– ScienceMode –
RehaStim™ Stimulation Device

Description and Protocol

Thomas Schauer[1], **Nils-Otto Negaard, Carsten Behling**[2]

[1] Technische Universität Berlin, Control Systems Group, Germany
E-mail: schauer@control.tu-berlin.de
[2] HASOMED GmbH, Mageburg, Germany
E-mail: carsten.behling@hasomed.de

21st September 2009

1 Introduction

ScienceMode is a serial communication protocol to directly control the 8-channel stimulator RehaStim^{TM} by the company HASOMED GmbH (see http://www.hasomed.de or http://www.rehastim.de/) from an external device, preferably a PC, via the standard USB interface. The connection between PC and stimulator corresponds to the USB 1.1 standard. Galvanic isolation is provided by the USB interface of the stimulator. The ScienceMode offers great flexibility for research applications.

2 Stimulation Device

The current-controlled 8 channel stimulator RehaStim^{TM} is a certified medical product and possesses two independent current sources which are multiplexed to 4 outputs each. The main processor of the stimulator is an ultra-low-power 16-bit RISC mixed-signal processor from Texas Instruments (MSP430). There are two independent stimulation modules hosting each one of the current sources. Each stimulation module owns another microprocessor (MSP430) which is responsible for the pulse generation timing. The stimulation module A has the stimulation channels 1 to 4, stimulation module B has the stimulation channels 5 to 8. Since both modules function independently simultaneous pulse generation on module A and B is possible.
Figure 1 shows the stimulator device. The device is operated by a touch panel with illumination. The technical specifications are listed in the Table 1. Galvanic isolation between high voltage generation/electrodes and the rest of the stimulator electronics has been additionally realised for safety reasons.

Figure 1: Portable 8 channel stimulator RehaStim.

Table 1: Technical details of the RehaStim stimulation device.

Current	$0 \ldots 126$ mA in 2 mA steps
Pulsewidth	$0, 20 \ldots 500\,\mu$s in 1 μs steps
Frequency	see Section 4
Pulse form	biphasic
Channels	8 (2 times 4 on two modules)
Serial ports	USB with galvanic isolation

Figure 2 shows the form of a delivered bi-phasic pulse on an ideal resistive load. Current amplitude and pulsewidth are defined in the figure. Notice that there is a fixed pause of 100 μs between the two phases of the pulse. At the end of the pulse the remaining charge on the electrodes and skin is removed by an active shortcut (change of electrode polarity for 1 μs).

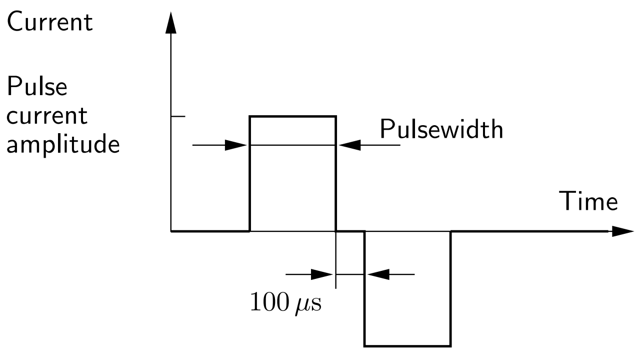

Figure 2: Definition of pulsewidth and current amplitude for a biphasic pulse.

The stimulator provides a skin resistance check for safety reasons. The resistance is determined by analysing the effect of a small test impulse which is send before each stimulation pulse. If the resistance is not inside normal ranges then a stimulation pulse will not be generated.

3 How to activate/deactivate the ScienceMode

The operating manual of the RehaStim device should give detailed instructions how to activate/deactivate the ScienceMode.

4 Pulse Generation Modes

The stimulator offers different modes of stimulation pulse generation.
The following modes of pulse generation are available:

- **Single Pulse Mode:** On an external command the stimulator generates a single pulse on a specified channel with desired current amplitude and pulsewidth. The stimulator will generate the pulse immediately after processing the command. Complex stimulation patterns may be generated by sending more than one command. The external device (PC) is responsible for controlling the stimulation timing, i.e. the stimulation inter-pulse interval.

- **Continuous Channel List (CCL) Mode:** Using this mode, the generation of complex patterns is greatly simplified. The main processor and the processors of the stimulation modules control the pulse generation by means of timer-interrupts. A list of stimulation channels has to be specified, on which pulses or even pulse groups (doublets or triplets) will repeatedly be generated. Figure 3 defines the main stimulation period t_1 and the inter-pulse time t_2 of doublets and triplets with the help of an example. The channel list is repeatedly processed with time period t_1. Pulse generation takes place on the selected channels, ordered by the channel numbers. For each selected channel a time slot of 1.5 ms is reserved, even if current or/and pulsewidth are zero for the channel. At least 1.5 ms pass between the stimulation of different channels of one module. The stimulation modules A and B process the channel list in parallel with a time offset of 0.6 ms. Module A generates pulses on the channels 1 to 4 if applicable, and module

B generates pulses on the channels 5 to 8 if applicable. The inter-pulse time t_2 of doublets and triplets is fixed for all channels and is set during an initialisation step. When doublets or triplets have to be generated the channel list will be processed two or three times with period t_2. Stimulation takes place only on the channels on which doublets or triplets have to be generated.

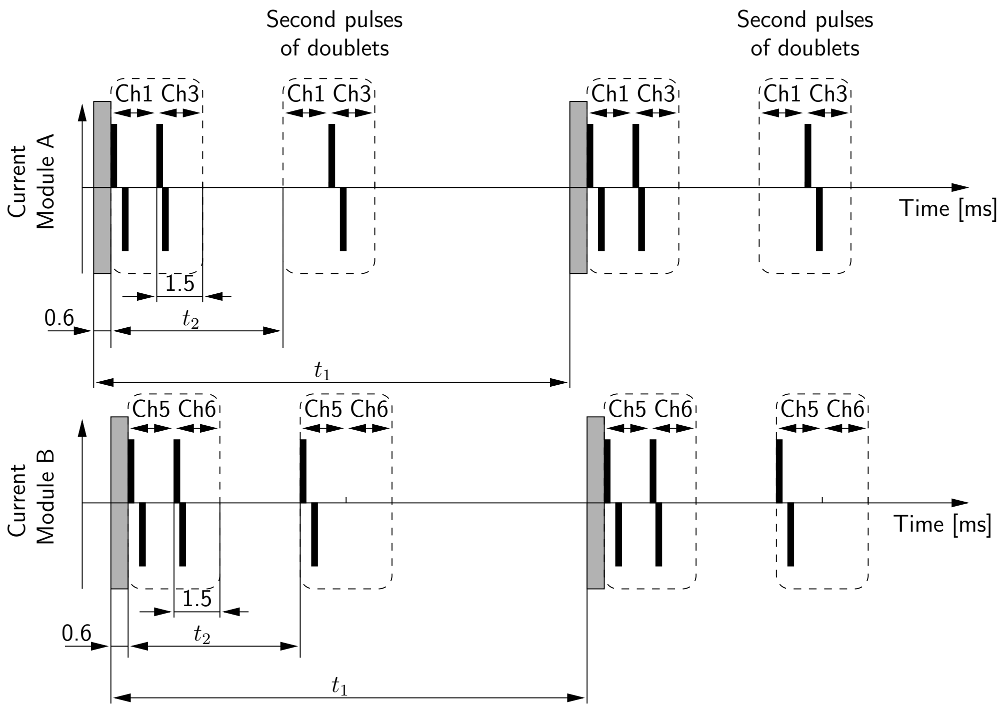

Figure 3: Example for the Continuous Channel List Mode of the stimulator. The channel list encloses the channels 1, 3, 5 and 6. The main stimulation frequency is $1/t_1$. Doublets are generated on the channels 3 and 5 with a frequency $1/t_2$. The grey bars indicate some communication periods in which the actual stimulation settings are transfered from the main controller to the stimulation modules. Stimulation pulses are represented by black bars where the same width and amplitude have been assumed for simplicity.

The main stimulation frequency is specified by the period t_1. Some channels of the channel list can be assigned to a lower frequency which has the period nt_1 where n is a positive integer.

The CCL mode must be initialised by a command. Used channels, main time t_1, inter-pulse time t_2 and the maximal size of pulse groups must be chosen at this stage. The minimal possible inter-pulse time t_2 depends on the maximal number of channels assigned to the individual stimulation modules as follows

$$t_2 \geq 1.5\,\text{ms} \cdot \max(n_{ch_A}, n_{ch_B}) \tag{1}$$

where n_{ch_A} and n_{ch_B} are the number of selected channels of stimulation modules A and B respectively. The minimal possible period t_1 depends on t_2 and the maximal size n_{pg} of pulse groups used ($n_{pg} = 1$ for single pulses, $n_{pg} = 2$ for doublets, $n_{pg} = 3$ for triplets). The constraint for t_1 is then

$$t_1 \geq n_{pg} \cdot t_2 + 1.5\,\text{ms} \tag{2}$$

where the 1.5 ms above results from a communication between the main processor and the stimulation modules as indicated in Figure 3. The period t_1 can be changed in the range $3 \ldots 1023.5$ ms in 0.5 ms steps subject to the constraint (2), and t_2 can be altered in the range $3 \ldots 16$ ms in 0.5 ms steps subject to the constraint (1). Using all 8 channels, a minimal group time t_2 of 6 ms can be achieved, i.e. a doublet or triplet frequency of 180 Hz. When only doublets and single pulses with a frequency of 180 Hz are applied ($n_{pg} = 2$), a minimal value of 13.5 ms for period t_1 is obtained, i.e. a maximal possible main frequency of 74.1 Hz.

Another external command deactivates the CCL mode. When the CCL mode is active, the pulse parameters (pulsewidth, current amplitude and group mode (single pulse, doublet or triplet) of the selected channels can be altered by a corresponding external command. The new parameters will be used from the next processing of the channel list onwards.

The main processor controls the frequency $1/t_1$ by which the channel list is processed. Each time the stimulation cycle repeats, the main processor sends the actual stimulation settings to the two stimulation modules which then generates the individual pulses on the selected channels. The processors on the stimulation modules guarantee that specified pulsewidths and inter-pulse times of the doubles and triples are realised.

- **One Shot Channel List (OSCL) Mode:** Just as in the Continuous Channel List Mode a channel list is defined by an initialisation step. However, processing of the channel list is not automatically repeated so that the time period t_1 ceases to exist. Instead, the channel list will be processed once and pulses and/or pulse groups are generated, only if an external command is issued. Pulse parameters (pulsewidth, current amplitude and group mode of the selected channels) are specified by the external command.

  The OSCL mode offers the possibility to control the main stimulation frequency by an external device while the inter-pulse time of the doublets and triplets is realised by the stimulation modules.

5 Protocol

5.1 Serial Settings for FTDI driver

Table 2: Serial settings

Parameter	Value
Baudrate	115200
parity	no
data bits	8
stop bit	2
flow control RTS/CTS	on

5.2 Overview on commands

All information is contained in packets of one or more bytes conforming with some predefined sequence. In each packet all bytes other than the first have bit 7 clear; the starting byte has bit 7 set.

Table 3: Commands with their 2 bit identification number `Ident`

No.	Command	Identification number `Ident`
1	Channel list mode initialisation	00
2	Channel list mode update	01
3	Channel list mode stop	10
4	Single pulse generation	11

5.3 Channel list mode initialisation command

Table 4: Variables used for channel list mode initialisation command

Variable	Bits	Value/Range	Description
`Ident`	2	0	Command identification number
`Check`	3	0..7	Checksum = sum of all logical variables Modulo 8 = (`N_Factor`+`Channel_Stim` +`Channel_Lf`+ `Group_Time`+`Main_Time`) modulo 8
`N_Factor`	3	0..7	Defines how many times the stimulation is skipped for channels specified in `Channel_Lf` as well as in `Channel_Stim`. 0 = no skip 1 = skip once ... 7 = skip seven times
`Channel_Stim`	8	0..255	Defines the actives channels. Bit 0 corresponds to channel 1. ⋮ Bit 7 corresponds to channel 8. A bit set activates a channel.
`Channel_Lf`	8	0..255	Defines the low frequency channels. Bit 0 corresponds to channel 1. ⋮ Bit 7 corresponds to channel 8. A bit set activates a channel for low frequency stimulation but only if the same bit is also set in `Channel_Stim`.
`Group_Time`	5	0..31	Defines the interpulse-interval t_{s_2} by $t_{s_2} = \texttt{Group\_Time} \cdot 0.5\,\mathrm{ms} + 1.5\,\mathrm{ms}$
`Main_Time`	11	0..2047	Defines the main time period t_{s_1} by $t_{s_1} = \texttt{Main\_Time} \cdot 0.5\,\mathrm{ms} + 1\,\mathrm{ms}$

Table 5: Definition of the channel mode list initialisation command

Byte	Bits	Value	Variable	Bit no. with respect to the variable
Byte 1	7	1		
	6	0	`Ident`	1
	5	0	`Ident`	0
	4		`Check`	2
	3		`Check`	1
	2		`Check`	0
	1		`N_Factor`	2
	0		`N_Factor`	1
Byte 2	7	0		
	6		`N_Factor`	0
	5		`Channel_Stim`	7
	4		`Channel_Stim`	6
	3		`Channel_Stim`	5
	2		`Channel_Stim`	4
	1		`Channel_Stim`	3
	0		`Channel_Stim`	2
Byte 3	7	0		
	6		`Channel_Stim`	1
	5		`Channel_Stim`	0
	4		`Channel_Lf`	7
	3		`Channel_Lf`	6
	2		`Channel_Lf`	5
	1		`Channel_Lf`	4
	0		`Channel_Lf`	3
Byte 4	7	0		
	6		`Channel_Lf`	2
	5		`Channel_Lf`	1
	4		`Channel_Lf`	0
	3	X		
	2	X		
	1		`Group_Time`	4
	0		`Group_Time`	3
Byte 5	7	0		
	6		`Group_Time`	2
	5		`Group_Time`	1
	4		`Group_Time`	0
	3		`Main_Time`	10
	2		`Main_Time`	9
	1		`Main_Time`	8
	0		`Main_Time`	7
Byte 6	7	0		
	6		`Main_Time`	6
	5		`Main_Time`	5
	4		`Main_Time`	4
	3		`Main_Time`	3
	2		`Main_Time`	2
	1		`Main_Time`	1
	0		`Main_Time`	0

5.4 Channel list mode update command

Table 6: Variables used for channel list mode update command

Variable	Bits	Value/Range	Description
Ident	2	1	Command identification number
Check	5	0..31	Checksum = sum of all logical variables Modulo 32 = (Mode+Pulse_Width+Pulse_Current) modulo 32
Mode	2	0..2	Mode = 0: generate single pulse Mode = 1: generate doublet Mode = 2: generate triplet
Pulse_Width	9	0,10..500	Pulse width in μs
Pulse_Current	7	0..127	Current in mA

Table 7: Definition of the channel mode update command

Byte	Bits	Value	Variable	Bit no. with respect to the variable
Byte 1	7	1		
	6	0	`Ident`	1
	5	1	`Ident`	0
	4		`Check`	4
	3		`Check`	3
	2		`Check`	2
	1		`Check`	1
	0		`Check`	0
For each channel activated in the channel list, the next three bytes are send in increasing order with respect to the channel number.				
Byte 2	7	0		
	6		`Mode`	1
	5		`Mode`	0
	4	X		
	3	X		
	2	X		
	1		`Pulse_Width`	8
	0		`Pulse_Width`	7
Byte 3	7	0		
	6		`Pulse_Width`	6
	5		`Pulse_Width`	5
	4		`Pulse_Width`	4
	3		`Pulse_Width`	3
	2		`Pulse_Width`	2
	1		`Pulse_Width`	1
	0		`Pulse_Width`	0
Byte 4	7	0		
	6		`Pulse_Current`	6
	5		`Pulse_Current`	5
	4		`Pulse_Current`	4
	3		`Pulse_Current`	3
	2		`Pulse_Current`	2
	1		`Pulse_Current`	1
	0		`Pulse_Current`	0
⋮				

5.5 Channel list mode stop command

Table 8: Variables used for channel list mode stop command

Variable	Bits	Value/Range	Description
`Ident`	2	2	Command identification number
`Check`	5	0..31	Checksum = sum of all logical variables Modulo 32 = (0) modulo 32 = 0

Table 9: Definition of the channel mode stop command

Byte	Bits	Value	Variable	Bit no. with respect to the variable
Byte 1	7	1		
	6	1	`Ident`	1
	5	0	`Ident`	0
	4	0	`Check`	4
	3	0	`Check`	3
	2	0	`Check`	2
	1	0	`Check`	1
	0	0	`Check`	0

5.6 Single pulse generation command

Table 10: Variables used for single pulse generation command

Variable	Bits	Value/Range	Description
`Ident`	2	3	Command identification number
`Check`	5	0..31	Checksum = sum of all logical variables Modulo 32 = (`Channel_Number` + `Pulse_Width` + `Pulse_Current`) modulo 32
`Channel_Number`	3	0..7	`Channel_Number` = 0 is channel no. 1 ⋮ `Channel_Number` = 7 is channel no. 8
`Pulse_Width`	9	0,10..500	Pulse width in μs
`Pulse_Current`	7	0..127	Current in mA

Table 11: Definition of the single pulse generation command

Byte	Bits	Value	Variable	Bit no. with respect to the variable
Byte 1	7	1		
	6	1	Ident	1
	5	1	Ident	1
	4		Check	4
	3		Check	3
	2		Check	2
	1		Check	1
	0		Check	0
Byte 2	7	0		
	6		Channel_Number	2
	5		Channel_Number	1
	4		Channel_Number	0
	3	X		
	2	X		
	1		Pulse_Width	8
	0		Pulse_Width	7
Byte 3	7	0		
	6		Pulse_Width	6
	5		Pulse_Width	5
	4		Pulse_Width	4
	3		Pulse_Width	3
	2		Pulse_Width	2
	1		Pulse_Width	1
	0		Pulse_Width	0
Byte 4	7	0		
	6		Pulse_Current	6
	5		Pulse_Current	5
	4		Pulse_Current	4
	3		Pulse_Current	3
	2		Pulse_Current	2
	1		Pulse_Current	1
	0		Pulse_Current	0

5.7 Acknowledgement by the stimulator

Table 12: Acknowledgement byte

Byte	Bits	Value	Variable	Bit no. with respect to the variable
Byte 1	7		`Ident`	1
	6		`Ident`	0
	5			
	4			
	3			
	2			
	1			
	0	1 = OK, 0 = error	`Error_Code`	0

5.8 Examples

Single pulse example 1

Sending a pulse on channel 3 with a pulse width of $200\,\mu$s (binary 011001000) and a current of 120 mA (binary 1111000). Hence, `Channel_Number` is 2 (binary 010). In this case, the checksum is $(2 + 200 + 120) \bmod 32 = 2$ (binary 00010). In binary format, the modulo 32 operation can be easily performed by just taking the 5 LSB of the sum 2+200+120 (binary 101000010). The byte sequence for the command is:

Byte 0 – 11100010
Byte 1 – 0010XX01
Byte 2 – 01001000
Byte 3 – 01111000

The bits with XX do not have a meaning. In hex code the command is E2 21 48 78.
The return value would be in hex C1 (binary 11000001) if no error occurred else C0 (11000000).

Single pulse example 2

Sending a pulse on channel 6 with a pulse width of $221\,\mu$s (binary 011011101) and a current of 55 mA (binary 0110111). Hence, `Channel_Number` is 5 (binary 101).The checksum is in this case $(5 + 221 + 51) \bmod 32 = 25$ (binary 11001). In binary format, the modulo 32 operation can be easily performed by just taking the 5 LSB of the sum 5+221+55 (binary 100011001). The byte sequence for the command is:

Byte 0 – 11111001
Byte 1 – 0101XX01
Byte 2 – 01011101
Byte 3 – 00110111

The bits with XX do not have a meaning. In hex code the command is F9 51 5D 37.
The return value would be in hex C1 (binary 11000001) if no error occurred else C0 (11000000).

Channel list mode initialisation example 1

The channel list mode shall be initialised with the following parameters:
`Main_Time`=98 (binary 00001100010) gives t_{s_1}=50 ms
`Group_Time`=7 (binary 00111) gives t_{s_2}=5 ms
`N_Factor`=1 (binary 001)
Channels to be activated are 1,2 and 5 whereas channel 5 runs with lower frequency. This leads to `Channel_Stim`= (binary 0001 0011) = 19 and `Channel_Lf`= (binary 0001 0000) = 16. The checksum is given as 98+7+1+19+16 modulo 8 = 5 (binary 101). In binary format, the modulo 8 operation can be easily performed by just taking the 3 LSB of the sum 98+7+1+19+16 = 141 (binary 10001101). The command byte sequence is then

Byte 1 – 10010100
Byte 2 – 01000100
Byte 3 – 01100010
Byte 4 – 0000XX00
Byte 5 – 01110000
Byte 6 – 01100010

In hex code the command is 94 44 62 0 70 62.

Channel list mode initialisation example 2

The channel list mode shall be initialised with the following parameters:
`Main_Time`=31 (binary 00000011111) gives t_{s_1}=16.5 ms
`Group_Time`=9 (binary 01001) gives t_{s_2}=6 ms
`N_Factor`=2 (binary 010)
Channels to be activated are 2,3,6 and 8 whereas the channel 2 and 3 run with lower frequency. This leads to `Channel_Stim`= (binary 1010 0110) = 166 and `Channel_Lf`= (binary 0000 0110) = 6. The checksum is given as 31+9+166+6+2 modulo 8 = 6 (binary 110). In binary format, the modulo 8 operation can be easily performed by just taking the 3 LSB of the sum 31+9+166+6+2 = 141 (binary 11010110). The command byte sequence is then

Byte 1 – 10011001
Byte 2 – 00101001
Byte 3 – 01000000
Byte 4 – 0110XX01
byte 5 – 00010000
byte 6 – 00011111

In hex code the command is 99 29 40 61 10 1F.

Channel list mode update example

This example is for the initialisation example 2 above.

Pulse width channel 2: 100 (001100100)
Pulse width channel 3: 200 (011001000)
Pulse width channel 6: 300 (100101100)
Pulse width channel 8: 400 (110010000)

Current amplitude channel 2: 52 (0110100)

Current amplitude channel 3: 55 (0110111)
Current amplitude channel 6: 72 (1001000)
Current amplitude channel 8: 92 (1011100)

Mode channel 2: 0 (00)
Mode channel 3: 2 (10)
Mode channel 6: 1 (01)
Mode channel 8: 1 (01)

The checksum is given by 100+200+300+400+52+55+72+92+0+2+1+1 modulu 32 = 1+2+8+16 = 28 (binary 11011).

In binary format, the modulo 32 operation can be easily performed by just taking the 5 LSB of the sum 100+200+300+400+52+55+72+92+0+2+1+1 = 1275 (10011111011).

The command byte sequence is then

Byte 0 – 10111011
Byte 1 – 000XXX00
Byte 2 – 01100100
Byte 3 —00110100
Byte 4 – 010XXX01
Byte 5 – 01001000
Byte 6 – 00110111
Byte 7 – 001XXX10
Byte 8 – 00101100
Byte 9 – 01001000
Byte 10 – 001XXX11
Byte 11 – 0010000
Byte 12 – 01011100

In hex code the command is BB 00 64 34 41 48 37 22 2C 48 23 10 5C.