A. REMOTE CONTROL

The **USB 1.1** interface is the serial one working with 12 MHz clock. Its speed is relatively high and it ensures the common usage of USB in all produced nowadays Personal Computers.

The functions which are developed in order to control data flow in the serial interface ensure:

- bi-directional data transmission,
- remote control of the instrument.

The user, in order to programme the serial interface, has to:

1. send "the function code",
2. send an appropriate data file

or

3. receive a data file.

A.1. Input / output transmission types

The following basic input / output transmission types (called functions) are available:

#1 input / output of the control setting codes,
#2 output of the measurement data in the vibration level meter (**VLM**) mode,
#3 output of the measurement data in **1/1 OCTAVE** or **1/3 OCTAVE** mode,
#4 read out the data file from the internal Flash-disc and/or the special file located in the RAM memory,
#6 remote setting of the user filters,
#7 special control functions,
#9 send the setup file to the internal Flash-disc.

A.2. Function #1 – input / output of the control setting codes

Function #1 enables the user to send the control setting codes to the instrument and read out a file of the current control state. A list of the control setting codes is given in Tab. A.1.

The format of **#1** is defined as follows:

#1,Xccc,Xccc,(...),Xccc;

or

#1,Xccc,X?,Xccc,(...),X?,Xccc;

where:

X - the group code, **ccc** - the code value,
X? - the request to send the current X code setting.

The instrument will output a control settings file for all requests **X?** in the following format:

#1,Xccc,Xccc,(...),Xccc;

In order to read out all current control settings the user should send to the device the following characters:

#1;

The instrument will output a control settings file in the format:

#1,Xccc,Xccc,(...),Xccc;

Example: The following sequence of characters:

#1,U106,N4000,Z0:1,Z0:2,Z0:3,Z0:4,Z0:5,Z0:6,M3,Y1000,Xa1,Xv1,Xd1,XA0,XR0,S0;

means that:

- the **SV 106** is investigated (U106),
- the unit's number is **4000** (N4000),
- the **Vibration Level Mode** is selected in channel 1 (Z0:1),
- the **Vibration Level Mode** is selected in channel 2 (Z0:2),
- the **Vibration Level Mode** is selected in channel 3 (Z0:3),
- the **Vibration Level Mode** is selected in channel 4 (Z0:4),
- the **Vibration Level Mode** is selected in channel 5 (Z0:5),
- the **Vibration Level Mode** is selected in channel 6 (Z0:6),
- the **1/3 OCTAVE analyser** function is selected (M3),
- the measurement start delay is equal to **1000** milliseconds (Y1000),
- the reference level for acceleration measurement is set to **1 µms^{-2}** (Xa1),
- the reference level for velocity measurement is set to **1 nms^{-1}** (Xv1),
- the reference level for displacement measurement is set to **1 pm** (Xd1),
- the AutoSave option is switched off (XA0),
- the RAM file will not be created (XR0),
- the instrument is in the **STOP** state (S0).

Note: *All bytes of that transmission are ASCII characters.*

Note: Any setting can be changed only when the instrument is in the STOP state (S0).

A.3. Function #2 – read-out of the measurement results in the VLM mode

Function **#2** enables one to read out the current measurement data in the **VLM** Mode.

Notice: *This function can also be programmed while measurements are taking place. In this case, the RMS values measured* ***after entering #2 function*** *will be sent out.*

#2 function has a format defined as follows:

#2,p,X?,X?,X?,(...),X?;

where:

X - the code of the result,
p - the number of the results set
-1,-2 – for reading vibration dose results for channels 1-3 and 4-6
13,14 - for reading vector results for channels 1-3 and 4-6
1,2,3,..,12 – for reading profile results
(calculated from the formulae: *ChannelNumber + 6 * (ProfileNumber - 1)*

Notice: *After entering the* ***STOP*** *condition, #2 function is no longer active and has to be reprogrammed in order to read-out successive measurements.*

The instrument will send the values of the results in the format defined as follows:

#2,p,Xccc,Xccc,Xccc,(...),Xccc; (where **p** - the number of the results set)

or

#2,?; (when the results are not available).

The codes of the results in the case of **VLM** mode are defined as follows:

V the overload flag (ccc equals to 0 or 1);
T time of the measurement (ccc – value in seconds);
P the **P–P** value (ccc – the value in dB);
Q the **PEAK** value (ccc – the value in dB);
M the **MTVV** value (ccc – the value in dB);
R the **RMS** value (ccc – the value in dB);
H the **VDV** value (ccc – the value in dB);
v the underrange value (ccc – the value in dB).

Example: After sending to the instrument the string:

#2,1,T?,V?,P?,R?;

one should receive the following answer:

#2,1,T3,V0,P76.92,R64.50;

The codes of the results in the case of **Vibration Dose** mode are defined as follows:

a the **Current Dose** value (ccc – the value in dB);
b the **Daily Dose value** (ccc – the value in dB);
c the **Current Exposure** value (ccc – the value in dB);
f the **Daily Exposure** value (ccc – the value in dB);
g the **EAV Time** value (ccc – value in seconds);
h time left to reach **EAV** value (ccc – value in seconds);
i the **ELV Time** value (ccc – value in seconds);
j time left to reach **ELV** value (ccc – value in seconds).

Example: After sending to the instrument the string:

#2,-1,c?,f?,g?,h?;

one should receive the following answer:

#2,-1,c-27.89,f-13.44,g172800,h172800,i172800,j172800;

The codes of the results in the case of **Vector** mode are defined as follows:

P the **PPV** value if vector type is set to PPV (ccc – value in dB).
M the **MTVV** value if vector type is set to MTVV (ccc – value in dB).
R the **RMS** value if vector type is set to RMS (ccc – value in dB).

Notice: *All bytes of that transmission are ASCII characters.*

A.4. Function #3 – read-out of the measurement results in 1/1 OCTAVE and 1/3 OCTAVE mode

Function **#3** enables one to read out the current measurement data in **1/1 OCTAVE**, **1/3 OCTAVE**.

#3 function format is defined as follows:

#3,n;

where: **n** – the number of channel (1, 2, 3, 4, 5, or 6)

The device will respond, sending the last measured spectrum (when in STOP state) or currently measured spectrum (in RUN state) in the following format:

#3,n;<Status Byte> <LSB of the transmission counter> <MSB of the transmission counter> <data byte> (...) <data byte>

Status Byte gives the information about the current state of the instrument.

D7	D6	D5	D4	D3	D2	D1	D0

where:

D7	= 1	denotes "overload indicator",
D6	= 1	denotes "averaged spectrum",
D5	= 0	the instantaneous current result (RUN State),
	= 1	the final result (STOP State),
D0 to D4		reserved bits.

Note: *The measurement result is coded in binary form as dB•100 (e.g. 34.5 dB is sent as binary number 3450).*

A.5. Function #4 – read-out of the data file from the internal flash-disc and/or the special file located in the RAM memory

Function **#4** enables the user to read-out the data file from the internal Flash-disc memory. The data file formats are given in Appendix B.

#4 function formats are defined as follows:

#4,0,\;	the file containing the catalogue,
#4,1,FILE NAME;	the file containing the measurement results or saved setup,
#4,1,FILE NAME,addr;	the file containing the measurement results or saved setup,
#4,2,Bnnn;	the file containing logger,
#4,3;	the special file contained in the RAM memory (**RAMfile**),

where:

FILE NAME	not longer than eight-character name,
addr	is the logical address of the file in the internal Flash-disc memory,
nnn	the number of the logger file (one or more digits - depends on requirements).
RAMfile	the special name for the file contained in the RAM memory, may be used also with the format: **#4,1,RAMfile;**

Notice: The "\" character is the obligatory catalogue file name *(it must be sent to the instrument).*

The device will respond sending the specified file/catalogue in the following format:

#4,k;<4 bytes giving the file size (in binary form)><data byte>...<data byte>

where character **k** corresponds to the file type:

0 for the file containing the catalogue,
1 for the file containing the measurement results or saved setup,
2 for the file containing the logger file.

All data words are sent as **<LSB>**,**<MSB>**.

When an error is detected in the file specification or data, the instrument will send: **#4,?;**

The catalogue of the files is a set of the records containing 16 words (16 bits each). Each record describes one file saved in the instrument's Flash-disc. The record structure is as follows:

words 0 - 3 8 character file name,
word 4 file type (binary number),
word 5 reserved,
word 6 least significant word of the file size,
word 7 most significant word of the file size,
word 8 least significant word of the file logical address,
word 9 most significant word of the file logical address,
word 10 measurement start date,
word 11 measurement start time,
words 12 - 15 reserved.

For logger and the RAMfile the **logical address** is always set to 0.

For files containing saved setup measurement start date and time are always set to 0.

Notice: *If the* ***DEFRAGMENTATION*** *function is performed after the read out of the files catalogue the logical addresses of the files could be wrong.*

The measurement **start date** is coded as a word with bits:

b15 ... b3 b2 b1 b0

where:

b15 b14 b13 b12 b11 b10 b9 is a year minus 2000.
b8 b7 b6 b5 is a month (1..12),
b4 b3 b2 b1 b0 is a day (1..31).

The measurement **start time** is coded as number of seconds counted from 00:00:00 divided by 2.

The structure of the files containing the measurement results, saved setups and/or logger files is described in details in Appendix B.

A.7. Function #6 – remote setting of the user filters

Function **#6** enables one to send to the instrument the coefficients of the user filters. In the available formats description of **#6** functions the following symbols are used:

type	- 0 for the vibration filters,
	- 1 for the acoustic filters,
name, **name$_1$**, **name$_2$**	- filter names given by the user,
v	- real type value, expressed in [dB],
first	- integer type value (number of the coefficient in the user filter),
pos	- integer type value (Total value number),
avd	- for the vibration filters: 0 - Acc, 1- Vel, 2 - Dil,
	- for the acoustic filters this parameter is always equal to 0,
cal	- the calibration coefficient given as the real number expressed in [dB].
chn	- channel number (1, 2, 3 or 4).

#6 function formats are defined as follows:

#6,type,L;

This function returns the list of the defined (existing in the instrument) filters in the following format: **#6,type,n,name$_1$, ... ,name$_n$;**

#6,type,W,name,v,v,...,v;

This function sets the coefficients of the new user filter named as **name**. The **name** parameter should be unique (in the instrument there is not any other filter with the same name, otherwise it will be an error). The function answers in the format: **#6;**

#6,type,R,name;

This function returns the coefficients of the user filter named as **name**. If the **name** filter does not exist, an error occurs. The function returns in the following format: **#6,type,n,v$_1$,v$_2$, ... ,v$_n$;**

#6,type,D,name;

This function deletes from the instrument the user filter named as **name**. If the **name** filter does not exist, an error occurs. The function answers in the format: **#6;**

#6,type,S,name,v,v,...,v;

This function sets the user filter named as **name**. If the **name** filter already exists, its coefficients are redefined. If the **name** filter does not exist, the filter is created. The function answers in the format: **#6;**

#6,type,C,name,first,v,v,...,v;

This function sets the coefficients in the user filter named as name starting from the first position. If the **name** filter does not exist, an error occurs. The function answers in the format: **#6;**

#6,type,N, name$_1$, name$_2$;

This function changes the name of the user filter from **name$_1$** to **name$_2$**. The function answers in the format: **#6;**

#6,type,@,chn,L;

This function returns the names of the user filters, assigned to the channel **chn** consecutive **TOTAL** values, in the following format: **#6,type,chn,3,name$_1$,name$_2$,name$_3$;**

#6,type,@,chn,pos,?;

This function returns the description record of the user filter assigned to the **pos TOTAL** value of channel **chn** in the following format: **#6,type,@,chn,pos,name,avd,cal;** (the description record contains: the name of the filter, its type and the calibration coefficient).

#6,type,@,chn,pos,*;

This function recovers the predefined filter for the **pos TOTAL** value of channel **chn** and returns the following format: **#6,type,@,chn,pos,name,avd,cal;**

#6,type,@,chn,pos,name,avd,cal;

This function sets the description record of the user filter assigned to the **pos TOTAL** value of channel **chn** in the following format: **#6,type,@,chn,pos,name,avd,cal;**

The returned parameters: **name**, **avd** and **cal** are set in the description record after the execution of the function. In the case of an error they can differ from the current parameters of the function.

Notice: *In the case of an error all these functions return the following sequence of the characters:* ***#6?;***

A.8. Function #7 – special control functions

Function **#7** enables the user to perform special control functions. **Some of them should be used with the extreme care.**

#7 function formats are defined as follows:

#7,CB;

This function deletes all logger files in current directory on SD card. The function returns **#7,CB;** This function is not accepted while the instrument is in the RUN state.

#7,BF;

This function returns free space in the format:
#7,BF,ddddd; (**ddddd** - number of bytes in decimal format).

#7,BN;

This function returns the number of logger files created to the current time in the format: **#7,BN,ddddd;** (**ddddd** - number of logger files in decimal format).

#7,RT;

This function returns current real time clock settings in the format:
#7,RT,hh,mm,ss,DD,MM,YYYY;
where **hh:mm:ss** denotes the time and **DD**/**MM**/**YYYY** gives the date.

#7,RT,hh,mm,ss,DD,MM,YYYY;

This function sets the current real time clock and returns the following sequence of characters: **#7,RT;**

#7,AS;

This function returns current real time and date settings for the AutoStart function in the format: **#7,AS,e,hh,mm,ss,DD;** where e=1 if AutoStart function is switched ON or 0 if it is switched OFF, **hh:mm:ss** gives the time and **DD** gives the day for the current date.

#7,AS,e,hh,mm,DD;

This function uses the given time and date settings for AutoStart function and returns the following sequence of characters: **#7,AS;**

#7,SS;

This function saves the current settings of the instrument in the EEPROM memory. The function returns the following sequence of characters: **#7,SS;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DA;

This function deletes all files containing measurement results and instrument's settings from the current directory. The function returns the following sequence of characters: **#7,DA;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DF;

This function deletes all files containing measurement results from current directory. The function returns the following sequence of characters: **#7,DF;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DF,fileName;

This function deletes file named **fileName** containing measurement results. The function returns the following sequence of characters: **#7,DF;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DF,fileName<iAddr;

This function deletes file located at internal address **iAddr** containing measurement results from the internal flash memory. The function returns the following sequence of characters: **#7,DF;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DS;

This function deletes all files containing instrument's settings from the internal flash memory. The function returns the following sequence of characters: **#7,DS;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DS,fileName;

This function deletes file named **fileName** containing instrument's settings from the internal flash memory. The function returns the following sequence of characters: **#7,DS;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,DS,fileName<iAddr;

This function deletes file containing instrument's settings located at internal address **iAddr** from the internal flash memory. The function returns the following sequence of characters: **#7,DS;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,AN,FName;

This function sets the name of the file for the Autosave function as the **FName**. The given name has to start with the '@' character and contain no more than 8 characters. The function returns the following sequence of characters: **#7,AN;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,AN;

This function returns current file name used by Autosave function in the format: **#7,AN,FName;**.
This function is not accepted and not performed while the instrument is in the RUN state.

#7,AV;

This function returns analyser firmware version in the format **#7,AV,XX.XX.XXC;** where XX.XX.XX is firmware version, C – firmware subversion.

#7,US;

This function returns unit subtype in the format **#7,US,XX;** where XX is subtype number.

#7,AL,?;

This function returns activated alarms list in the format: **#7,AL,XX,XX,...,XX;** where XX is alarm identifier.

#7,AL,XX;

This function returns SMS message text for activated alarm, where XX is alarm identifier.

#7,AL,R;

This function disables all alarm conditions and returns **#7,AL,R1;**

#7,LB;

This function returns current file name used for logging in the format: **#7,LB,FName;**.

#7,UH;

This function returns device selected for file storing in the format: **#7,UH,XX;**, where XX can be one of: 0 - internal memory, 2 - SD card.

#7,UH,XX;

This function sets the device selected for file storing, where XX can be one of: 0 - internal memory, 2 - SD card. Function returns **#7,UH,XX;** where XX is selected device.
This function is not accepted and not performed while the instrument is in the RUN state.

#7,RC,?;

This function returns state of remote control mode in the format: **#7,RC,x;** where x can be 0 (disabled) or 1 (enabled).

#7,RC,x;

This function sets state of remote control mode to disabled in case X equals 0 or enabled otherwise. Function returns **#7,RC;** upon success.

#7,CS;

This function loads factory settings.
The function returns the following sequence of characters: **#7,CS;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,PO;

This function switches the instrument off.
The function returns the following sequence of characters: **#7,PO;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,LT;

This function reloads transducer parameters from TEDS.
The function returns the following sequence of characters: **#7,LT;**
This function is not accepted and not performed while the instrument is in the RUN state.

#7,IM,?;

This function returns instrument mode in the format **#7,IM,X;** where X equal to 0 means Simple mode and Advanced otherwise

#7,IM,X;

This function sets instrument mode to Simple if X equals 0, and to Advanced if X equals 1.
Function returns the following sequence of characters: **#7,IM,X;**, where X is current instrument mode.
This function is not accepted and not performed while the instrument is in the RUN state.

#7,BS;

This function returns battery charge level in the format **#7,BS,nn;** where nn is a percent value. When battery state is not available (i.e. unit is powered from external source) function returns **#7,BS,-1;**.

Notice: *For the unknown function and/or in the case of the other error, all these functions return the following sequence of characters:* ***#7,?;***

A.9. Function #9 – writing setup files to the internal flash-disc

Function #9 allows uploading files containing instrument setup to the internal Flash-disc. The function expects files in format described in Appendix B, paragraph B.9. **Function should be used with extreme care.**

The #9 function format is defined as follows:

#9,2,Len,<data byte> ... <data byte>

where:

Len - length of transferred file in bytes as ASCII,

<data byte> - byte of data in binary form.

Function responds with "**#9,1;**" on success and with "**#9,0;**" on failure.

A.10. Control setting codes

The control setting codes used in the **SV 106** instrument (starting from the internal software version 3.21.6) are given in the table below.

Table A.1. Control setting codes

Group name	Group code	Code description
Unit type	**U**	U106 (read only)
Serial number	**N**	Nxxxx (read only)
Software version number * 100	**W**	Wxxx xxx - Analyzer version number * 100 (read only)
Channel mode	**Z**	Z0:n - Vibration LM / Analyzer for channel n

Calibration factor	**Q**	Qnnnn:c nnnn - real number with the value of the calibration factor for channel c in dB ∈ (-99.9 ÷ 99.9)
Measurement function	**M**	M1 - **Level Meter** M2 - **1/1 OCTAVE analyser** M3 - **1/3 OCTAVE analyser**
Execution of **1/1 OCTAVE** or **1/3 OCTAVE** analysis in channel n	**e**	e0:n - Spectrum analysis in channel n disabled e1:n - Spectrum analysis in channel n enabled
Range of channel n	**R**	R1:n - **316 ms^{-2}** (**VLM**)
Filter type in **1/1 OCTAVE** or **1/3 OCTAVE** analysis in channel n for **VLM**	**i**	i0:n - **HP** filter in channel n (read only)
Filter type in profile for **VLM**	**I**	I0:m **HP** filter for profile m I5:m **Vel3** filter for profile m I16:m **Wk** filter for profile m I17:m **Wd** filter for profile m I18:m **Wc** filter for profile m I19:m **Wj** filter for profile m I20:m **Wm** filter for profile m I21:m **Wh** filter for profile m I22:m **Wg** filter for profile m I23:m **Wb** filter for profile m I24:m **Wf** filter for profile m I118:m **BL Wc** filter for profile m I119:m **BL Wj** filter for profile m I120:m **BL Wm** filter for profile m I121:m **BL Wh** filter for profile m I122:m **BL Wg** filter for profile m I123:m **BL Wb** filter for profile m I124:m **BL Wf** filter for profile m m = ChannelNo + 6 * (ProfileNo - 1)
Detector type in profile for **VLM**	**E**	E4:m - **1 s** detector in profile m m = ChannelNo + 6 * (ProfileNo - 1)
Logger type in profile in the case of **VLM**	**G**	G0:m - **None** logger in profile Gxx:m - xx - sum of values for profile m: 1 – logger with **PEAK** values 2 – logger with **P–P** values 4 – logger with **MAX** values 8 – logger with **RMS** values 16 – logger with **VDV** values m = ChannelNo + 6 * (ProfileNo - 1)
Storing the results of **1/1 OCTAVE** or **1/3 OCTAVE** analysis in channel n in logger file	**g**	g0:n - switched off (**none**) in channel n g4:n - switched on (**RMS**) in channel n
Logger time step	**d**	dnnnn - nnnn number in milliseconds ∈ (**100**, **200**, **500**, **1000**) dnns - nn number in seconds ∈ (1 ÷ 60) dnnm - nn number in minutes ∈ (1 ÷ 60)
Integration time	**D**	D0 “infinite” number Dnns nn number in seconds Dnnm nn number in minutes Dnnh nn number in hours

Repetition cycle	**K**	K0 - infinity (measurement stopped when the STOP button is pressed or when remote setting S0 is received) Knnnn -nnnn number of repetitions ∈ (1 ÷ 1000)
Detector type in the **LEQ** (for **SLM**) and/or **RMS** (for **VLM**) function	**L**	L0 - **LINEAR**
Measurement Trigger Mode (TriggerMode)	**m**	m0 - **OFF** m1 - **SLOPE +** m2 - **SLOPE –** m3 - **LEVEL +** m4 - **LEVEL –** m5 - reserved m6 - **GRAD +** m7 - **RTC**
Source of the triggering signal for measurement functions: M1 and M6 (TriggerSource)	**s**	s0 - Vector 1-3 value s1 - Vector 4-6 value s2 - reserved s3 - **RMS** value from profile 1 s4 - External trigger
Channel of the triggering signal	**c**	c1 - channel 1 c2 - channel 2 c3 - channel 3 c4 - channel 4 c5 - channel 5 c6 - channel 6
Source of the triggering signal for measurement function	**o**	o0 - Vector 1-3 value o1 - Vector 4-6 value o2 - reserved o3 - **RMS** value from profile 1 o4 - External trigger
Source of the triggering signal for measurement function M3	**t**	t0 - Vector 1-3 value t1 - Vector 4-6 value t2 - reserved t3 - **RMS** value from profile 1 t4 - External trigger
VLM's trigger level (TriggerLev)	**n**	nxxx - xxx level given in dB ∈ (60 ÷ 200)
VLM's vector trigger level (VecTriggerLev)	**h**	hxxx - xxx level given in dB ∈ (60 ÷ 200)
Number of the records from the logger taken into account before the fulfilment of the triggering condition (TriggerPre)	**p**	pnn - nn number of the records taken into account before the fulfilment of the triggering condition ∈ (0 ÷ 20)
Number of the records from the logger taken into account after the fulfilment of the triggering condition (TriggerPost)	**q**	qnn - number of the records taken into account after the fulfilment of the triggering condition ∈ (0 ÷ 200)
Delay in the start of measurement	**Y**	Ynn nn delay given in milliseconds ∈ (0 ÷ 60000)
Reference level for acceleration (RefLev_a)	**Xa**	Xannn nnn reference level for acceleration given in μms^{-2} ∈ (1 ÷ 100)
Reference level for velocity (RefLev_v)	**Xv**	Xvnnn nnn reference level for velocity given in nms^{-1} ∈ (1 ÷ 100)
Reference level for displacement (RefLev_d)	**Xd**	Xdnnn nnn reference level for displacement given in pm ∈ (1 ÷ 100)

AutoSave option	**XA**	XA0 - switched OFF XA1 - switched ON, file names are numbered
Using the RAMfile instead of the flash disk while storing results with the AutoSave option switched on	**XR**	XR0 - switched OFF XR1 - switched ON
Extended I/O Mode	**x**	x0 - **AC/Int.** in **Analogue** mode x1 - **AC/Int.** in **Digital In** mode x2 - **AC/Int.** in **Digital Out** mode
External I/O Channel for analogue **AC/Int.** mode	**y**	yn - n - channel number between 1 and 6
State of the instrument (Stop or Start)	**S**	S0 - **STOP** S1 - **START**
Menu lock mode	**Xb**	Xb0 - menu unlocked Xb1 - menu partially locked Xb2 - menu fully locked
Vector calculation mode	**XB**	XB0:n - switched OFF XB1:n - HAV XB2:n - WBV XB3:n - user defined XB3:n - MTVV XB4:n - PPV n - 1 for channels 1-3, 2 for channels 4-6
Channel coefficient for vector calculation	**XC**	XCxx:k:n - xx - value of coefficient *100 $\in (0 \div 200)$ - n - channel number - k: 1 - vector 1-3 2 - vector 4-6
Storing vector in logger file	**XD**	XD0:k - switched OFF XD8:k - switched ON k: 1 - channels 1-3 2 - channels 4-6
Type of vibration dose	**XE**	XE1:k - HAV XE2:k - WBV k: 1 - channels 1-3 2 - channels 4-6
Vibration dose exposure time	**XF**	XFnn nn - time in minutes $\in (0 \div 1440)$
Vibration dose standard	**XG**	XG0 - Great Britain XG1 - Italy XG2 - Poland XG3 - French XG4 - user defined XG5 - German XG6 - China
X axis channels	**XH**	XHN:1 - N - x axis channel for channels 1-3 XHN:2 - N - x axis channel for channels 4-6
Y axis channels	**XI**	XIN:1 - N - y axis channel for channels 1-3 XIN:2 - N - y axis channel for channels 4-6
Z axis channels	**XJ**	XJN:1 - N - z axis channel for channels 1-3 XJN:2 - N - z axis channel for channels 4-6

Spectrum **MAX** store	**XT**	XT0 spectrum **MAX** switched OFF XT1 spectrum **MAX** switched ON
Spectrum **MIN** store	**Xt**	Xt0 spectrum **MIN** switched OFF Xt1 spectrum **MIN** switched ON
Trigger gradient level for **VLM**	**Xh**	Xgnn - nn – gradient level in dB/ms ∈ [1,100]
RTC trigger start time	**Xr**	Xrnn - nn – time in seconds ∈ [0,86399]
RTC trigger step time	**Xs**	Xs0 - use integration time for step Xsnn - nn – step in seconds ∈ [1,86400]
Function for Digital In **AC/Int.** mode	**XP**	XP0 - trigger pulse
Function for Digital Out **AC/Int.** mode	**XQ**	XQ0 - trigger pulse XQ1 - alarm pulse
AC/Int. polarization	**XU**	XU0 - positive XU1 - negative
AC/Int. active level	**XV**	XV0 - active low XV1 - active high
Logger writing trigger	**XXk**	XXk0 - switched OFF XXk1 - switched ON
Wave writing trigger	**XXl**	XXl0 - switched OFF XXl1 - switched ON
Logging mode	**XXm**	XXm0 - switched OFF XXm1 - logger XXm2 - wave
Channel input type	**XXn**	XXn0:P - direct XXn1:P - IEPE XXn2:P - building direct XXn3:P - building IEPE P - channel number
Measurement trigger	**XXu**	XXu0 - switched OFF XXu1 - switched ON
Dosimeter enable	**XXv**	XXv0 - switched OFF XXv1 - switched ON
Noise compensation	**XXK**	XXK0 - switched OFF XXK1 - switched ON
Simple trigger mode	**XXXi**	XXXi0:K - OFF XXXi1:K - LEVEL - XXXi2:K - LEVEL + XXXi3:K - SLOPE - XXXi4:K - SLOPE + XXXi5:K - GRADIENT – XXXi6:K - GRADIENT + K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger

Simple trigger integration period	**XXXj**	XXXj0:K - logger step XXXj1:K - 100 ms XXXj2:K - 1 s XXXj3:K - elapsed integration time XXXj4:K - integration period K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger
Simple trigger source	**XXXk**	XXXk0:K:M - Vector XXXk1:K:M - PEAK XXXk2:K:M - P–P XXXk3:K:M - MAX XXXk4:K:M - RMS XXXk5:K:M - VDV XXXk6:K:M - first spectrum bar ... XXXk21:K:M - last 1/1 Octave spectrum bar XXXk22:K:M - first 1/1 Octave total XXXk23:K:M - second 1/1 Octave total XXXk24:K:M - third 1/1 Octave total XXXk44:K:M - last 1/3 Octave spectrum bar XXXk45:K:M - first 1/3 Octave total XXXk46:K:M - second 1/3 Octave total XXXk47:K:M - third 1/3 Octave total K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger M: simple trigger source type (cf. XXXm)
Simple trigger level	**XXXl**	XXXlN:K - N – level in dB*10 K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger
Simple trigger source type	**XXXm**	XXXm0:K - Vector XXXm1:K - Profile XXXm2:K - Spectrum K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger
Simple trigger source channel	**XXXp**	XXXpN:K - N - channel K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger

Simple trigger source channels	**XXXq**	XXXq0:K - Channels 1-3 XXXq1:K - Channels 4-6 K: simple trigger identifier. One of: 0 - alarm trigger 1 - logger trigger 2 - wave trigger 5 - event trigger
Hand-Arm EAV User limit	**XXXr**	XXXrN - N – limit value*100
Hand-Arm ELV User limit	**XXXs**	XXXsN - N – limit value*100
Whole Body EAV User limit	**XXXt**	XXXtN:P - N – limit value*100 P - axis number 1 - X axis 2 - Y axis 3 - Z axis
Whole Body ELV User limit	**XXXu**	XXXuN:P - N – limit value*100 P - axis number 1 - X axis 2 - Y axis 3 - Z axis
vibration dosimeter user unit type	**XXXv**	XXXv0:P - m/s^2 XXXv1:P - $m/s^{1.75}$ P - limit index 0 - H-A EAV 1 - H-A ELV 2 - WBV EAV 3 - WBV ELV
enabled channels	**XXXw**	XXXw0 - all channels XXXw1 - channels 1-3 XXXw2 - channels 4-6
enable 2nd profile	**XXXx**	XXXx0 - 2nd profiles disabled XXXx1 - 2nd profiles enabled
interface language	**XXXy**	XXXy0 - english XXXy1 - polish XXXy2 - italian XXXy3 - russian XXXy7 - german