MINDMEDIA
NEURO AND BIOFEEDBACK SYSTEMS

Firmware specifications

for

PhysioLOGx-4

Author: Mark Janssen

Data of issue: 01-02—2011
Revision: Rev1.2

Reviewed by: xx xx
Approved by: xx xx

Mind Media B.V.
Louis Eijssenweg 2B
6049CD Roermond-Herten
The Netherlands

(t) +31 (0)475 10 01 23
(f) +31 (0)475 33 06 02
(i) www.mindmedia.com
(e) sales@mindmedia.com

Table of Contents

1 Introduction

1.1 Scope

This document describes the firmware specifications for the PhysioLOGx-4. This low-budget system will succeed the EEG trainer 2/4 system.

The PhysioLOGx-4 system has two fast, 1024SPS (samples per second), ExG channels and two slower, 256SPS, auxiliary channels. Beside the physiological inputs it also includes a light (for generating light stimuli) output, an audio output (for generating audio stimuli) and two digital in-/outputs (TTL1 and TTL2).

The PhysioLOGx-4 is connected to the computer with a high speed USB connection, which has a transfer rate up to 1MBaud.

1.2 Intended audience

This document is written for:

- Mind Media B.V. development engineers

1.3 Symbols and definitions

Key	Explanation

1.4 References

Reference #	Title	Version

1.5 Document history

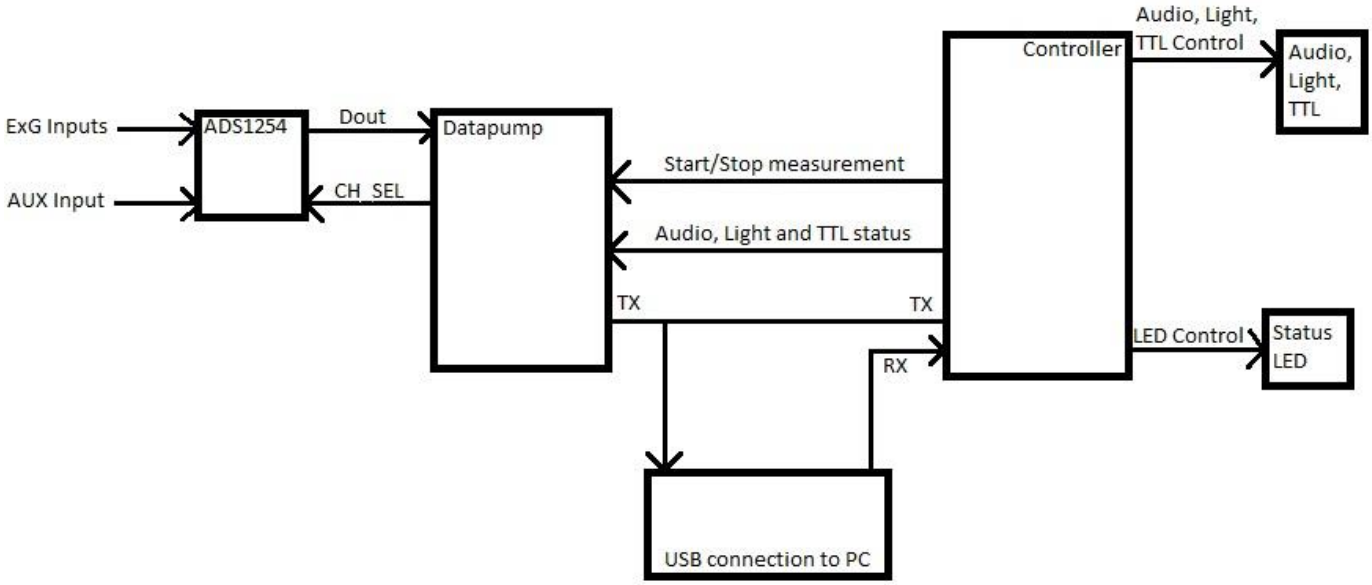
Revision	Date	Author	Description
1.0	24-03-2011	M. Janssen	Initial Version
1.1	31-03-2011	M. Janssen	Minor changes
1.2	01-02-2012	M. Janssen	Removed NACK command, added additional audio functionality.

2 System overview

The PhysioLOGx-4 system is based on two microcontrollers, one called “datapump” the other called “controller”. The datapump performs all time critical tasks and has one dedicated function: throughput of measured data and status bits to the PC by an USB connection. The controller performs all non-time-critical tasks and is used for the higher interface functions (i.e. start/stop measurement, provide device information, etc.).

Spec.1: The datapump and the controller share the same USB transmit line, this means that the two controllers cannot simultaneously send data to the PC.

The controller is the default transmit channel to the PC. The datapump will have exclusive control over the transmit line when the controller is requested a start measurement and the datapump is signaled that a measurement is started. The controller may not use the transmit line until a stop measurement is requested and the datapump is signaled that the measurement has stopped.



3. Serial protocols

This chapter describes the digital interface of the PhysioLOGx-4. The USB connection between the controllers and the PC is managed by a UART-to-USB bridge, FT232R, at a baud rate of 1Mbps.

3.1 Data format

The data format used for communication between the controller and the PC is described below. The controller is able to send and receive data from the PC. The PC sends a "command" to the front-end (PhysioLOGx-4 device), the front-end returns a response to the PC.

Description	Format	Value	Remark
Header	Byte[2]	0xAAAA	Header that identifies a command/response
Command ID	Byte[2]	0..65535	Command/response identifier
Command size	Byte[2]	0..65535	Total amount of bytes in the command/reponse
Payload	Byte [0..65527]	0..255	Command/response payload
Checksum	Byte[2]	0..65535	(Header + Command ID + Command size + Payload[0..n] + Checksum) MOD65536 == 0

Most significant byte first.

The data format used for communication between the datapump and the PC is described below. The datapump is only capable of sending data.

Description	Format	Value	Remark
Header	Byte[1]	0xAA	Header that identifies a datapacket
Packet count	Byte[1]	0..255	Packet number. To monitor the stability of the connection
ExG Channel A	Byte[3]	0..16777216	First sample of ExG channel A
ExG Channel B	Byte[3]	0..16777216	First sample of ExG channel B
AUX Channel C	Byte[3]	0..16777216	Sample of channel C
ExG Channel A	Byte[3]	0..16777216	Second sample of ExG channel A
ExG Channel B	Byte[3]	0..16777216	Second sample of ExG channel B
ExG Channel A	Byte[3]	0..16777216	Third sample of ExG channel A
ExG Channel B	Byte[3]	0..16777216	Third sample of ExG channel B
AUX Channel D	Byte[3]	0..16777216	Sample of channel D
ExG Channel A	Byte[3]	0..16777216	Fourth sample of ExG channel A
ExG Channel B	Byte[3]	0..16777216	Fourth sample of ExG channel B
Status	Byte[4]	Byte1: 0..255 Byte2: 0..255 Byte3: 0..255 Byte4: 0..255	Four status bytes. Contain the status of the TTL1, TTL2, Audio and Light I/O's. Sampled at 1024SPS. bit7..bit4 = don't care. Bit3 = TTL2 status. Bit2 = TTL1 status. Bit1 = Light status. Bit0 = Audio status. 1 == Active, 0 == Not active
Checksum	Bytes[1]	0..255	(Byte[0] + Byte[1] + … Byte[n] + Checksum) MOD256 == 0

Most significant byte first.

Spec.2: The controller will signal the datapump when a measurement is started.

Spec.3: When a measurement is started the datapump will send a continuous data stream with a sample rate of 1024SPS to the PC.

Spec.4: While a measurement is active, the datapump has exclusive control over the transmit line and the controller is unable to send and receive commands.

3.2 Command overview

All possible commands and responses used for communication between the PC and the controller PIC are listed in the table below.

Category	PC - Message ID	Dir.	Controller PIC - Response ID	Packet-type	Remark
Device info	Read device info	>		0x0003	Device info can only be read and written if no measurement is started!
		<	Device info	0x0002	
	Write device info	>		0x0004	
		<	#1. Acknowledge	0x0000	Send if payload size is correct
EEPROM data	Read EEPROM	>		0x0006	EEPROM data can only be read and written if no measurement is started!
		<	EEPROM data	0x0005	
	Write EEPROM	>		0x0007	
		<	#1. Acknowledge	0x0000	Send if payload size is correct
Config I/O	Config I/O	>		0x0008	Allowed only during measurement!
		<		0x0000	Send if payload size is correct
Generate tone	Generate tone	>		0x0009	Allowed only during measurement!
		<		0x0000	Send if payload size is correct
Generate light	Generate light	>		0x000A	Allowed only during measurement!
		<		0x0000	Send if payload size is correct
Start measurement	Start measurement	>		0x000B	
		<	#1. Acknowledge	0x0000	Send if payload size is correct
Stop measurement	Stop measurement	>		0x000C	
		<	#1. Acknowledge	0x0000	Send if payload size is correct

3.3 Commands/responses

0x0000 Acknowledge

Field	Value	Remark
Response ID	0x0000	All acknowledge reponses contain an error report
Response size	0x0031	
Payload	Byte[41]	Error report
Direction	Frontend -> PC	

The error report contains specific results:

Field	Value	Description
Cause	Byte[1]	0x00 = ERR_NO_ERROR 0x01 = ERR_WRONG_CHK_SUM 0x02 = ERR_WRONG_CMD_ID 0x03 = ERR_WRONG_PAYLOAD_SIZE 0x04 = ERR_ARG_OUT_OF_RANGE
Arg1	Byte[4]	Additional error parameter 1
Arg2	Byte[4]	Additional error parameter 2
System Error Text	Byte[32]	System error text, max 31 characters, last character is always 0.

0x0002 Device info

Field	Value	Remark
Response ID	0x0002	
Response size	0x0012	
Payload	Byte[10]	Device info
Direction	Frontend -> PC	

The device info contains specific information:

Field	Value	Description
Device ID	Byte[2]	Set in the firmware, cannot be changed from outside.
Software Version	Byte[2]	Set in the firmware, cannot be changed from outside.
Hardware Version	Byte[2]	Set by the write device info command.
Serial Number	Byte[4]	Set by the write device info command.

0x0003 Read device info

Field	Value	Remark
Command ID	0x0003	Allowed only if no measurement is started!
Command size	0x0008	
Payload	-	No payload
Direction	PC -> Frontend	

0x0004 Write device info

Field	Value	Remark
Command ID	0x0004	Allowed only if no measurement is started!
Command size	0x0012	
Payload	Byte[6]	Hardware PCB version, Device serial number
Direction	PC -> Frontend	

0x0005 EEPROM data

Field	Value	Remark
Response ID	0x0005	
Response size	Byte[2]	
Payload	Byte[2..248]	Address[1b], Size[1b], free programmable EEPROM memory [max. 246b].
Direction	Frontend -> PC	

0x0006 Read EEPROM

Field	Value	Remark
Command ID	0x0006	Allowed only if no measurement is started!
Command size	0x000A	
Payload	Byte[2]	Address[1b], Size[1b],
Direction	PC -> Frontend	

0x0007 Write EEPROM

Field	Value	Remark
Command ID	0x0007	Allowed only if no measurement is started!
Command size	0x000A .. 0x006E	
Payload	Byte[2..248]	Address[1b], Size[1b], free programmable EEPROM memory [max. 246b].
Direction	PC -> Frontend	

0x0008 Config I/O

Field	Value	Remark
Command ID	0x0008	Allow only if a measurement is started!
Command size	0x0009	
Payload	Byte[1]	TTL configuration
Direction	PC -> Frontend	

The TTL configuration byte contains specific information:

Field	Value	Description
	Bit[7,6,3,2]	Not used
TTL1 Control	Bit[5]	TTL1 Control (‘1’ = input, ‘0’ = output)
TTL2 Control	Bit[4]	TTL2 Control (‘1’ = input, ‘0’ = output)
TTL1 Status	Bit[1]	TTL1 Status (‘1’ = high, ‘0’ = low)
TTL2 Status	Bit[0]	TTL2 Status (‘1’ = high, ‘0’ = low)

0x0009 Generate tone

Field	Value	Remark
Command ID	0x0009	Allow only if a measurement is started!
Command size	0x0014	
Payload	Byte[12]	Audio configuration
Direction	PC -> Frontend	

The audio configuration contains specific information:

Field	Value	Description
Duration	Byte[2]	Total audio program duration
Frequency	Byte[2]	Audio tone frequency
Left ON Time	Byte[2]	Left speaker ON time
Left OFF Time	Byte[2]	Left speaker OFF time
Right ON Time	Byte[2]	Right speaker ON time
Right OFF Time	Byte[2]	Right speaker OFF time

0x000A Generate light

Field	Value	Remark
Message ID	0x000A	Allow only if a measurement is started!
Message Size	0x0014	
Payload	Byte[12]	Light configuration
Direction	PC -> Frontend	

The light configuration contains specific information:

Field	Value	Description
Duration	Byte[2]	Total light program duration
Left ON Time	Byte[2]	Left side ON time
Left OFF Time	Byte[2]	Left side OFF time
Left side intensity	Byte[1]	Left side intensity
Right ON Time	Byte[2]	Right side ON time
Right OFF Time	Byte[2]	Right side OFF time
Right side intensity	Byte[1]	Right side intensity

0x000B Start measurement

Field	Value	Remark
Command ID	0x000B	
Command size	0x0008	
Payload	-	No payload
Direction	PC -> Frontend	

0x000C Stop measurement

Field	Value	Remark
Command ID	0x000C	
Command size	0x0008	
Payload	-	No payload
Direction	PC -> Frontend	

4 Additional specifications

4.1 Device info

Spec.5: The controller is able to receive device information (serial number and hardware PCB version) from the PC, and store these values in the EEPROM of the controller.

Spec.6: The controller is able to read the device information (serial number, hardware PCB version, software version and device ID) from the EEPROM and send the device information to the PC.

4.2 EEPROM data

Spec.7: The controller has 246bytes free programmable EEPROM memory. Start address is 0, end address is 245.

4.3 Light output

Spec.8 The controller is able to independently drive two separate led circuits.

Spec.9: The controller is able to independently set the intensity of the two circuits.

Spec.10: The led circuits can independently be set with different ON and OFF times (duty cycle). If these times are 0 the concerning circuit if turned off.

Spec.11: Light circuits are activated for a specified duration, which may not exceed 65536ms (2^{16}). If the duration is 0 both circuits are either turned off or not turned on.

Spec.12: Immediately after a led circuit is activated a feedback output notifies the datapump of the light activity.

Spec.13: Immediately after activity on both led circuits has ended the feedback output is cleared notifying the datapump that the light activity had ended.

Spec.14: If one or both of the light circuits are activated while an audio circuit is active, audio activity will be stopped and the requested light activity will start.

4.4 Audio output

Spec.15: The controller is able to interface with a DAC module used to produce an audio tone by the BioMonitor4 electronics.

Spec.16: The controller is able to independently drive two separate audio circuits.

Spec.17: The audio circuits can independently be set with different ON and OFF times (duty cycle). If these times are 0 the concerning circuit is turned off.

Spec.18: The controller is able to generate tones with frequencies between (200Hz and 10000Hz).

***Spec.*19:** Audio circuits are activated for a specified duration, which may not exceed 65536ms (2^{16}**).**

***Spec.*20:** The amplitude of the sine signal used to drive the speakers is fixed and managed by the electronics.

Spec.21: Immediately after an audio circuit is activated a feedback output notifies the datapump of the audio activity.

Spec.22: Immediately after activity on both audio circuits has ended the feedback output is cleared notifying the datapump that the audio activity had ended.

Spec.23: If one or both of the audio circuits are activated while a light circuit is active, light activity will be stopped and the requested audio activity will start.

4.5 TTL IN-/OUTPUTS

Spec.24: The controller is able to independently change the direction of the TTL lines (input/output).

Spec.25: The controller is able to independently change the pin state of the TTL lines if configured as output.

Spec.26: The control of TTL in-/outputs is independent from the light and audio circuits.

Spec.27: A feedback line will constantly notify the datapump if the pin state of the TTL lines is either high or low, independent of the direction of the TTL lines.

4.6 SYSTEM STATUS

Spec.28: The controller is able to show the system status via a bi-color led, which can be green, orange and red.
A green light indicates that system power is ON and OK.
An orange light indicates that data acquisition has started.
A blinking red light indicates a system error.

4.7 ADS1254

Spec.29: The datapump is able to change the channel selection of the ADS1254.

Spec.30: The datapump will wait for the ADS1254 to signal that data from the selected channel is ready to be read.

4.7 STATUS BYTE

Spec.31: Status from the TTL, audio and light lines is measured at the highest possible data rate (1024SPS).

Spec.32: The status byte, send by the datapump to the PC, has the following layout:

Bit 7	Bit 6	Bit5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
x	x	x	x	TTL2 status	TTL1 status	Light Status	Audio status

<end of document>