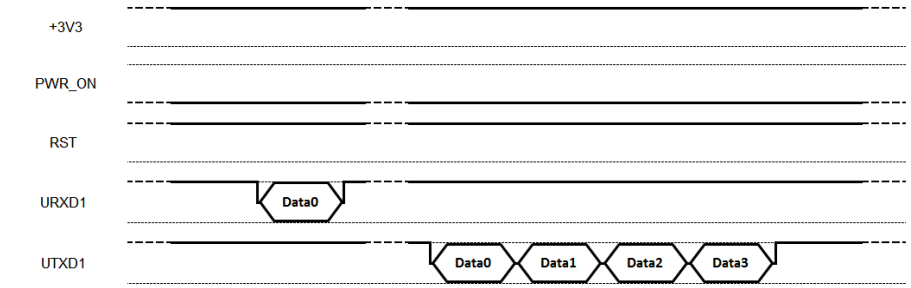


Figure 15: Timing diagram for the abort procedure by software (UART interface)

Get firmware version

The precondition for the request of the firmware version is an initialized module. That condition is shown by a low level on the SLAVE_TRIGGER line or the respective status message (see Chapter 7). The host system has to send the **g**et **v**ersion command (see Table 9). If the module receives that command, the current firmware version will be send to the host system. After the transition from low to high on the SLAVE_TRIGGER line the host system has to request this information with four dummy bytes, if you are using TWI interface. In case of the UART interface the module will automatically send the information. The bytes “Data1” and “Data2” are the firmware version. “Data1” contains the major release and “Data2” contains minor release (e.g. 1 and 0 means version 1.0).

The described firmware request procedure is visualized by Figure 16 for the TWI interface. Figure 17 is valid for the UART interface.

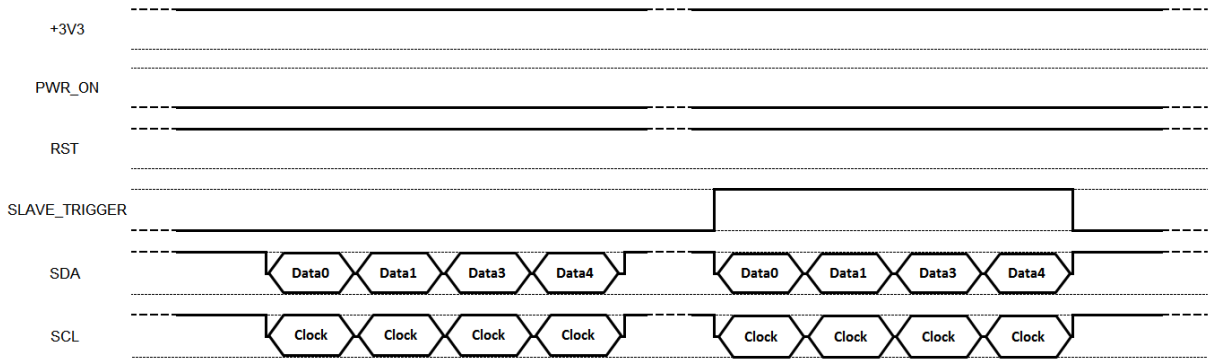


Figure 16: Timing diagram for the request of the firmware version (TWI interface)

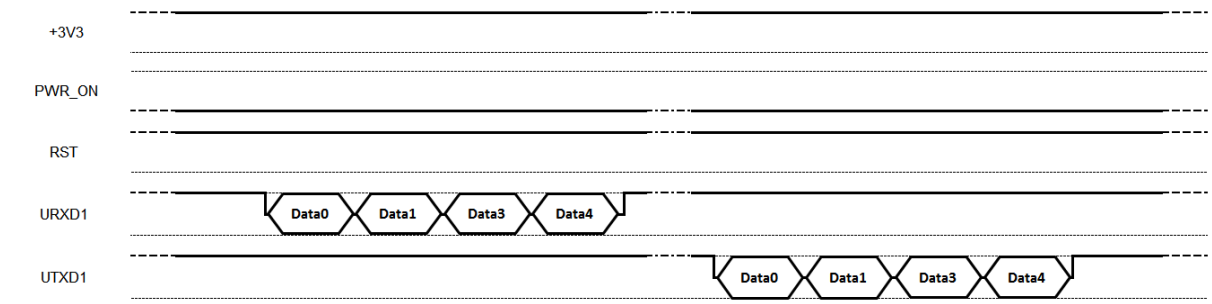


Figure 17: Timing diagram for the request of the firmware version (UART interface)

⚠ **Note:** It is not possible to request the firmware version during the execution of commands.

Get Status or error code

The precondition for the request of the status is an initialized module. That condition is shown by a low level on the SLAVE_TRIGGER line or the respective status message (see Chapter 7). The host system has to send the **g**et **s**tatus command (see Table 9). If the module receives that command, the current status will be send to the host system. After the transition from low to high on the SLAVE_TRIGGER line the host system has to request this information with four dummy bytes, if you are using TWI interface. In case of the UART interface the module will automatically send the information. The bytes “Data1” and “Data2” are the status information (e.g. 1 and 1 means S11 or 4 and 0 means E40).

The described status/error request procedure is visualized by Figure 18 for the TWI interface. Figure 19 is valid for the UART interface.

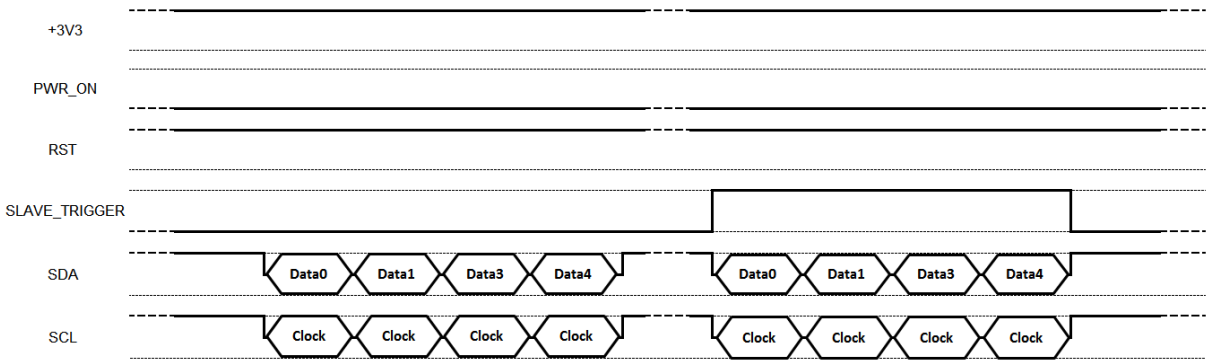


Figure 18: Timing diagram for the request of status or error codes (TWI interface)

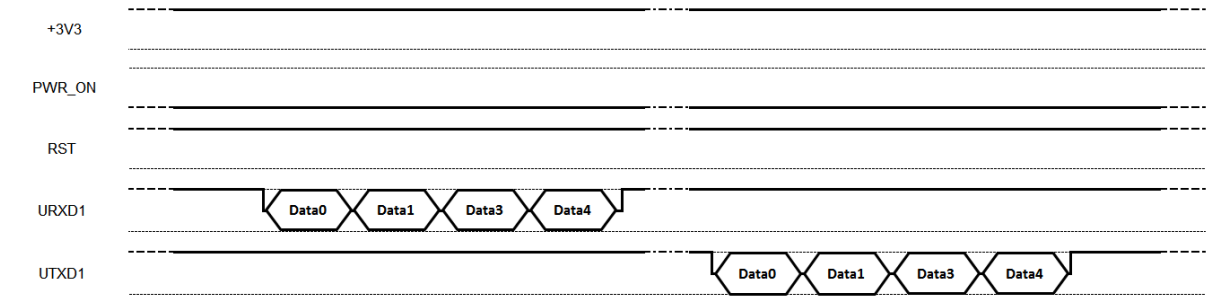


Figure 19: Timing diagram for the request of status or error codes (UART interface)

7 Status and error codes

Table 14: Status codes

Code	Meaning
S00	An action is completed.
S10	An action was aborted by the host system.
S11	Storage capacity is reached, because 100 measurements are taken.

Table 15: Error codes in case of firmware failure

Code	Meaning
E20	The timer for the sample rate is wrong initialized. The host system has to reset the module.

Table 16: Error codes in case of hardware failure

Code	Meaning
E30	The negative voltage supply does not work correctly.

Table 17: Error codes in case of algorithm failure

Code	Meaning
E40	To less valid oscillations.

⚠ **Note:** The messages from the module will only consist of the code number without the leading capital letter.

CE 0482

PAR Medizintechnik GmbH & Co. KG
Sachsendamm 6
10829 Berlin
Germany
Tel: +49 30 2350700
Fax: +49 30 2138542

info@par-berlin.com
http: www.par-berlin.com