



Sensor Simulator MSS-1010

rev. 3.0



Caution:

Please read carefully this manual before operating the MSS-1010 sensor simulator

Contents:

1. Product Description.....	4
2. MSS-1010 Map and legend.....	5
3. Using MSS-1010 Sensor Simulator	6
3.1 Root Menu	6
3.2 Voltage Output (mV Voltage Generator Mode)	6
3.3 IEPE (ICP [®]) Output – Current Sinking.....	7
3.4 Charge output (pC).....	10
3.5 Speed and OSO [®] Output	12
3.6 BOV Sensor Input.....	13
3.7 Settings.....	14
4. PC connection	15
5. Charging connector	17
6. Device Dimensions	18

1. Product Description

Sensor simulator MSS-1010 is a battery powered instrument suitable for simulation of various kinds of sensors. Access to different simulations is done through a menu shown on the sensor simulator screen. The navigation through the menu is done by keys on the device. The sensor simulator is powered by four rechargeable Ni-MH batteries, which are charged by external regulated power supply at 9Vdc. It also has an USB connection for controlling a device over PC cable connection.

The device is suitable for inspecting measurement lines according to ISO 10816 standards. It is especially designed for Condition Monitoring Systems (CMS) and/or Vibration Monitoring Systems (VMS) which connect to SCADA Systems.

More detailed specifications of sensor simulator are given below.

Outputs

- Single-ended voltage (mV)
- IEPE (ICP®) – current sinking
- Single-ended and differential charge (pC)
- Tachometer (TTL)
- OSO® - Optical Speed Output
- BOV – Bias Output Voltage

Frequency range	1Hz to 1kHz, accuracy $\pm 1\%$
Amplitude	Adjustable up to 10V RMS
Distortion	<1%, from 10Hz to 1kHz, 10mV to 10V <1%, from 2Hz to 1kHz, 10mV to 10V

Transfer Characteristics

Amplitude accuracy	$\pm 1\%$ of settings on any range
Frequency stability	$\pm 1\%$ of settings on any range
Harmonic distortion	1% maximum, from 10Hz to 1kHz

Environmental Characteristics

Temperature	
Operating from	-10°C to +65°C
Storage from	-18°C to +65°C
Humidity	95% R.H. maximum

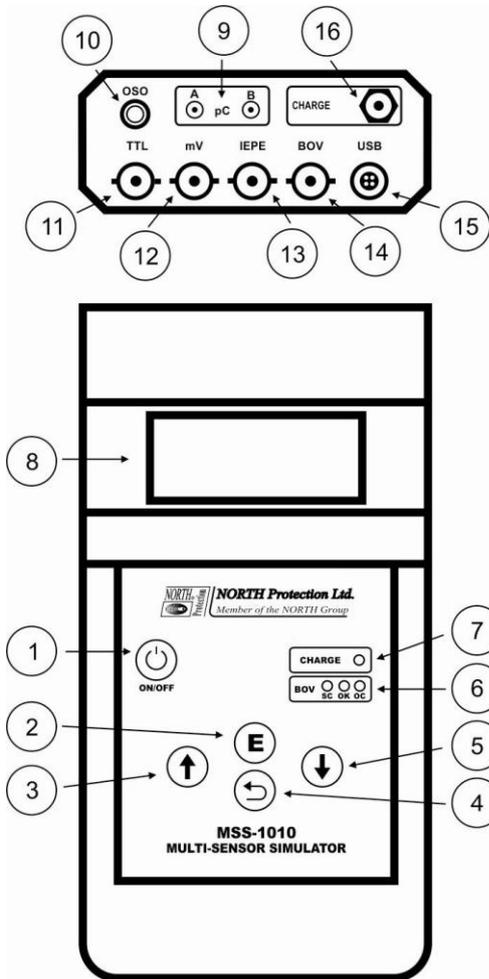
Power

Battery	4 x AA rechargeable Ni-MH supplied
Autonomy	More than 5 hours when fully charged

Physical Characteristics

Dimension	196mm x 100mm x 40mm
Weight	0.5kg typical
Case	ABS plastic
Connection	mV , IEPE, BOV and T achometer - standard BNCs Charge (single-ended and differential) - Microdots USB - LEMO
Front panel controls	Five sealed switches (arrows, E, Back and ON/OFF button)
Front panel display	4 line LCD panel with 64 character

2. MSS-1010 Map and legend



LEGEND

- 1 - Power button, ON/OFF switch
- 2 - ENTER button, use for entering and confirmation in menu
- 3 - UP arrow, use for navigation through menu
- 4 - BACK button, use for exiting menu
- 5 - DOWN arrow, use for navigation through menu
- 6 - BOV test semaphore
- 7 - Charging LED diode
- 8 - LCD display
- 9 - microDot connectors for charge output
- 10 - OSO[®] optical speed output
- 11 - BNC connector for TTL output
- 12 - BNC connector for mV output
- 13 - BNC connector for IEPE output
- 14 - BNC connector for BOV testing
- 15 - LEMO connector for USB connection
- 16 - Charging connector for external charger

3. Using MSS-1010 Sensor Simulator

Management of the functions of the device is done by the front panel, that is by the keyboard. The keyboard consists of four keys and one LED. The key for switching device On/Off is at the top and it is to be pressed in order to switch the device On/Off. LED represents a light indication and it lights when the device is connected to the external source of power supply, that is a charger at 9Vdc.

Four keys in the middle are used for navigating the menu and for choosing functions we want to use. The arrows are used for moving up and down through the menu, as well as for changing the given size value, while we use the key **E** (**Enter**) and for confirmation and for Entering and Exiting the selected menu item.

3.1 Root Menu

The Root Menu consists of seven items and each represents a unit for itself. Below is a manual for using each Operating Mode, as well as for using a simulator for each Mode. Navigation through the Menu is done by arrows keys \downarrow and \uparrow , whereas choosing of the function is done by pressing the key **E** (**Enter**).

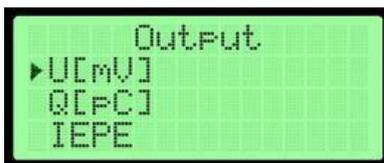


Figure 1. Root Menu – First Part



Figure 2. Root Menu – Second Part

3.2 Voltage Output (mV Voltage Generator Mode)

In this Operating Mode it is possible to set the voltage and the frequency value for a sinusoidal output signal. After choosing the mV Mode, and before it is possible to choose voltage and frequency, the message appears on the screen with a warning that the mV Output (mV BNC connector) must be used for the results to be accurate.

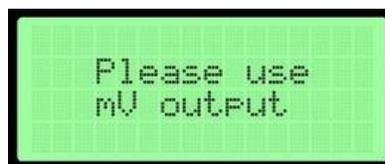


Figure 3. Message when starting the mV mode

After the warning, the submenu appears on the screen in which one can set the wanted voltage and frequency values that will be given on the way out. The screen in which these possibilities are active is given below.

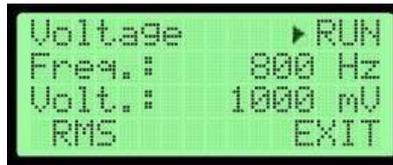


Figure 4. Voltage Generator Mode (mV mode)

By ↓ and ↑ arrows keys we navigate through options, and options are made of several functions. If we set the cursor on the position frequency, by pressing the **E** (Enter) key the arrows appear signaling that we entered the mode for changing the frequency. We use ↓ and ↑ arrows keys for setting the desired frequency value.

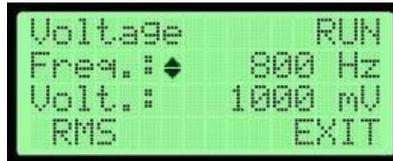


Figure 5. Change of Frequency in mV mode

When we set the appropriate frequency value by pressing the **E** key, we return to the Menu. The same happens if we want to change voltage. With **BACK** button we return the Menu too. By setting the cursor on the position Volt and by pressing the **E** key we enter the menu for changing output voltage, and the arrows for marking this mode appear on the screen, like in the case of change of frequency. If we set the cursor on the position RMS, by pressing **E** key we enter the change of signal generating mode. In this menu we can change the output voltage to generate for the given voltage value as **RMS** (Root Mean Square), **0-P** (Zero to Peak) or **P-P** (Peak To Peak) value. By setting the cursor on **EXIT** and by pressing **E** key we return to the Root Menu. We can also return to the Root Menu by pressing **BACK** button on any position in this Menu.

If we set the cursor on the command **RUN** and press the **E** key at mV output the voltage with a scale based on the given parameters appears. The voltage is of a sinusoidal shape with the given values of voltage and frequency. The indication that the device is in the **RUN** mode is that the output is generated is represented by a stop sign. By pressing the **E** key again the sensor simulator stops generating voltage and the cursor switches back to bold right arrow. This can be done by pressing **BACK** button on any cursor position too. The view of the screen while generating voltage on the output is given below.



Figure 6. Sensor Simulator in Signal Generating Mode

3.3 IEPE (ICP[®]) Output – Current Sinking

IEPE output is suitable for the simulation of sensors which on their input must have IEPE supply (DC Constant Current Source 2mA...20mA, Voltage 18V...30Vdc) in order to work. This means that for using this output modules, IEPE power supply need to be used. In this working mode it is possible to set

displacement, velocity or acceleration value, which after the command for generating turns to AC voltage of sinusoidal shape, whose scale is based on the given values (displacement, velocity or acceleration), frequency and the sensor sensitivity set in mV/g units. After selecting the IEPE mode, and before the possibility for setting parameter values, a warning appears on the screen to use IEPE output (IEPE BNC Connector).

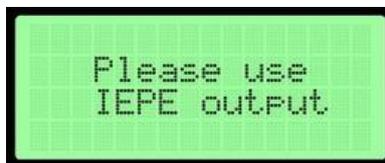


Figure 7. Message at Starting IEPE Mode

After choosing the IEPE mode, the next step is choosing the frequency at which the voltage proportional to the value of displacement, velocity or acceleration will be generated. In this part the values of sensor sensitivity can be also set, which in case of IEPE sensor can be set to 50mV/g, 100mV/g, 500mV/g and 1000mV/g.



Figure 8. First step in IEPE sensor simulation

Navigation through submenus is done by arrows keys \downarrow and \uparrow . By pressing the **E** key, we select values which we want to change. The cursor is changed to bold up/down arrows meaning that we successfully entered into value change Mode. The value is changed by the arrow keys \downarrow and \uparrow . Pressing the **E** key, we confirm the choice of the new value and return to previous menu selection. This can be done using **BACK** key too.



Figure 9. Screen for change of frequency

By setting the cursor on NEXT and by pressing the **E** key, we enter the submenu for selecting the type of values which we want to simulate. The view of the screen for selecting functions is given below. We can choose setting the values of Displacement, Velocity or Acceleration. By pressing **BACK** button we can return to Root Menu.

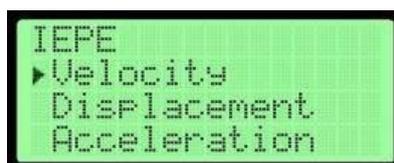


Figure 10. Screen for selecting function

By arrow keys ↓ and ↑ we navigate the menu, whereas by the **E** key we select the desired function. If for the given function, frequency and sensitivity values deviate from the maximal ie. the minimal voltage value, a message appears on the screen informing us about an Error, and the software returns us again to the possibility of setting the frequency and sensitivity values. The Error can occur only in case of Displacement and Velocity.

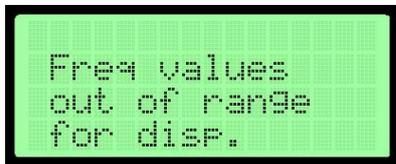


Figure 11. Error report for target displacement

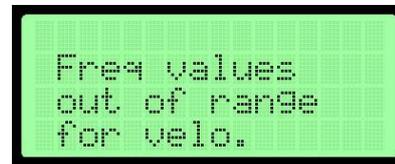


Figure 12. Error report for target velocity

Depending of the choice, if the set values are according to the mode, we reach the final part of the subprogram, where it is possible to set directly the output values of Displacement, Velocity and Acceleration. The Displacement can be expressed in RMS, 0-P or P-P values, while the values of Velocity and Acceleration can be expressed in RMS or 0-P values. Depending on the selected mode of measurement, one of the following screens will appear. In every case under the value of chosen mode there will a value of frequency that we setup in previous step.

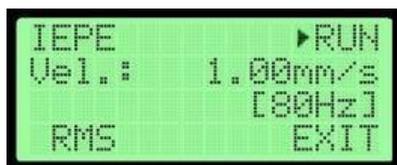


Figure 13. Displacement mode is selected

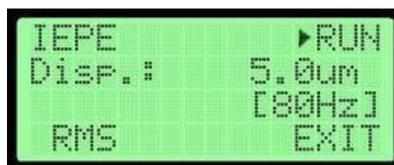


Figure 14. Acceleration mode is selected

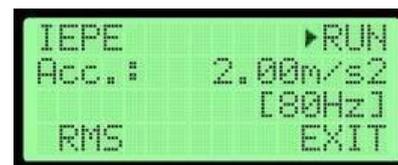


Figure 15. Velocity mode is selected

If we select Displacement, the unit we can change is **µm** for metrical units, whereas in case of imperial units we can change **mils**. Velocity values are given in **mm/s** in case the metrical system is set, whereas the **in/s** is used in case of the imperial system. For acceleration **m/s²** is used as a metrical unit, and **g** as an imperial unit. The way we can change units in which simulation is expressed will be discussed later, in section 2.7.

Each of these cases has several possibilities. If we bring cursor to the **RUN**, by pressing **E** we will start the simulation which is directly proportional to the given value of Displacement, Acceleration or Velocity. During generating of signal, the cursor will turn into a Stop sign (little bold square), representing that the Simulator is active and the AC voltage occurs on the output. The look of the screen during the active mode is given below.



Figure 16. The view of the screen when generating output

In case that any of the three parameters is activated, the cursor changes into a stop sign. By pressing again the **E** key, the generating of signal is stopped and the cursor have its previous shape. We can stop signal generating with **BACK** key too. When the cursor is set on a parameter and **E** key is pressed, we enter the change of Velocity, Acceleration or Displacement value mode, depending which type of measurement is

selected. In any case, change of parameters is done with ↓ and ↑ arrow keys. When the wanted parameter is set, by pressing **E** key, we return to the menu and the chosen value becomes valid. Then, there is the possibility of changing the way the parameter is expressed, meaning that, depending on the selected parameter, different modes can be chosen. As we have already mentioned, if the Displacement parameter is selected, it can be expressed in RMS, 0-P and P-P values, whereas for Acceleration and Velocity RMS or 0-P values are used. The mode is selected by pressing again **E** key. By choosing the function EXIT and by pressing **E** key, we return to the Root Menu. By pressing **BACK** button in this menu, regardless of chosen mode, we return to menu for changing frequency and sensitivity.

3.4 Charge output (pC)

Charge mode is suitable for a simulation sensor which on its output gives charge. When selecting this function, will see a message informing us to use output signed with **pC** in order to secure accurate and safe functioning of the equipment. The view of the screen is given below.

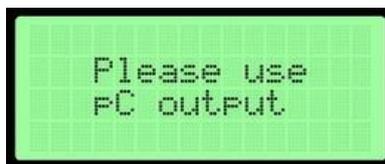


Figure 17. Message when activating pC mode

After the message, a menu will appear on the screen in which we can change frequency and sensitivity values. For sensitivity values two values are offered, 50pC/g and 100pC/g. Frequency is in range from 1Hz to 1kHz. The view of the screen is given below.

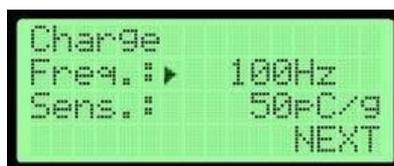


Figure 18. Selection of wanted frequency and sensitivity

The arrow keys ↓ and ↑ are used for navigating the menu, whereas with the key **E** we enter the Menu for changing values. When entering the mode for change of value instead of the cursor, two-sided arrows appear on the screen, indicating that the values can be change. The change of values is done by arrow keys ↓ and ↑. With the **E** key, the value we have set is saved and we return to the previous menu item. We can do this by pressing **Back** button too. The cursor regains its previous shape and that is a visual indication that the value is successfully saved. By pressing the **E** key, if the cursor is on NEXT, we move to selecting the parameter which we want to simulate. By pressing **BACK** button we can return to Root Menu.

As in the case of IEPE sensor simulation, Charge Sensors can simulate three parameters - Velocity, Acceleration and Displacement. In this part of the menu, we can choose which of these parameters we want to simulate. When selecting the parameter, the screen should look like this:

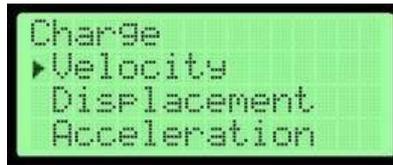


Figure 19. Choice of wanted parameter for simulation

By arrow keys ↓ and ↑, we navigate through the menu, while by the **E** key we select the wanted parameter. If the values exceed the range for the chosen parameter because of the choice of frequency and sensitivity, a message appears informing us about an Error, and the software takes us a step back in order to change frequency and sensitivity. Examples of such messages are given below:

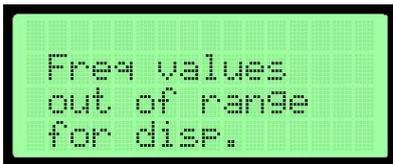


Figure 20. Error report for target displacement

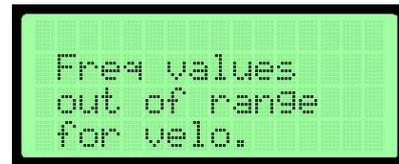


Figure 21. Error report for target velocity

If the value which we want to simulate fit into the range the device can generate, the final screen appears, in which we are able to set the final parameter. Depending of the chosen parameter, the screen can look as below:

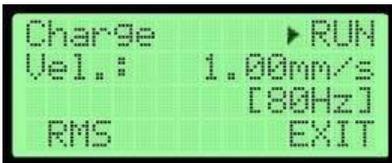


Figure 22. Screen with velocity

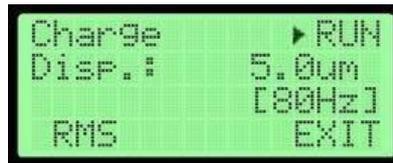


Figure 23. Screen with displacement



Figure 24. Screen with acceleration

In the case that any parameter is selected, if we mark RUN with the cursor, by pressing E, we set the generating of the signal proportional to the given parameters. If the Velocity parameter is chosen, the values are given in **mm/s** for the metrical mode, ie. in **in/s** for the imperial mode. When the Displacement parameter is selected, in the metrical mode **µm** is used, while in the imperial mode it is **mils**. For Acceleration in the metrical mode **m/s²** is used, and in the imperial mode **g**. There is also information about selected frequency in previous step.

Considering value, it can be represented in three ways for Displacement (RMS, P-P and 0-P), while for Velocity and Acceleration only two are used (RMS and 0-P).

The view of the screen with an activated generator is given below:



Figure 25. Screen with activated generator

When any of the three parameters is activated, the cursor changes into a stop sign. By pressing again the **E** key, generating of signal is stopped and the cursor regains its previous shape. Generating also can be stopped by pressing **BACK** button. When the cursor is set on the parameter and by pressing E, we enter the change of Velocity, Acceleration or Displacement value mode, depending which type of measurement we have chosen. In any case, the change of parameters is done by ↓ and ↑ arrow keys. When the desired parameter is set, by pressing the **E** key, we return to the menu and the parameter we have chosen becomes valid. We can choose the way the parameter is expressed and depending on the chosen parameter, different modes can be selected. As we have already mentioned, in case of Displacement parameter is chosen, it can be expressed in RMS, 0-P and P-P values, while for Acceleration and Velocity RMS or 0-P values are used. The mode is selected by pressing again the **E** key. By selecting the function EXIT and by pressing the **E** key, we return to the menu. With **BACK** button we return to the submenu for changing frequency or sensitivity for fast frequency change.

3.5 Speed and OSO[®] Output

Speed output is suitable for generating square wave signal that simulate the sensor which on its output gives a wave of rectangular impulses depending on the number of RPM. The voltage level of the sensor is on the TTL level (+5V) with the Duty cycle of 50%. **OSO[®]** output (**O**ptical **S**peed **O**utput) is suitable for the simulation of sensors which do not have a voltage output, but express the **RPM** by flashing of OSO LED diode. In both cases, the range in which the signal moves is from 10 RPM to 60,000 RPM, resolution of 1 RPM. When this type of generator is selected, a message appears on the screen that it is necessary to connect the conditioner to the **TTL** output(TTL BNC Connector). If we use OSO[®], we do not connect to BNC, but to the probe or we keep the conditioner against the LED diode, which sends the information about the RPM by light impulses.

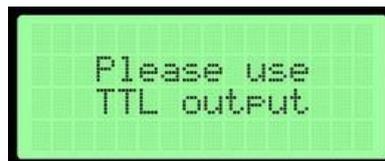


Figure 26. Message after activation TTL

After the message, the possibility of choosing the number of revolution per minute (RPM) that we want to simulate will appear on the screen, as well as the possibility of turning On/Off OSO[®] optical output.



Figure 27. Choosing appropriate RPM and activating OSO[®] screen

Navigating through the menu is done by the arrow keys ↓ and ↑. By the **E** key, we make a choice of the wanted option. Bringing the cursor to the RUN function and by pressing **E** key, generating of square wave signal is set. If **OSO**[®] is set to ON, light pulses will start together with generating TTL Output. Like in other working modes, the sign stop by RUN function, indicates that the system is active and generating square wave is in process on the output.



Figure 28. TTL and **OSO**[®] output started

By pressing **E** key again, we return to the change of parameters. By setting the cursor on RPM and by pressing **E** key, we enter the menu for changing RPM values we want to simulate. The two-sided arrows represent a visual indication of this, like in Figure 29.



Figure 29. Changing value of RPM

By pressing **E** key again, we return to the choice of function and save the set values. Like for RPM, **OSO** can also be changed from ON to OFF, ie. from OFF to ON by pressing **E** key after setting the cursor on **OSO**[®]. If it is set to ON, **OSO**[®] will be activated when the generator starts and if it is set to OFF **OSO**[®] it will not be activated after starting of the TTL square wave signal generator. By bringing the cursor on EXIT and by pressing **E** key again, we leave the subprogram and return to the Root Menu for selecting function. We can also leave this subprogram by pressing **BACK** key.

3.6 BOV Sensor Input

The purpose of **BOV** sensor input is to test sensors which require IEPE Power Supply as it mentioned before. When connecting sensor for testing, we have to be careful, since IEPE voltage of 24Vdc is also present on this output. **BOV** test is a fast way of identification of several types of malfunction of the IEPE sensors. The diagnostic is based on measuring the sensor Bias Voltage, when it is connected to internal 24Vdc IEPE Power Supply, builded in equipment. Depending on the voltage, we can determine the functionally state of the tested sensor.

After starting the program for the BOV sensor test, a message will appear on the screen about a necessary use of the output marked by **BOV** (BOV BNC Connector) on the front panel.

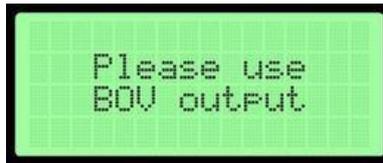


Figure 30. Message after activation BOV

After this message, a menu with two functions appears on the screen, RUN and EXIT. By arrow keys ↓ and ↑, we navigate between these two functions. Measurement is started by pressing the **E** key when the cursor is on the RUN function. After the measurement is completed, a message appears on the screen considering the voltage value, and also if the sensor is in short circuit, whether the circuit is open or the voltage values are within the normal range and the sensor is in good condition. The examples of these messages are given below. The voltage range, by which the program checks the functionality of the sensor, is the following: if the sensor bias voltage is below 7Vdc, the sensor is bad (Short Circuit). If voltage is higher than 15Vdc then the sensor is also bad or there is no sensor connected (Open Circuit). If the measured voltage is between 7Vdc and 15Vdc, the sensor is good (Normal).

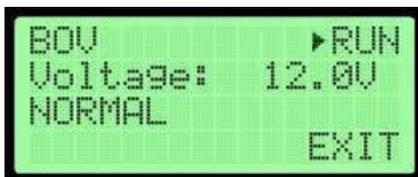


Figure 31. Sensor is OK

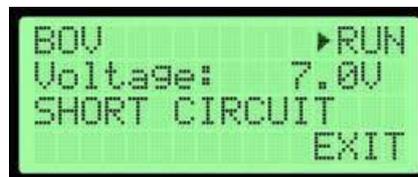


Figure 32. Sensor is in short circuit

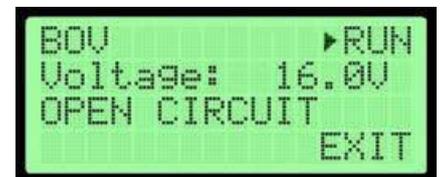


Figure 33. Sensor is in open circuit

By bringing the cursor on EXIT and by pressing **E** key, we return from the subprogram to the Root Menu. We can also leave this subprogram by pressing **BACK** key with cursor on any position.

3.7 Settings

This part of the program has the function to determine which measurement system will be used in **IEPE** and **Charge** subprograms. This device can express units in either the Metrical or the Imperial system. This change is done here, and the changed value is saved in the device's memory and is kept even after the device is turned Off. There is also an option for changing language.

As in all the subprograms by now, here navigation is done by ↓ and ↑ arrow keys, and the entry into the change or the exit from the subprogram by **E** key (if the cursor is on EXIT). The view of the subprogram is given below.



Figure 34. Settings subprogram

By bringing the cursor on Units and by pressing **E** key, we enter the menu for the change of units, and by ↓ and ↑ arrow keys, we change units into Metric or Imperial. By pressing again **E** key, the selection is saved in internal EEPROM where it is kept even after the device is turned Off. By setting the cursor on EXIT and by pressing **E** key, we return to the Root Menu. We can also return to Root Menu by pressing **BACK** button on any cursor position.

4. PC connection

Sensor Simulator MSS-1010 can be also controlled from the PC (notebook) through the USB interface using 4 pole LEMO connector on front panel of the Sensor Simulator and appropriate interface cable. When the connection is established, we will get the next message on LCD screen:

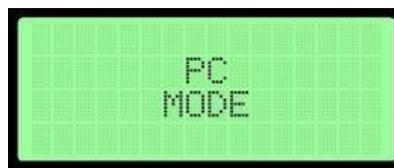


Figure 35. PC connection

Attention

The outer terminal of every BNC connector of Sensor Simulator MSS-1010 is galvanically interconnected by means of a USB cable with the computer frame. During the use, it is necessary to prevent a possible occurrence of grounding loops, e.g. by means of battery notebook operation, disconnection from LAN, etc.

Now, the Sensor Simulator MSS-1010 is ready for communication with PC. We cannot use Simulator's keyboard to setup any output values. For communication and setting up the values we can use program SS- 1000Control or any lab program that support binary serial COM port communication.

Communication parameters are: **9600 bauds, No Parity, 8 data bits, 1 stop bit.**

Protocol is always consisted from two special characters and eight bytes for information. Every communication word start with special character **:** and ends with second special character **#**. In the middle are eight bytes. First three carries information about voltage level (first byte is most significant); second three carries information about frequency (first byte of frequency in array is most significant); seventh byte carry information about starting or stopping generation of signal and eight byte carry information about chosen function.

Desired voltage must be multiplied by 100 before parsing it to three bytes, because this way we can have two decimal spot in the voltage that is combined on the device.

Explanation of bytes

: $b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 \#$

- b_1 – most significant byte of voltage level (17bit to 24bit of voltage value)
- b_2 – 2nd byte of voltage level (9bit to 16bit of voltage value)
- b_3 – least significant byte of voltage level (1bit to 8bit of voltage value)
- b_4 – most significant byte of frequency value (17bit to 24bit of frequency value)
- b_5 – 2nd byte of frequency level (9bit to 16bit of frequency value)
- b_6 – least significant byte of frequency level (1bit to 8bit of frequency value)
- b_7 – start/stop byte (if we want to start the generation, set the value to 1; if we want to stop, than set the value to 0)
- b_8 – information about chosen function (0 – mV/charge, 1 – Battery voltage, 2 – OSO ON, 3 – OSO OFF, 4 – ping, 5 – IEPE)

Example

If we want to send number 560.50 we must multiply it with 100 and that number parse to three bytes. This can be easily done by shifting, but there are also some other way to do this. On this way we have 24bit variable that can be sent to Sensor Simulator. Voltage range is from 10 mV to 10 V.

Frequency is sent to the device on same way. We cannot generate frequency higher than 1 kHz.

Protocol

On every command we will receive appropriate answer. Commands and answers are given below.

For voltage generation without IEPE we use function 0. Command for this is : $v_1 v_2 v_3 f_1 f_2 f_3 ss 0 \#$ where v_1 , v_2 and v_3 are three bytes that carries information about desired voltage, f_1 , f_2 and f_3 are bytes that carries information about desired frequency, ss byte can be 1 if we want to start generation or 0 if we want to stop generation of voltage on mV output. Eight byte carries information about function and it is 0 for voltage generation without IEPE. If everything is ok, and values are correct and in the range we will receive : $OK \#$ for answer. If any values are out of range we will receive answer : $E 0 \#$.

For information about battery we use function 1. Command for this is : $0 0 0 0 0 0 0 1 \#$. Answer is consisted from 4 bytes, : $b_1 b_2 \#$ where battery voltage can be calculated from the formula given below:

$$U_{bat} = \frac{b_1 \cdot 256 + b_2}{100}$$

For enabling OSO[®] we must use next command : $0 0 0 0 0 0 0 2 \#$. Answer will be : $O 1 \#$.

For disabling OSO[®] we must use command : `0 0 0 0 0 0 3 #`. Answer will be : `0 0 #`.

For ping we use command : `0 0 0 0 0 0 4 #` and if there is device on the other end of the cable we will receive answer : `! ! #`. This is very useful if there are a lot of ports and we are searching for one where is our device connected.

For IEPE voltage generation we use function 5. Command for this is : `v1 v2 v3 f1 f2 f3 ss 4 #` where `v1`, `v2` and `v3` are three bytes that carries information about desired voltage, `f1`, `f2` and `f3` are bytes that carries information about desired frequency, `ss` can be 1 if we want to start generation or 0 if we want to stop generation of voltage on IEPE output. If everything is ok, and values are correct and in the range we will receive : `0 K #` for answer. If any values are out of range we will receive answer : `E 0 #`. IEPE function can generate accurate voltage up to 6V.

In every command, first and last bytes are given in ASCII and since this is byte (binary) protocol we must use their values from ASCII table. For `:` this is decimal 58 of hex 0x3A and `#` is 35 decimal or 0x23 for hexadecimal value. All other values are bytes, not ASCII codes. Also, when you receive answer, given symbols are coded in ASCII, but you can use their byte value to operate with them.

Caution

Using a protocol to made custom applications is only for ADVANCED users. This can damage your device if you don't know how to use it. Higher rates of voltage can damage your equipment (especially in IEPE mode). We are not responsible for any damage you made using this option. Please use this option on caution. It is safer to use manufacturer application which can be downloaded from the official web site.

5. Charging connector

When the batteries are discharged, the screen will indicate like on Fig. 36.



Figure 36. Screen with empty battery

After about 15 seconds, instrument will automatically shut off to prevent the generation of signal which is outside specified accuracy for sensor simulator. In that case it is necessary to connect charger. This connector is used for connecting external regulated power supply 9Vdc for charging internally mounted batteries. When charger is connected, the CHARGE led is lit. When the charger is connected to device DO NOT use it to generate any signals. Charging time for fully charged batteries depends from capacity of the battery and it is approximately 16 hours. Integrated charger electronics will automatically stop the charging process when the batteries are fully charged.

6. Device Dimensions

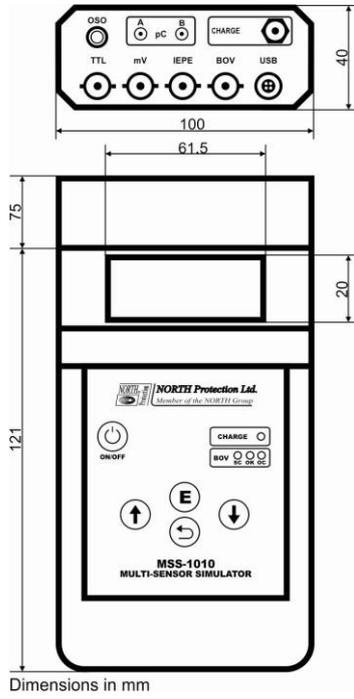


Figure 37. Device dimensions in millimeters

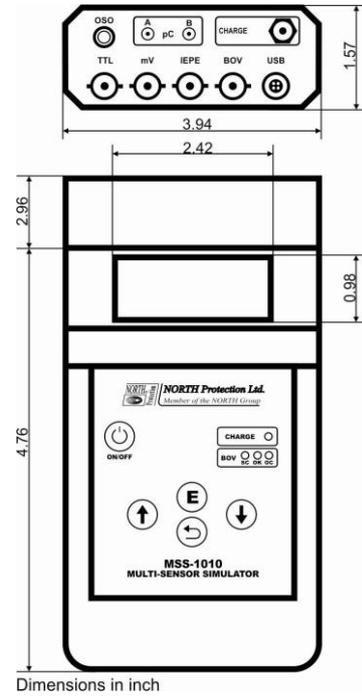


Figure 38. Device dimensions in inches