Signal-Elektronik und Netz-Dienste GmbH

GEOLON-MLS

Marine Longtime Seismocorder

User Manual

Table of contents

1 Features 2
1.1 General 2
1.2 Key Features 2
1.3 Sample Rates and Resolution 3
1.4 Physical Dimensions 4
1.5 Environmental Properties 4

2 Functional Description 5
2.1 General 5
2.2 Elements 6
2.3 Modularity 7
2.4 How to Use PCMCIA Memory Modules 10

3 PC Software 11
3.1 SENDCOM 11
3.2 SEND2X 12
3.3 Files 12
3.4 Time Slips 15

4 Interface Description 16
4.1 Power Supply 16
4.2 Analog Inputs 17
4.3 Serial Interface (RS232) 21
4.4 Auxiliary 22

5 Command Description 23
5.1 Help 24
5.2 Information 24
5.3 Acquisition Control 24
5.4 Time Services 26
5.5 PCMCIA Module Services 27
5.6 Accessing Data 28
5.7 Definition of PASSCAL Parameters 29

6 System Responses 31
6.1 Interactive Operation 31
6.2 Stand-alone Operation 36
6.3 Error Indication 36

7 Description of a Typical Operation Session 37
7.1 Interactive Operation via External PC 37
7.2 Stand-Alone Operation without PC 38
7.3 Analysis of Recorded Data 39
7.4 Brief Description for Experienced Users 39
7.5 Examples for Interactive Operation 40

8 Loading a New Software Release 42

9 Troubleshooting 42
9.1 Emergency firmware erase for booting problems 42
9.2 MLS not responding/hanging 42

10 Support and Service 43

11 Appendix: SEND2X manual 44

User Manual

Document No. MLS05j.mnl
January 2007

Manufactured by:

Signal Elektronik GmbH

Rostocker Str. 20
D-20099 Hamburg

Phone: +49 40 375008 03
Fax: +49 40 375008 93

URL: http://www.send.de
e-mail: office@send.de

1 Features

1.1 General

GEOLON-MLS is a high-precision instrument for acquisition, processing and storage of seismic signals. It is optimised for long time (more than 1 year) stand-alone operation on the ocean bottom. Nevertheless, due to its compact size and simple handling the instrument can also be used for long time data acquisition in land applications. Protected by a suitable housing, e.g. a sealed tube, it may be buried in the ground and stay there for long time stand-alone operation.

A front-end with four analog input channels behind the front plate is designed as exchangeable unit. Three channels are prepared for connection of a 3-axis-seismometer; adaptation of different types of seismometers is possible. One channel is prepared for connection of a hydrophone or a pressure sensor and therefore it is equipped with a low-noise preamplifier with four different preselectable gain factors. A special version of the MLS for data acquisition from a depth sensor is available on request. For this version, the hydrophone channel is adapted to the depth sensors and cannot be used to acquire data from a hydrophone anymore. This version is marked by a type-label 'MLS 13' on the housing.

1.2 Key Features

Time synchronisation	DCF77 or single pulse
Internal time base drift	< 0.05 ppm (0 ... +30°C)
Power consumption	Recording: @ 50 sps 230 mW without preamplifier (LOWN2X) for hydrophone ; 250 mW with LOWN2x; 255 mW with 12 PCMCIA microdrives as storage devices; Lowbat standby: 100 mW
Storage medium	PCMCIA flash-disk / hard disk
Storage capacity	12 PCMCIA slots type II (at present good for 12 GB flash cards or 24 GB hard disks)
Weight	1.5 kg without batteries and PCMCIA storage modules

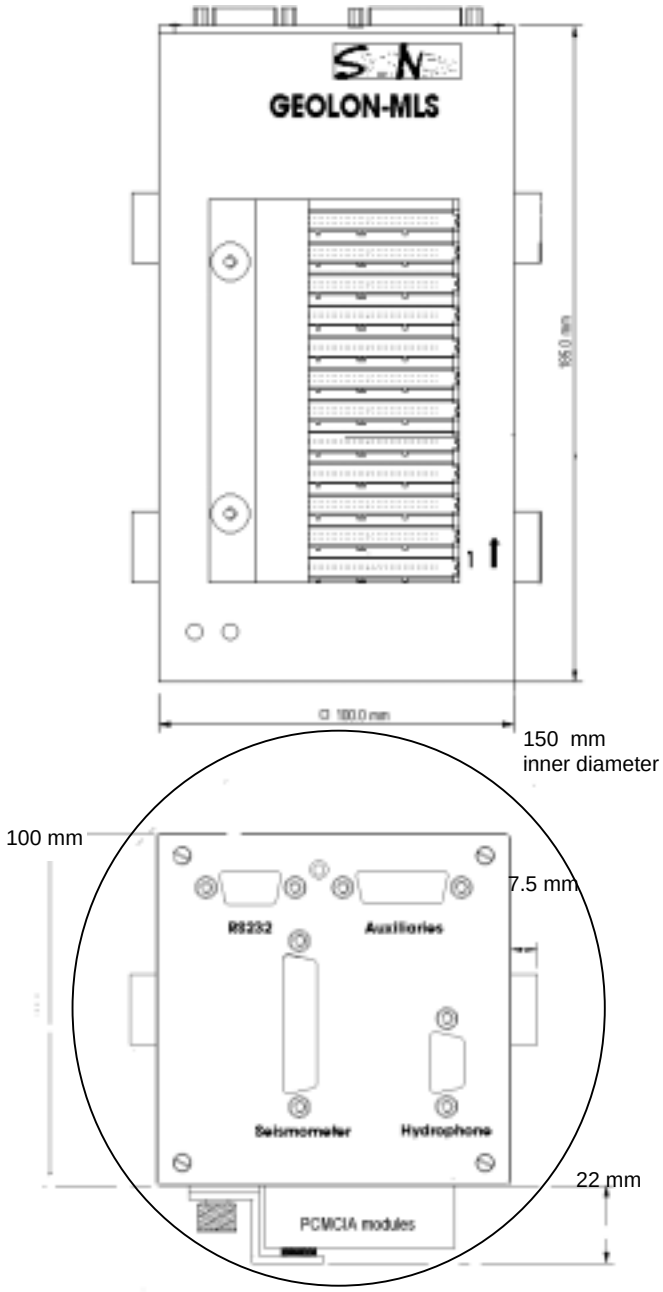
1.3 Sample Rates and Resolution

Samples per second	f-3dB (Hz)	Resolution (Bits)	Signal-to-Noise Ratio (dB)
1	0.3	22	120
2	0.7	22	120
5	1.7	22	120
10	3.3	22	114
20	6.7	21	110
30	10.0	20	106
50	16.7	19	100
100 *	33	18	96
200 *	67	15	78

Further sample rates selectable are: 3, 4, 6, 25, 40, 60, 75, 120*, 130*, 150* Hz

*optional

1.4 Physical Dimensions



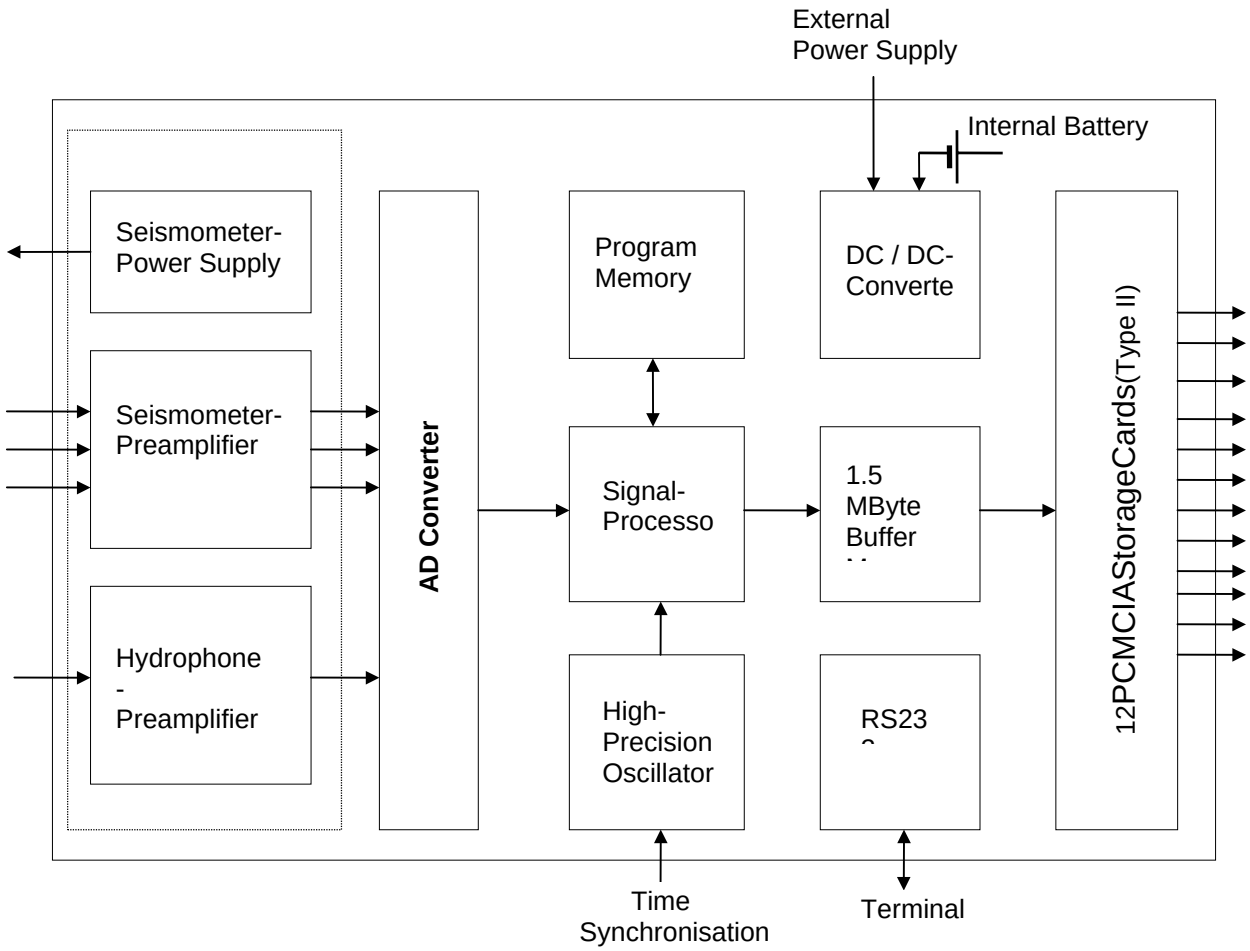
1.5 Environmental Properties

Operation Temperature:	using mil-spec flashcards	-20°C to 70°C
	using hard disks	0°C to 55°C
	for max. time base stability	0°C to 30°C
Storage Temperature:		-45°C to 70°C
Humidity		100% non condensing

2 Functional Description

2.1 General

In the block diagram the components of GEOLON-MLS are shown as well as the data flow with its successive processing steps.



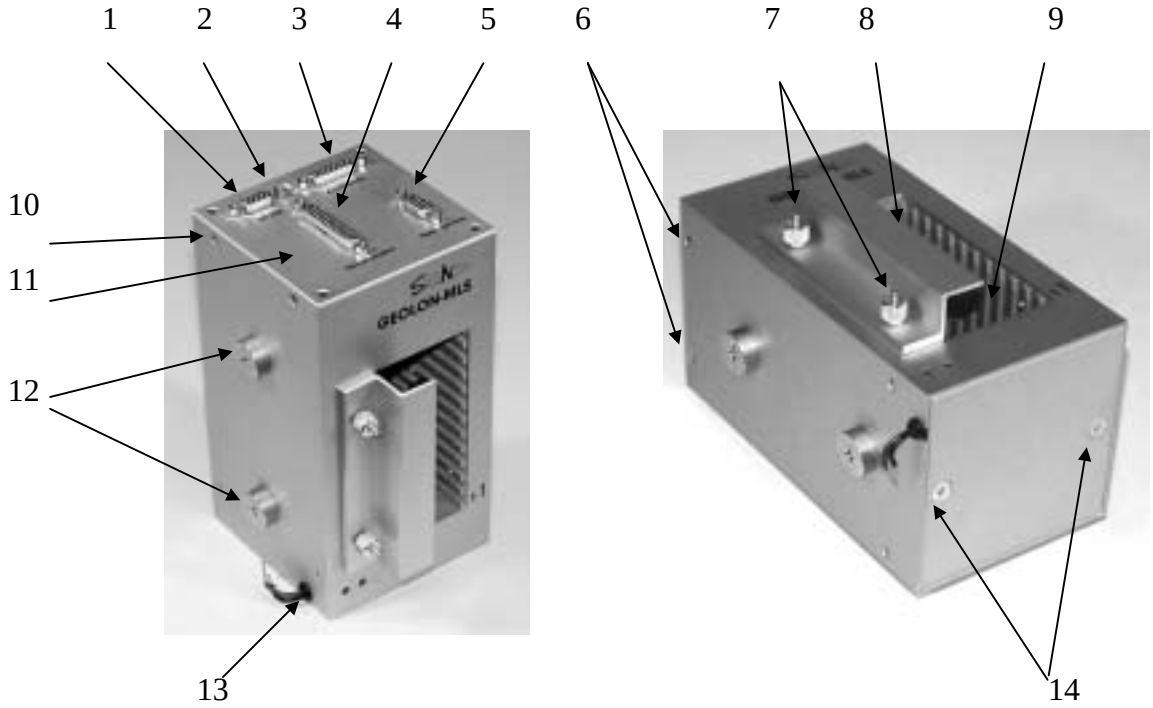
Block Diagram

The instrument can be parameterised and programmed using an ASCII terminal via its RS232 interface. After low pass filtering the signals of the four input channels are digitised using Sigma-Delta A/D converters. A final decimating sharp digital low-pass filter is realised in software by a Digital Signal Processor. The effective signal resolution depends on the sample rate as shown in the table in chap.1.3 The sample rate is software selectable as described in chap. 5.3. Finally, the samples are permanently stored on PCMCIA flash- or hard-disk memory modules.

A high precision oscillator, which is synchronised using DCF77 compatible pulses, controls the time management hardware. Time deviation during recording is determined using the SKEW command and then written to the **MLS.SYS** file. In addition, time slips can appear due to oscillator technology. Please find a detailed description in chap. **Fehler! Verweisquelle konnte nicht gefunden werden.**.

The recorded data is played back by plugging the PCMCIA storage cards into a PC with PCMCIA interface. All necessary PC-software is part of the standard deliverables (see chap.3). The software package SEND2X can be used for Linux computers to read the acquired data from PCMCIA storage disks and to generate a non standard SEG-Y file, which is compatible to the processing software of RefTek, as well as other formats.

2.2 Elements



1 Connector for serial interface (RS232)

2 LED

3 Auxiliaries connector

4 Connector for 3-axis seismometer

5 Connector for hydrophone (or pressure sensor or depth sensor)

6 Screws to fix the analog electronic exchangeable unit

7 Screws to fix the PCMCIA cover

8 Z-shaped PCMCIA cover

9 12 PCMCIA slots

10 Slot for screw driver to lift the front plate

11 Front plate of analog electronic exchangeable unit

12 Interfaces for installation (on request)

13 Lead-in hole for power connection cables

14 Quick-fix screws to close the battery case (View of open case in chap.4.1)

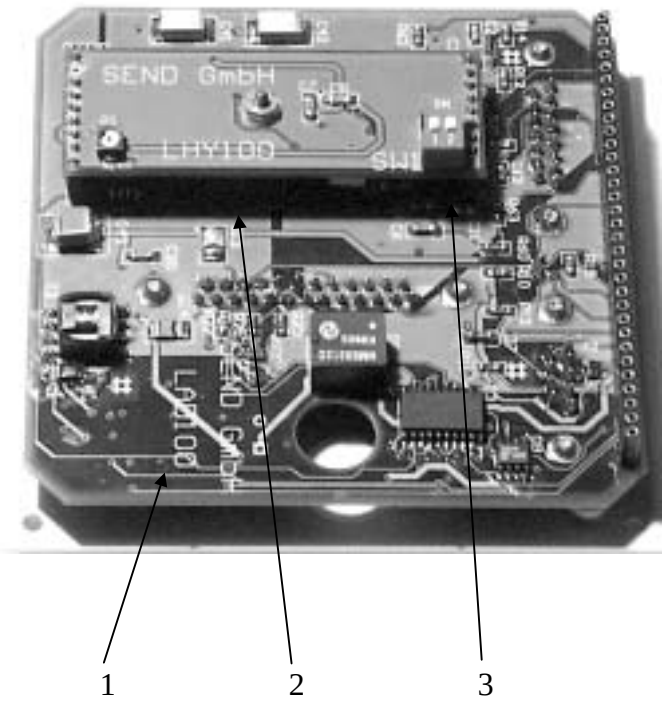
2.3 Modularity

The modular design of GEOLON-MLS enables easy adaptation to different sensor types. For this purpose the analog electronics are mounted on the rear side of the front plate and become accessible when the front plate is lifted. There are two PCBs that may be changed: the LAD1xx which is seismometer specific, and a smaller one for adaptation of a hydrophone (LHY100) or a pressure sensor (LDR100). Available PCBs for sensor adaptation and their specifications are described in chap. 4.2.

If the use of different sensor types is rather unlikely the suitable PCBs should be selected and installed. If a frequent change of sensors is expected, it is recommended to prepare different sensor specific exchange units each consisting of a front plate and the suitable PCBs. These analog units can easily be exchanged as a whole.

How to lift the front plate and how to change the sensor specific PCBs is shown in the following illustrations.

Insert a screw driver into the slot (No. 10 in chap. 2.2)
and turn it to loosen the front plate until it can be taken off.

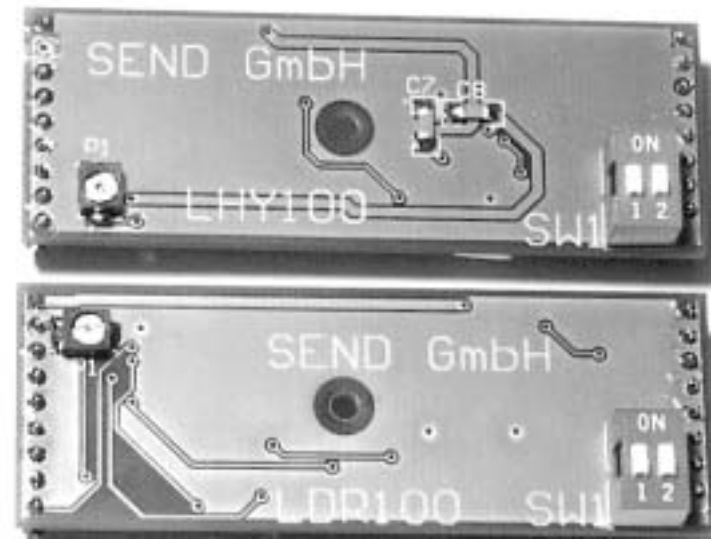


1 Seismometer specific PCB: LAD100

2 Piggy-back PCB for hydrophone / pressure sensor: LHY100

3 Switch SW1 for selection of preamplification gain

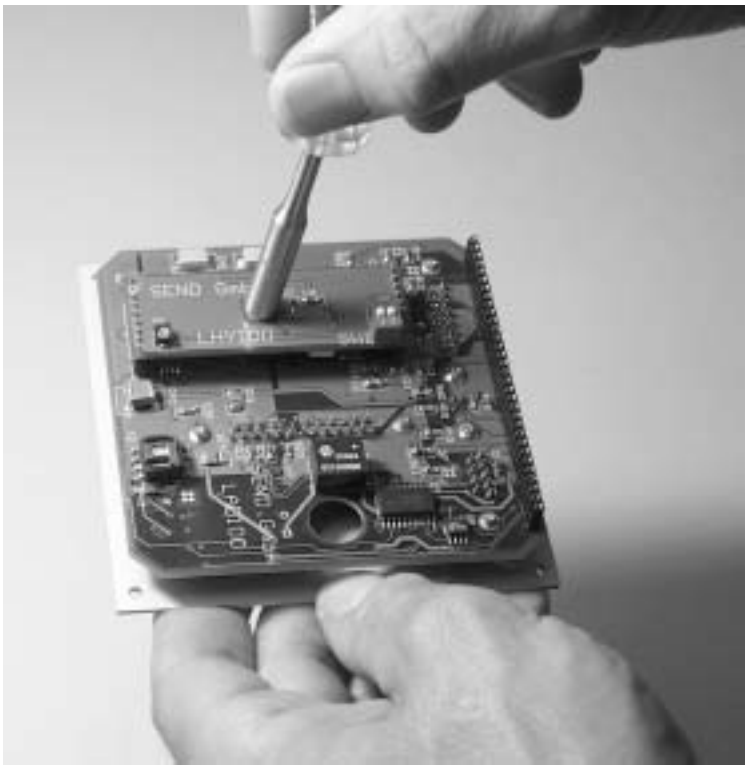
Analog electronic exchangeable unit



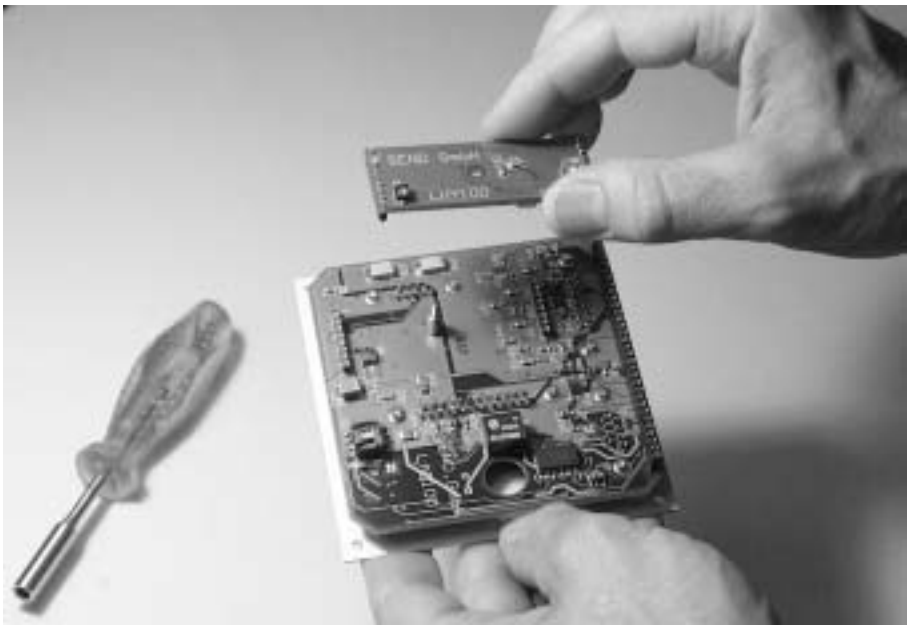
Piggy-back PCB for use of a hydrophone: LHY100

Piggy-back PCB for use of a pressure sensor: LDR100

Change of piggy-back PCB:



Remove the nut.



Remove / exchange the piggy-back PCB.

Since the hole for the fixing peg is positioned eccentrically in the PCB an incorrect plug in of the PCB is not possible.

2.4 How to Use PCMCIA Memory Modules

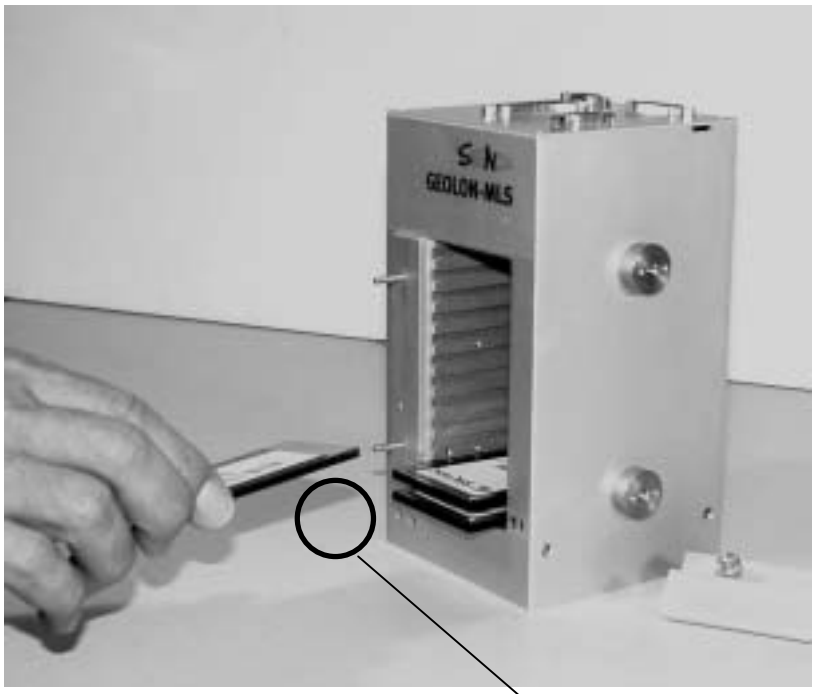
GEOLON-MLS is equipped with 12 slots for PCMCIA ATA memory modules. A maximum of 12 PCMCIA cards of Type II or Type I can be inserted. Cards of Type III need the space of two slots due to their thickness, so a maximum of 6 modules of Type III can be inserted.

PCMCIA flash cards as well as PCMCIA hard disks may be used, and they may be mixed arbitrarily. Slots to be used may be selected arbitrarily, too, with the exception of slot 1:

Note: the first card has to be inserted in slot 1 !

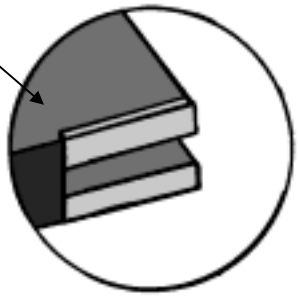
2.4.1 How to Plug In PCMCIA Storage Cards

The housings of PCMCIA ATA modules are coded as shown below in order to avoid an incorrect plug-in. In the correct position the cards can be inserted smoothly. If you feel significant resistance please make sure that the card is inserted in the correct position.



Attention !

Insert the PCMCIA-card as shown above:
the slotted edge on the right!



3 PC Software

The CD-R with PC software contains the following programs:

SENDCOM
SEND2X

3.1 SENDCOM

This program is used to communicate with GEOLON-MLS via the RS232 interface, e.g. for interactive configuration and for real-time display of recorded data on the PC monitor.

SENDCOM is a JAVA program, which can be run on any JAVA-enabled Windows or Linux computer.
As default, SENDCOM uses COM1 as communication port. If you want to connect to another port, you can do so from the SENDCOM menu bar.

SENDCOM also allows the “show” command (see command section) to display actual sensor data from any of the four channels in realtime.

3.2 SEND2X

This program-package converts the compressed recordings of SEND recorders into different formats. SEND2X is available for the operating system Linux.

The current version allows the conversion of raw data into a binary file, an audio-wave file, or into the SEG-Y format.

On the delivered CD-R you will find a program-library and a script, which includes an example of combining the different programs. Each of the programs can be used separately.
Concerning GEOLON-MLS, the library includes following programs:

mlsread
binwrite
gsewrite
paswrite
sacwrite
segywrite
wavewrite
resample

All programs using as input device the standard input, partly as output device the standard output. This allows an easy combination of the different programs through 'piping'.
A detailed description of SEND2X can be found in the Appendix.

3.3 Files

GEOLON-MLS creates two different types of files:

- data files
- **MLS.SYS** - file

3.3.1 Data Files

Names of data files are automatically generated by GEOLON-MLS during recording and by the PC after conversion and they have the following structure:

<4-digit serial number>
<3-digit day-of-year>
<1-digit sequence number>
.
<2-digit hour>
<1-digit channel>

- <4-digit serial number>
  is derived from the GEOLON-MLS serial number and consists of 3 sub-fields:

  <1-digit year code> year since 1990, counting above "9" continues with "a", "b", etc.
  <1-digit month code> "a" for October etc.
  <2-digit serial number>

- <3-digit day-of-year>
  day number of the synchronisation time.

- <1-digit sequence number>
  recording sequence number. This field is only inserted after conversion into a PASSCAL file. "Raw data", i.e. original recorded compressed data has an "_" (underline) character in this position.

- <2-digit hour>
  hour of the synchronisation time.

- <1-digit channel>

for raw data, this field contains the slot number of the medium. After conversion, this is the channel number of the PASSCAL file.

3.3.2 MLS.SYS

This file contains all control, status, and identification information of the actual experiment and the particular PCMCIA card. It is the first file on each PCMCIA card; the remaining storage capacity is available for data files.

As an example the contents of a typical MLS.SYS - file is described hereafter:

File content	Comment
Set basic parameters:	
EVERY 7 DAYS **AFTER 3 HOURS LEVELLING**	first after 3 hours, then weekly
ALL CHANNELS **50 RATE**	hydrophone and seismometer sampling-frequency is 50 Sps
EXPERIMENT experiment_name" **COMMENT experiment_comment"**	
Optional passcal strings:	
PASSCAL	enter the PASSCAL sub menu
12 EXPNUMBER	EXPNAME already set (alias for EXPERIMENT) EXPCOMMENT already set (alias for COMMENT)
1234 STATNUMBER **STATNAME geolon exploration"** **STATCOMMENT geolon configuration by file on pc-card"**	
HYDROPHONE CHANNAME hydrophon" **HYDROPHONE CHANSENSOR hydro_labs_1"** **HYDROPHONE CHANSENSORNUMBER #12345678901"** **HYDROPHONE CHANCOMMENT preamplifier LHY100"**	
SEISMOMETER1 CHANNAME Spahr Webb" **SEISMOMETER1 CHANSENSOR hydro_labs_2"** **SEISMOMETER1 CHANSENSORNUMBER #23456789012"** **SEISMOMETER1 CHANCOMMENT Z axis"**	
SEISMOMETER2 CHANNAME Spahr Webb" **SEISMOMETER2 CHANSENSOR hydro_labs_2"** **SEISMOMETER2 CHANSENSORNUMBER #34567890123"** **SEISMOMETER2 CHANCOMMENT N-S axis"**	
SEISMOMETER3 CHANNAME Spahr Webb" **SEISMOMETER3 CHANSENSOR hydro_labs_2"** **SEISMOMETER3 CHANSENSORNUMBER #45678901234"** **SEISMOMETER3 CHANCOMMENT E-W axis"**	
EXIT	leave PASSCAL sub menu

DCF77 synchronise time, format cards

REC start recording

All subsequent messages are append by GEOLON-MLS when the SKEW or CLOSE command is executed:

\\

[sync_time]
01.10.1999 12:17:00

[this_card]
card #1

[file_name]
card #1: 9967274_.121

[skew_time]
08.10.1999 12:35:00

[system_time]
08.10.1999 12:34:59 and 689 ms

[deviation]
-311 ms

[messages]
0 messages

[settings]
Date & Unit 08.10.1999 12:35:20 990905
Synchronized 01.10.1999 12:17:00
Channels all
Sampling 50 Hz, 19 bits
Capacity 261312 KB data, cards closed
Start 01.10.1999 12:18:13
Stop 08.10.1999 12:51:34
Levelling Every 7 days After 3 hours
Status cards closed
Experiment experiment_name
Comment experiment_comment

[FIR-FILTER]
2 decimation
44 coefficients
#1 : -0.000022315
#2 : -0.000190424

... ...

#44 : -0.000022315

[delay]
240 ms, 12 samples

Delay introduced by the digital FIR filter

3.4 Time Slips

The Seascan oscillator inside the MLS has a free running 4 MHz oscillator with its "natural" temperature dependence on frequency (about 20ppm). From this oscillator an intelligent "divider" is fed, producing a temperature compensated 1 Hz pulse. This pulse is used by the MLS for time keeping. The 4 MHz oscillation is used to generate the sampling frequency. Therefore, the sampling frequency is much less precise then the time marks. That means, while the time of the internal clock is accurate (especially after the linear correction of the drift of the oscillator) the sampling rate can vary slightly sometimes. Then, the sampling period is not constant every time. Over long measurement periods, these small failures could be summarised to a period of one sample period. Then you will get a "time slip" message during conversion of the data. That happens, whenever there are 99 (negative value) or 101 (positive value) samples between two second time marks (at a preset sample rate of 100 Hz). These information are given for the case, that for the detection of an event a precision is needed, which has to be better than one sample period. The utility package SEND2X contains the program RESAMPLE, which provides a resampling of the data for attaining a equidistant sample rate.

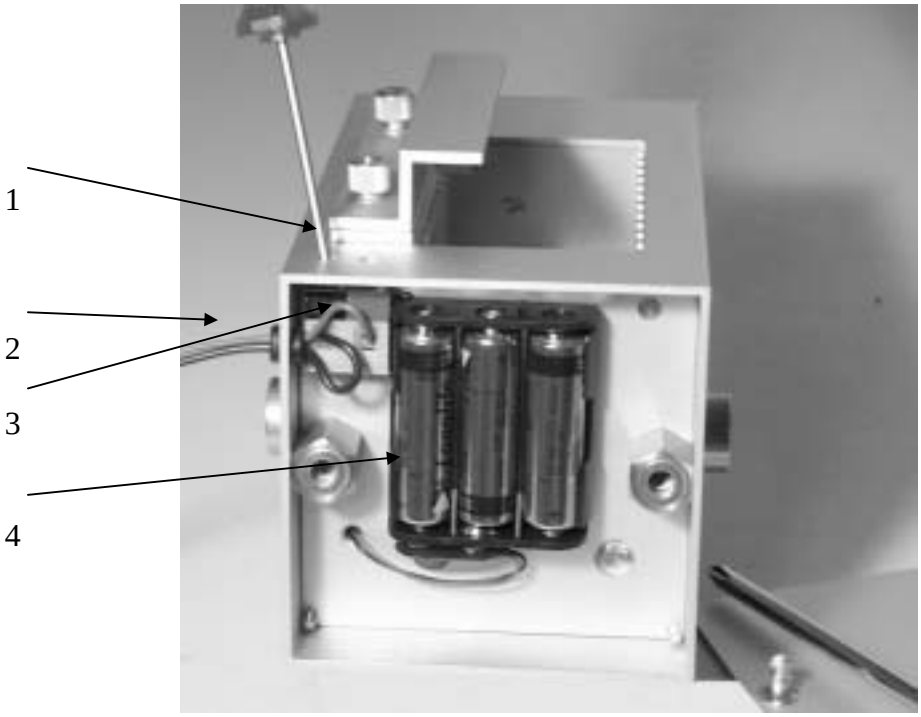
4 Interface Description

4.1 Power Supply

Power has to be supplied to GEOLON-MLS via the red (+) and black (-) terminals in the bottom case. The input voltage range is specified in the table. An integrated high-efficiency DC/DC converter generates the voltages needed by GEOLON-MLS.

Note: When hard disks are used as storage medium, the power supply must be capable of delivering at least 8 Watt of peak power during the acceleration phase of the hard disk!

In order to keep the time synchronisation during brief power interruptions, e.g. for experiment preparation, 3 size AA batteries can be inserted. The battery case is positioned adjacent to the power supply terminals as shown below. With the AA batteries only, it is **not possible** to record data!



Bottom case for power supply connection and batteries (opened)

1 Screw driver for terminal screws

2 Cables for power supply connection fed through the hole in the side wall

3 power supply terminals
red +
black -

4 three batteries inserted

External Input Voltage Range	Necessary Internal Battery Cells (Voltage)
6.2 ... 16.5 V	3 AA cells / (4.8 V)

Power of the internal batteries is only activated if at least one PCMCIA card is inserted. This function avoids unintentional discharge of inserted batteries.

A ‘low bat’ message will be generated when the input voltage drops below 6.8 to 5.8 V. The recorder then stops recording and switches to stand-by mode. In this mode, it just keeps synchronization alive at a power consumption of about 17mA. A shut-down function is initiated when the input voltage drops below 4.4 V. This keeps the processor from operating in uncontrolled conditions.

4.2 Analog Inputs

GEOLON-MLS has four analog input channels. Three channels are earmarked for connection of a three-component seismometer (see chap. 4.2.1); one channel is prepared for connection of a hydrophone and a pressure sensor and is therefore equipped with an additional preamplifier (see chap. 4.2.2). To activate the seismometer channels, the hydrophone / pressure sensor channel or all of them the CHANNELS command can be used (see chap.5.3). Please note, that the MLS version adapted to depth sensors can not be used for acquiring hydrophone data unless the proper exchangeable electronic unit is installed.

4.2.1 Seismometer

The analog signal conditioning for the 3 seismometer channels is identical. The input is a single-ended current amplifier with an input current range of +/- 192 µA for a full-scale signal. Three one-pole low-pass RC-filters (-3dB @ 150 Hz) give the necessary low-pass response in order to prevent aliasing under all operating conditions.

A current input has been realised in order to be independent from the different output signal levels of different seismometer models. It is best practice to install the series resistor which "converts" the voltage output of the seismometer into the current input for GEOLON-MLS on the seismometer side. This way the underwater cable which connects the seismometer output to GEOLON-MLS input is at "virtual ground" minimising the danger of electrolytic corrosion. Alternatively, this resistor may also be installed inside GEOLON-MLS on custom order.

Pinout of the Seismometer Connector

Pin	Signal	Remarks
1	GND	
2	Channel 1 (-)	Ground return of single-ended input. Z- seismometer signal.
3	GND	
4	Channel 2 (-)	Ground return of single-ended input. NS-, X- seismometer signal.
5	GND	
6	Channel 3 (-)	Ground return of single-ended input. EW-, Y- seismometer signal.
7	GND	
8	2048 Hz	Synchronous chopper clock, CMOS-level (0-5V)
9	OUT1	Control output, CMOS-level (0-5V)
10	GND	
11	+6 V	Seismometer power supply, 25 mA max.
12	-6 V	Seismometer power supply, 25 mA max.
13	GND	

14	Channel 1 (+)	Single-ended input, +/- 192 µA full scale, Z+ seismometer signal.
15	GND	
16	Channel 2 (+)	Single-ended input, +/- 192 µA full scale, NS+, X+ seismometer signal.
17	GND	
18	Channel 3 (+)	Single-ended input, +/- 192 µA full scale, EW+, Y+ seismometer signal.
19	GND	
20	GND	
21	OUT2	Control output, CMOS-level (0-5V)
22	level	Signal for control of gimbaling, 3.3V and 200 mA max. for MLS version 12, +12V and 50 mA max. for MLS version 10 (the version number is marked on the back side of each MLS) . See also "Levelling" command in chapter 5
23	GND	
24	GND	
25	GND	

4.2.2 Hydrophone / Pressure Sensor / Depth Sensor

For the regular version of MLS, two different types of sensors can be used: hydrophone or pressure sensor, which require different preamplifier modules as shown in the table below:

Sensor Type	Preamplifier Module
Hydrophone	LHY100
Pressure sensor	LDR100

(See chap. 2.3 on how to access and change the preamplifier modules.)

Since most users operate only hydrophones the standard model is equipped with LHY100; that is why the connector has been marked "Hydrophone". If you change the preamplifier module it is recommended to indicate on the front plate which module is installed.

The analog input signals coming from both sensor types are fed into the same 9-pin female D-SUB connector. Since different connector pins are used for different sensor types it is guaranteed that no damage can be caused in case of installation of the wrong preamplifier module. Nevertheless, before using GEOLON-MLS or installing it into a measurement device, be sure that the right preamplifier module (LHY100 or LDR100) is installed, because you will receive no signal with the wrong module.

The input sensitivity may be preselected by setting a switch combination on the PCBs LHY100 or LDR100 respectively, see photo in chap. 0 to find switch SW1.

The MLS version adapted to depth sensors is marked by a type label 'MLS 13' on the recorder housing. This version is prepared for depth sensors which send their data in ascii format via a RS232 interface to the recorder (e.g. 8CB7000-I of Paroscientific, Inc.). The MLS recorder transforms this data into binary format and stores them instead the hydrophone data on the disk. The data are stored in the origin units, given by the sensor, times 100 (e.g. in psi*100 for 8CB7000-I of Paroscientific, Inc.).

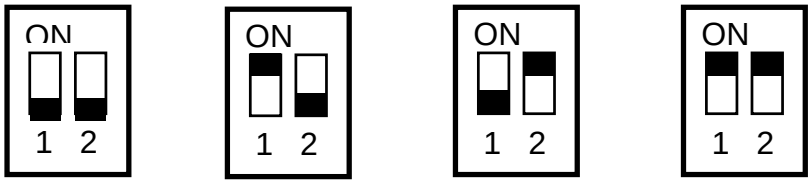
Hydrophone:

Pinout when using a hydrophone

Pin	Signal	Remarks
1	+5V	power supply for preamplifier module, 10 Ohm series resistor
3	IN	analog input, 0.03 ... 50 Hz, sensitivity switch-selectable (see below)
5	-5V	power supply for preamplifier module, 10 Ohm series resistor
8	GND	
9	GND	

Switch (SW1) positions for input sensitivity selection:

Switch position
SW1 on LHY100:



Input Sensitivity: 1.25 Vss 650 mVss 410 mVss 325 mVss

Note: Most hydrophones deliver very high impedance signals, which have to be preamplified before feeding into GEOLON-MLS. The use of the low-noise preamplifier LOWN22 is recommended which is best adapted to GEOLON-MLS. For connection use only cables with a grey connector!

Low-noise preamplifier
LOWN22 with grey connector

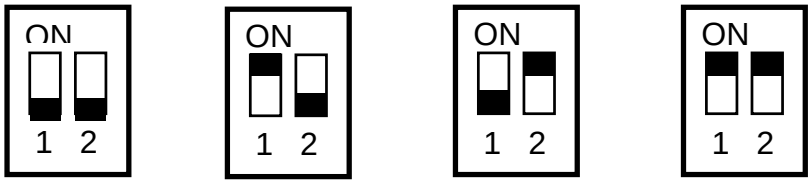
Pressure Sensor:

Pinout when using a pressure sensor

Pin	Signal	Remarks
2	Bridge +	input 2 of pressure sensor
6	Bridge -	input 1 of pressure sensor
7	UREF	power supply for strain-gauge bridge (+5V)
8	GND	
9	GND	

Switch (SW1) positions for input sensitivity selection:

Switch position
SW1 on LDR100:



Input Sensitivity: 266 mVss 133 mVss 92 mVss 66 mVss

Example: The Spahr Webb DPG sensor requires an input sensitivity of 92 mV.

Depth sensor:

Pinout when using Depth Sensors of Paroscientific or equivalent sensors

Pin	Signal	Remarks
1	+6V	power supply for depth sensor
2	TXD	transmit data to sensor (RXD of sensor)
3	RXD	receive data from sensor (TXD of sensor)
5	GND	

4.3 Serial Interface (RS232)

This interface is used for interactive operation with an ASCII terminal (e.g. for programming or parameterisation of GEOLON-MLS). If a modem is used for communication during stand-alone operation the modem can be connected to this interface too.

The signals are viewed from the PC-interface and adhere to the standard PC pin-out. A 9-pin modem extension cable is needed to connect GEOLON-MLS to a serial PC interface.

Pin	Signal	Remarks
1	DCD	always true (+5V)
2	RxD	output
3	TxD	input
4	DTR	input: needed to signal the presence of an RS232 terminal to the MLS
5	GND	
6	DSR	always true (+5V)
8	CTS	always true (+5V)

4.4 Auxiliary

This 15-pin female D-SUB connector carries the signal needed for time synchronisation and additional outputs which may be used to control external equipment. All logic inputs and outputs carry TTL-compatible signals.

Pin	Signal	Remarks
1	+5V	output; power supply, 10 Ohm resistor in series
2	GND	
3	-DCF77	input for time synchronisation, low-active, 10 kOhm pull-up
4	OUT1	auxiliary output
7	OUT2	auxiliary output, used by the RELEASE command
11	GND	
12	GND	
13	GND	
14	GND	
15	GND	

During booting MLS, the firmware can be erased by connecting Pin3 with Pin4 (see also chapter 9.1).

4.4.1 -DCF77 (Pin 3)

This time synchronisation input is used to set the time and date using a DCF77 compatible pulse train. An active low input signal is needed which may be open-collector because a pull-up resistor is connected internally.

An appropriate signal has to be fed into this input when the SYNCHRONIZE, DCF77 or RESYNCHRONIZE commands are executed.

4.4.2 OUT1 , OUT2 (Pins 4 and 7)

Auxiliary outputs. They are controlled by OUT1/OUT2 ON/OFF commands.

OUT1 is also used as an output for an activation signal for an external GPS receiver. The signal can be activated by using the RESYNCHRONIZE command (see also paragraphs 5.3 and **Fehler! Verweisquelle konnte nicht gefunden werden.**).

OUT2 is also used by the RELEASE command (see also paragraphs 5.3 and **Fehler! Verweisquelle konnte nicht gefunden werden.**).

5 Command Description

GEOLON-MLS is controlled by commands, which are transferred via the RS232 interface. This chapter describes these commands, their parameters and their functions.

The commands may be issued in any order; several commands may be entered on one line separated by spaces. Parameter settings are stored in non-volatile memory such that the current state will be retained even if power is removed.

Command	Command Group	Page
? <Questionmark>	Information	24
ChanComment	PASSCAL Parameters	30
CHANNELS	Acquisition Control	24
ChanName	PASSCAL Parameters	30
ChanSensor	PASSCAL Parameters	30
ChanSensorNumber	PASSCAL Parameters	30
CLOSE	PCMCIA Module Services	27
COMMENT	Information	24
DCF77	Time Services	26
DRIFT	Time Services	26
END	Acquisition Control	25
EXIT	PASSCAL Parameters	29
ExpComment	PASSCAL Parameters	30
EXPERIMENT	Information	24
ExpName	PASSCAL Parameters	30
ExpNumber	PASSCAL Parameters	29
FORMAT	PCMCIA Module Services	27
LEVELLING	Time Services	26
LICENSE	Information	24
LOAD	Acquisition Control	25
OUT1	Acquisition Control	25
OUT2	Acquisition Control	25
PASSCAL	PASSCAL Parameters	29
PASSCAL?	Information	24
PREVIEW	Acquisition Control	28
RATE	Acquisition Control	25
REC	Acquisition Control	25
RELEASE	Time Services	26
RESYNCHRONIZE	Time Services	26
REPAIR	PCMCIA Module Services	27
RETRIEVE	Accessing Data	28
SETTINGS	Information	24
SHOW	Accessing Data	28
SKEW	Time Services	27
StatComment	PASSCAL Parameters	30
StatName	PASSCAL Parameters	30
StatNumber	PASSCAL Parameters	30

SYNCHRONIZE	Time Services	27

5.1 Help

Each command is accompanied by a short help text line which briefly explains its use. To display the help text, the command must be preceded by a question-mark.

E.g. **?RATE** <cr> displays
<n> RATE sets the effective sampling rate of the system
N may be between 1 and 200

5.2 Information

The following commands display information on the current setting of parameters.

?
displays a list of all commands.

COMMENT <string>"
Defines <string> as a comment. It may be up to 40 characters long. <string> is terminated by either a " or <cr>. The same result is generated by ExpComment in the PASSCAL sub-menue.

EXPERIMENT <string>"
defines <string> as experiment name. It may be up to 24 characters long. <string> is delimited by either a " or <cr>. The same result is generated by ExpName in the PASSCAL sub-menue.

<key1> <key2> <n> LICENSE
activates software option <n>. When LICENSE is executed without parameters it displays the options which have been activated.

PASSCAL?
reports the setting of the PASSCAL description strings (only in PASSCAL menu).

SETTINGS
reports the actual setting of all configurable parameters. See the example in chap. 5.2.

5.3 Acquisition Control

While GEOLON-MLS is in command-mode (before executing the REC command), the A/D converter parameters may be set.

<string> CHANNELS
activates the channels to be recorded. Valid strings for this command are:

seismometer	activates 3 channels for a 3-component seismometer
hydrophone	activates 1 hydrophone channel only
depths	only available for recorder version MLS 13 (for depth sensors). The “depths” command replaces the “hydrophone” command and the “1” command activates the depths sensor channel only
all	activates 3 channels for seismometer and 1 channel for hydrophone

END

finishes a recording sequence. All data buffers are saved on the storage cards and GEOLON-MLS returns to command mode. Now power may be removed without loss of data.

LOAD

loads the file **MLS.SYS** of the storage card in slot 1. This file may contain any sequence of commands and loading this file is equivalent to typing all the information in. The size of the file is limited to 16k max. and it may be generated using a program editor on the PC and, after editing the file, copying it to the storage card.

When the SKEW command is executed, statistics about the recording sequences, time-of-synchronisation etc. will be appended to the command file of **MLS.SYS**.

When a storage card is present in slot 1 upon power-on, **MLS.SYS** will be loaded automatically when it is present.

OUT1

Pin4 of the auxiliary connector. OUT1 OFF places a low-level (0V) on Pin4, OUT1 ON places a high-level (5V) on Pin4.

OUT2

Pin7 of the auxiliary connector. OUT2 OFF places a low-level (0V) on Pin7, OUT2 ON places a high-level (5V) on Pin7. When the RELEASE command is used, OUT2 is used as release output and can not be controlled by ON/OFF.

PREVIEW (remote operation option)

controls the preview queue. PREVIEW ON enables the preview queue which is 128k samples deep. The preview queue stores the most recent data at a decimated sampling rate of 1 Sps. Therefore, the preview queue always holds data of the last 36 hours sampled at 1 Sps. PREVIEW OFF disables the preview queue.

Preview data may be retrieved using the RETRIEVE command.

<n> RATE

determines the sampling rate <n> in samples per second. n may be between 1 to 50 (standard) or between 1 to 200 when the 200_Sps option has been activated.

The -0.1 dB bandwidth of the highest signal frequency which may be reconstructed without aliasing is approximately 1/3 of the sampling rate. The actual sampling rate used may be slightly different from <n> and can be checked using the SETTINGS command.

Please note that for the depth sensor version of MLS (signed by MLS 13 label), only the sampling rates 1, 5 and 10 sps are activated. While the seismometer can be sampled with higher rates (e.g. 50, or 200 sps), the MLS stores the last value on the depth sensor channel sampled by 10 sps.

REC

puts GEOLON-MLS into recording mode using the parameters which have been entered previously and which may be displayed using the SETTINGS command. A new recording sequence is started with its unique initialisation block containing all PASSCAL description strings as well as the time of synchronisation.

5.4 Time Services

Synchronisation is performed using a DCF77 compatible pulse train (see chap. 4.4). During periods of interruption of the external power supply, synchronisation will be maintained by the internal battery (see chap. 4.1). If no battery is present, synchronisation will be lost.

DCF77

synchronises the internal time base. A DCF77 pulse coded signal must be connected to Pin3 (active low) of the auxiliary connector. This synchronisation process may take up to 2 minutes due to the slow DCF77 pulse pattern.

During the synchronisation process following information is displayed:

synchronised - time and date: - <sec> - <min> - <hour> - <day> - <month> - <year>

(NB: No storage cards may be added or removed after time synchronization, otherwise proper operation is not guaranteed. If it is be necessary to add or remove storage cards, the FORMAT, DCF77 or SYNCHRONIZE command must be executed again prior to recording.)

DRIFT

Determines the deviation of the internal clock from DCF77 time. This is similiar to the SKEW command, but does not close the disks and does not write the deviation value to the MLS.SYS file.

EVERY <n1> HOURS <n2> DAYS AFTER <n3> HOURS <n4> DAYS LEVELLING

activates the levelling signal on Pin22 of the seismometer connector <n3> hours and <n4> days after REC has been executed for the first time and repeats it every <n1> hours and <n2> days. The levelling signal is activated (+3.3V output for GEOLON-MLS12 or +12V output for GEOLON-MLS10, see also paragraph 4.2.1) for 20 minutes.

If this command is not used (default), the levelling signal is activated once 3 hours after executing REC without repetition.

<day month year hour minute second> RELEASE

sets the time for the release pulse. When the RELEASE time has been reached, Pin7 of the auxiliary connector is switched to a low-level (0 V). Before the RELEASE time has been reached or if no RELEASE time has been defined, Pin7 remains in its high state (+5V) unless it is explicitly controlled using OUT2 ON/OFF.

RESYNCHRONIZE (remote operation option)

re-synchronises the internal time base using a 1pps pulse on the -DCF77 input (Pin3 of the auxiliary connector). RESYNCHRONIZE may be used during data acquisition without any loss of data. It can be used in remotely operated experiments in order to re-synchronise the internal time base in regular intervals (e.g. once a month).

After resynchronisation, the deviation of the internal time base w.r.t. the synchronisation pulse is reported back and stored as a message. When the experiment is finished later on, these messages will be stored in the MLS.SYS file when the SKEW or CLOSE command is executed. Positive deviation values indicate that the internal time base was early, negative values indicate that it was late w.r.t. the synchronisation pulse.

The 1pps output of a GPS receiver or DCF77 pulses may be used as re-synchronisation pulse. The pulse must be active low, i.e. the high-to-low transition is used as synchronisation edge, and the low-period of the pulse may be at most 250 ms long.

RESYNCHRONIZE is not capable of detecting time deviations above 500 ms. Therefore, resynchronisation has to be performed frequent enough such that the time deviation at the time of resynchronisation remains under 500 ms. Given the long-term stability of the time base used ($5*10^{-8}$), a one month re-synchronisation interval should be safe.

During resynchronisation, OUT1 is activated in order to switch on an external GPS receiver.

SKEW

determines the deviation in milliseconds of the internal oscillator and clock circuitry w.r.t. an external DCF77 compatible signal on the -DCF77 input pin.

After determining the time deviation the data files of the storage cards are adjusted to reflect the actual file length used and the recording parameters and error messages are written to the **MLS.SYS** files on all storage cards.

See also chap. **Fehler! Verweisquelle konnte nicht gefunden werden.** and 6.3.

<day month year hour minute second> SYNCHRONIZE

synchronises the internal clock to an external synchronisation pulse. The pulse must be applied to the -DCF77 input (high-to-low transition). The time and date information entered as numbers before the command will be the time and date to which GEOLON-MLS will be set by the synchronisation pulse.

5.5 PCMCIA Module Services

CLOSE

The data files on the current set of storage cards are adjusted to reflect the actual file length used and the recording parameters and error messages are written to the **MLS.SYS** files on all storage cards (similar to SKEW without time deviation determination).

FORMAT

All PCMCIA slots are checked for the presence of storage cards and their capacity is recorded. After this configuration step has been finished successfully all error messages that may have been stored in the message queue are erased, and the system is ready to record data.

Please note that no storage cards may be added or removed after formatting, otherwise a proper operation is not guaranteed. If it is necessary to add or remove storage cards the FORMAT, DCF77 or SYNCHRONIZE command must be executed again prior to recording.

<n> REPAIR

If a read error has been reported, this command may be used for repair. <n> is the slot number of the affected disk.

5.6 Accessing Data

FROM <day month year hour minute second>
FOR [<n1> SECS] [<n2> MINS] [<n3> HOURS] RETRIEVE <filename> (remote operation option)

transfers recorded data starting at <day month year hour minute second> covering the time span specified in the FOR-clause using the RS232 interface. The data will be stored on the PC in compressed form as <filename>. After retrieval, it can be converted to various formats with the SEND2X software packet.
FROM <day month year hour minute second> has to be specified as six numbers separated by spaces.

PREVIEW RETRIEVE <filename> (remote operation option)

transfers the current content of the preview queue across the RS232 interface and stores it on the PC as <filename> in compressed form. After retrieval, it can be converted to various formats with the SEND2X software packet.

<string> SHOW

displays data of channel <string> on the PC using SENDCOM. SHOW may be used to check the proper operation of the sensor electronics prior to starting an experiment. ^C (ctrl-C) or closing the show window will terminate the command. <string> selects the channel to be displayed:

seismometer1	displays data of seismometer channel 1 (pins 2/14)
seismometer2	displays data of seismometer channel 2 (pins 4/16)
seismometer3	displays data of seismometer channel 3 (pins 6/18)
hydrophone	displays data of hydrophone channel

Note: With the depth sensor version of the MLS, the show command cannot be used to display data received from the depth sensor. For testing the depth sensor, please list the settings: The current pressure value will be displayed after the status line to indicate proper operation and communication with the depth sensor.

Please see the SENDCOM manual for details.

5.7 Definition of PASSCAL Parameters

A number of descriptive character strings can be set which will be stored in non-volatile memory. These strings are stored in every recording sequence and copied into the PASSCAL data file on conversion.

PASSCAL
enters the PASSCAL sub-menu which allows the definition of the following character strings which are defined according to the PASSCAL file standard. These parameters can only be set once for an entire measurement campaign.

EXIT
leaves the PASSCAL sub-menu returning to the main menu.

<n> ExpNumber
sets the Experiment-Number. <n> may be up to 2 digits long.

ExpName <string>"
defines <string> as Experiment-Name. <string> may be up to 24 characters long and it is delimited by either " or a <cr>. The same result is generated by EXPERIMENT.

ExpComment <string>"
defines <string> as Experiment-Comment. <string> may be up to 40 characters long and it is delimited by either " or a <cr>. The same result is generated by COMMENT.

<n> StatNumber
sets the Station-Number. <n> may be up to 4 digits long.

StatName <string>"
defines <string> as Station-Name. <string> may be up to 24 characters long and it is delimited by either " or a <cr>.

StatComment <string>"
defines <string> as Station-Comment. <string> may be up to 40 characters long and it is delimited by either " or a <cr>.

<n> ChanName <string>"
defines <string> as Channel-Name of the <n>th channel. <string> may be up to 10 characters long and it is delimited by either " or a <cr>. <n> may be between 1 and 4.

<n> ChanSensor <string>"
defines <string> as Channel-Sensor model description of the <n>th channel. <string> may be up to 12 characters long and it is delimited by either " or a <cr>. <n> may be between 1 and 4.

<n> ChanSensorNumber <string>"
defines <string> as Channel-Sensor serial Number of the <n>th channel. <string> may be up to 12 characters long and it is delimited by either " or a <cr>. <n> may be between 1 and 4.

<n> ChanComment <string>"
defines <string> as Channel-Comment of the <n>th channel. <string> may be up to 40 characters long and it is delimited by either " or a <cr>. <n> may be between 1 and 4.

6 System Responses

6.1 Interactive Operation

6.1.1 List of Messages

INFORMATION FOR THE USER
System
(0) space key detected
(1) loading *.MLS-file
(2) loading MLS.SYS-file
(5) unrecoverable error, shutdown
Self-check
(11) card-access is stuck, rebooting
(12) REC-command is stuck, rebooting
Error-recovery
(20) resuming recording
(21) finishing recording
(22) trying to restore settings
(23) settings restored, rebooting
(24) boot_delay
(25) abnormal termination, files not adjusted
(26) depth sensor not responding, MLS requested data again

USER-ERRORS
Invalid parameters
(30) range 1 to 12
(32) mask requires numeric input
(33) invalid option number
(34) sample rate not supported
(35) out of range
(36) range 1, 3, or 4
(37) range 1 to 4
Invalid status
(40) not synchronized
(41) not while recording
(42) use only while recording
(43) use only after REC or RESUME
(44) use only after HALT
(45) disks have been closed
(46) has not been licensed
(50) needs a parameter
(51) no card found
(52) can't format any cards
(53) insert disk in slot #1 first
(55) mask can only be used from terminal
(56) channel not active
Out of capacity
(60) cards full
(61) battery low

(62) data not available

UNLIKELY-ERRORS (indicating hardware trouble):
(70) card write error
(71) card read error, rebooting
(72) can't access card, rebooting
(73) directory corrupted, rebooting
(74) repairing bad sector
(75) skipping data due to slow card
(77) ADC-converter not present, rebooting

INTERNAL-ERRORS (Please give information to SEND):
(80) directory overflow
(81) card out of range, rebooting
(82) no more buffers, rebooting
(83) illegal geometry, rebooting
(85) init adc-seconds failed

6.1.2 Meanings of the Messages

(0) If - for any reasons - you have problems communicating with the MLS command-line interface, hold down the SPACE-key while resetting the MLS and then deny to "abort escape-boot-sequence"(type **n**o) and afterwards confirm to "ERASE MLS-STATE"(type **y**es). This will reset all non-volatile variables (except the PASSCAL configuration). If this is of no help, please follow the instructions of chapter 8.1.1.

(1) To update the firmware disconnect power for 5 minutes until the GEOLON-MLS is unsynchronised and put a PCMCIA-card with a firmware-update file (*.mls) in slot #1. The update will be performed in two steps:
1. The version- and ROM-number will be checked whether an update is sensible and possible; in this case the old firmware will be erased.
2. If no firmware is present the new firmware will be loaded. Due to this procedure message (1) appears three times.

(2) To download measurement-programs automatically, remove power for 5 minutes until GEOLON-MLS is unsynchronised and put a PCM-card with measurement-program file (mls.sys) in slot #1. If the previous recording was finished with the SKEW command GEOLON-MLS will attempt to download and execute the measurement-program file (otherwise use LOAD). If no measurement-program is defined and started GEOLON-MLS will enter interactive mode.

(5) After several reboots without writing any data to disk GEOLON-MLS shuts down. This may result from missing or defective PCMCIA-cards, measurement programs with syntax errors or hardware defects.

(11 to 12)
A hang-up of an internal task has been detected caused by either time-out or a reported error-condition. GEOLON-MLS will be reset autonomously and resume recording.

(20) GEOLON-MLS was reset during recording. The record time-window is not yet over, so recording is resumed. Attention: a certain amount of data could be missing. Please inform SEND about this reset.

(21) GEOLON-MLS was reset during recording. The record time-window is over, so recording is finished.

(22) After several resets in vain GEOLON-MLS tries once to rebuild its internal state completely.

(23) system restoration is over, rebooting system.

(24) system reboots cyclic; the delay between the attempts is increasing exponentially to save Flash-cards from reaching their maximum write-count early. Press RETURN anytime to skip such boot delays.

(25) something went wrong during recording, the files won't be truncated to allow to save any data which may be on the disks to a PC for later examination and recovery

(26) depth sensor did not respond to data request, GEOLON-MLS has sent new request

(30) your PCMCIA slot parameter was out of range.

(32) you should only enter numbers and displayed delimiters for the timemask.

(33) Licence key not valid

(34) your rate-parameter was out of range.

(35) time deviation was out of range. Perhaps an indication that synchronisation was lost.

(36) your analog input channel parameter was out of range.

(37) your channel parameter for show function was out of range

(40) the command you entered needs a synchronized timebase (via DCF77).

(41) you can't change record-parameter while recording.

(42) there's no recording running which could be ended, halted, or resumed.

(43) the current recording is already halted.

(44) the current recording is already started or resumed.

(45) you can't record any more after you executed SKEW.

(46) you did not license this feature

(50) refer to command description for syntax-details.

(51) you should insert any cards before entering DCF77 or SYNCHRONIZE.

(52) login of any cards failed due to old unsaved data on some disks. Execute DCF77 or SYNCHRONIZE manually to find the offending card.

(53) enter a card in slot #1 to download measurement programs via LOAD**.**

(55) you can't read any user entry within a measurement program.

(56) you can't watch data from a channel, which is not activated by CHANNELS.

(60) all disks are full of data, you can't record anything more.

(61) detection of low voltage: the system will interrupt recording immediately and store data left in the buffers including the directory before entering sleep mode until the voltage is in nominal range again.

(62) a date and time has been specified for the RETRIEVE command for which no data is available on the current set of storage cards.

(70) A write error of the PCMCIA-card occured. You may have reached the end of the lifecycle of your card. Check your PCMCIA-card on the PC with a vendor-specific service-program.

(71) A read error of the PCMCIA-card occured. This typically happens after a sudden powerfailure during writing. GEOLON-MLS itself will try automatically to repair such sectors after it encountered them. If this fails you get this message, probably the controller of the card has crashed, otherwise the card itself may be worn out.

(72) Very unlikely read error of the PCMCIA-card's identify sector. The card may have hung up due to hot plugging or bad contact. Eject and insert PCMCIA-card, reset GEOLON-MLS and try again.

(73) Invalid data in the assumed directory block. This error will occur if you change PCMCIA-cards after formatting. Reformat disks.

(74) Trying to repair a sector with wrong CRC-sum, this might lead to further trouble if the directory block was concerned.

(75) the write-performance of the PCMCIA-card is not sufficient to record all data. GEOLON-MLS switches to halt temporarily to provide big blocks of valid data (instead of skipping single samples).

(77) no data provided by the A/D-module, check whether it is attached firmly.

(80) you exceeded the maximum of 8000 recordings per PCMCIA-card, this is probably due to a corrupted directory.

(81) The software tries to access non-existent slots or sectors. This error will occur if you change PCMCIA-cards after formatting. Reformat disks.

(82) The system does not provide enough buffers. This error should never occur. Reload firmware.

(83) Illegal sector, heads or cylinder numbers in the identify sector. The card may have hung up due to hot plugging or bad contact. Eject and insert PCMCIA-card, reset GEOLON-MLS and try again.

(85) Please contact SEND support

6.2 Stand-alone Operation

Operation without PC is supported by the green LED in the front plate between the RS232 and auxiliary connector. The meanings of the blink signals are as follows:

Signal	Meaning of signal
————————	(continuously ON) ready for operation
·_ _ ·_ _ ·_ _	(Morse alphabet: W) waiting for synchronise pulse
··· ··· ···	(Morse alphabet: S) waiting for DCF77 synchronisation
··_· ··_·	(Morse alphabet: F) failure: old data on disk which are not saved
_ _ _ _ _ _	(Morse alphabet: O) formatting finished (ok); signal continues until next command causes change of signal
flickering	access to storage cards
·····-·····	(continuously short, twice a second) recording
_ _ _ _ _ _ _ _ _ _	(continuously long, once a second) halt during recording

6.3 Error Indication

If an error occurs during a measurement campaign, an error message will be stored in non-volatile memory. Each message includes the time and date, the task in which the error occurred and a brief error description.

If GEOLON-MLS is reset or if it is switched on after an error, an error report is displayed following the sign-on message. After the report, the user may choose to erase the report or retain it in non-volatile memory for a later time by answering "y" or "n" to the prompt.

When the CLOSE or SKEW command is executed upon termination of an experiment, all error messages are displayed and appended to the file **MLS.SYS** on the storage card in slot 1.

7 Description of a Typical Operation Session

7.1 Interactive Operation via External PC

7.1.1 Software Preparation

Install the software from the disk delivered with GEOLON-MLS on your PC. The different programs are described in chap. 3. **SENDCOM** is used for communication via the RS232 interface, e.g. for interactive configuration of GEOLON-MLS. **SEND2X** is used for conversion of recorded data into several formats.

7.1.2 Hardware Preparation

Due to the modular design of GEOLON-MLS with regard to the sensor interface (e.g. seismometer, hydrophone or pressure sensor etc), the user has to check whether GEOLON-MLS is equipped with the correct analog input modules. Usually the front plate should be labelled accordingly. If not, the instrument has to be opened as described in chap. 2.3 in order to get access to the analog PCB. The codes for the different sensor specific A/D converters are listed in chap. 4.2. Please make sure that the correct PCBs for seismometer and if necessary, for a hydrophone or a pressure sensor are installed. If a hydrophone or a pressure sensor is to be used the switches for the preamplifier gain preselection have to be set according to chap. 4.2.2.

After closing the front plate as described in chap. 2.3 the cable connections can be installed:

Connect GEOLON-MLS to the host-PC for interactive operation via the RS232 interface by means of a 9-pin modem extension cable.

Connect GEOLON-MLS to the external power supply specified in chap. 4.1.

(Note: The capacity of the internal battery is not calculated for long operations. Therefore, it is recommended to use an external power supply whenever possible.)

Connect the auxiliary signals necessary for your measurement project to the interface described in chap. 4.4. Connect at least the input for the time synchronisation (Pin3).

7.1.3 Preparation of a Measurement Campaign

The necessary storage cards have to be inserted into the slots of GEOLON-MLS always starting at slot 1. The capacities of the storage cards should be sufficient for the predicted duration of the measurement campaign; otherwise the recording will be prematurely terminated.

(Note: The storage cards will always be repartitioned and reformatted during initialisation on GEOLON-MLS to ensure a defragmented file-system. GEOLON-MLS will preserve the integrity of the MLS.SYS configuration file.)

A measurement campaign is prepared by using the commands described in chap. 5. In particular the following parameters must be set:

- sampling rate
- definition of sensor types used
- levelling condition
- sensor specific parameters (if required)
- PASSCAL character strings for experiment documentation

Configuration data are stored in non-volatile memory and remain available even after power down.

Before starting the data recording a time synchronisation according to chap. 5.5 has to be executed. The internal battery assures that GEOLON-MLS will not lose its time synchronisation while external power has been removed during experiment preparation.

Removal, exchange, or adding of storage cards is not allowed after formatting (because GEOLON-MLS holds a copy of the disk parameter block and the disk state in its non-volatile memory). If it should be necessary to make any PCMCIA-card changes, you have to repeat the time synchronisation by either the command DCF77 or SYNCHRONIZE (which both include the disk-initialisation), or by means of the FORMAT command, before any data recording can be started. The benefit of this process is that all PCMCIA-cards, which were used during a recording session, contain a unique ID consisting of a timestamp and the serial number of the specific GEOLON-MLS used.

The REC command puts GEOLON-MLS into recording mode, using the parameters defined before or the settings of the last recording (default values), respectively.

7.1.4 Closing an autonomous recording session

The autonomous recording session is described in chap. 7.2. The formal closing is executed interactively after having connected DCF77 to GEOLON-MLS and GEOLON-MLS to the PC again.

The SKEW command determines the deviation of the internal oscillator compared to a DCF77 signal on the -DCF77 input pin. After determining the deviation, the data files on the current set of storage cards are adjusted to reflect the actual file length used and the recording parameters and error messages are written to the **MLS.SYS** file on each card.
Thereafter, the cards may be removed.

7.2 Stand-Alone Operation without PC

GEOLON-MLS is designed for stand-alone data recording following a previously defined configuration and parameterization, which can be performed either interactively as described above or by using **MLS.SYS** on the storage card in slot 1.

After executing the preparatory tasks as described in chap. 7.1.3, GEOLON-MLS may be installed in the fixture for stand-alone operation, and external power should be connected as soon as possible due to the limited capacity of the internal batteries. Also the measurement sensors have to be connected to the activated and configured analog input channels. If a hydrophone with preamplifier (e. g. LOWN22, see note in chap. 4.2.2) is used, only cables with a grey connector shell should be used.

Data recording will be performed according to the actual parameters or until either the storage cards are full or the battery is discharged to the low-voltage threshold. In any case, GEOLON-MLS will finish the recording, writing all data remaining in internal buffers on the PCMCIA-disks before it shuts down. The power consumption during shutdown is reduced to a minimum because only the internal timebase is maintained. This normally allows you to execute the SKEW command later on.

Operation without PC is supported by the green LED on top between the connectors for RS232 and Auxiliaries. The meanings of the blink signals are described in chap.6.2.

7.3 Analysis of Recorded Data

After data recording, the storage cards can be played back using a PC with PCMCIA interface and the SEND2X software. This software generates several file formats on any mass storage device of the PC and thus enables analysis and evaluation of recorded data by means of any suitable program.

7.4 Brief Description for Experienced Users

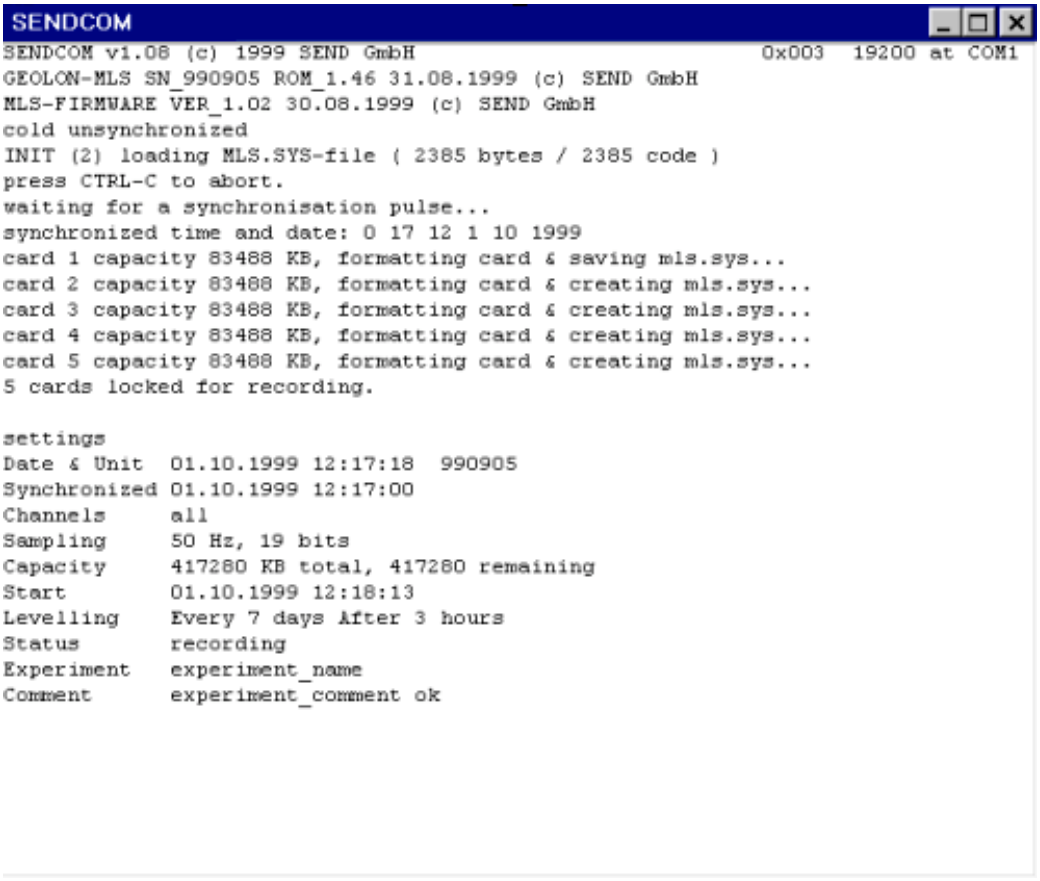
For users who have already experience with the operating instructions as described in chapters 7.1 to 7.3 the following short description may be a useful; it may also be used as a check-list.

1. Insert internal batteries
2. Plug in all PCMCIA modules
3. Connect DCF77
4. Connect external power
5. **MLS.SYS** file is read;
6. Standard contents: **ALL CHANNELS**
   50 RATE
   DCF77
   REC

1. Wait until synchronisation is completed and recording starts
2. (LED signal: continuous short)
3. Disconnect external power and DCF77.
4. The MLS is now ready for independent handling.
5. Carry out all installation work required for experiment preparation.
6. During this time, the MLS is powered by its internal batteries.
7. Switch on external power: REC is restarted.
8. Time counting for levelling starts from 0 (again).
9. Normally now the mission is executed.
10. After completion or interruption of the experiment, GEOLON-MLS is recovered.
11. Connect DCF77.
12. Issue END and SKEW.

7.5 Examples for Interactive Operation

7.5.1 Interactive Preparation of an Autonomous Recording Session



1 Response of PC-terminal software (Release-Nr.)

2 Loading **MLS.SYS**-file from the PCMCIA card in slot 1; it contains commands and parameters

3 Response after switch-on GEOLON-MLS (Series-Nr. and installed firmware)

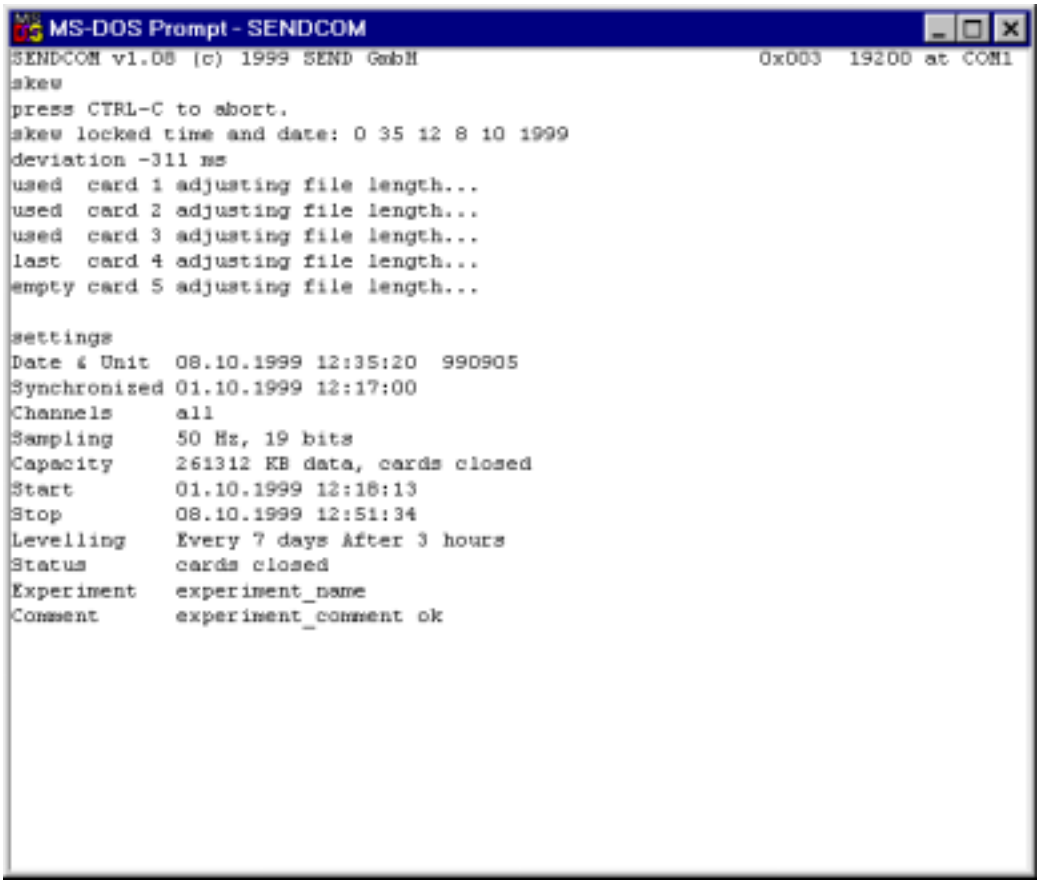
4 Response after synchronisation pulse

5 Formatting of all inserted PCMCIA-cards (5 cards are inserted)

6 Command SETTINGS followed by a report of actual parameters; this command is issued here just to display the parameters. For the MLS version adapted for depth sensors, the current pressure value of the depth sensor will be displayed after the status line.

7 Since the **MLS.SYS**-file (3) contains the REC command (Status: Recording), recording started automatically

7.5.2 Closing a Recording Session Interactively



1 To close a recording session GEOLON-MLS is connected to the PC and DCF77 is connected to GEOLON-MLS. The SKEW command is issued.

2 SKEW time is fixed and time deviation is calculated.

3 PCMCIA cards are closed.

4 The command SETTINGS followed by a report of actual parameters.

8 Loading a New Software Release

Whenever a new software release of "MLS" firmware becomes available it will be delivered as an ASCII-file with the filename <3-digit-type><3-digit-release>.MLS, e.g. MLS132.MLS. This stands for "software for the longtime-seismology unit, release 1.32".

This file must be copied to a storage card on the PC. Then the storage card must be inserted into slot 1 of GEOLON-MLS and power must be switched on. The software will be loaded in two steps:

- The *.MLS-file will be read off the card and stored in internal memory. During the process, the LED is blinking twice per second.
- The software will be compiled and stored into non-volatile program memory. During the process, the LED is blinking once per second.

As soon as the LED stops blinking the new software release has been successfully loaded. After phase 1 is finished, the storage card may be removed from slot 1. Before updating the firmware, the *.mls-file integrity is checked using a CRC; if it fails no update is performed.

9 Troubleshooting

9.1 Emergency firmware erase for booting problems

In very rare cases, GEOLON-MLS may get into a state in which it will not respond to the RS232 command interface, even after reset. Then the (faulty) firmware may be erased using the following procedure:

- Disconnect power.
- Connect pins 3 and 4 of the auxiliary connector by e.g. using a paper clip.
- Connect power and wait for 10 seconds.
- Disconnect power again and remove the connection between pins 3 and 4.
- Reconnect power.

Now GEOLON-MLS should boot with the ROM operating system sign-on message and new firmware can be loaded.

9.2 MLS not responding/hanging

In very rare cases, GEOLON-MLS might not return from a command issued, e.g. when trying to synchronize or skew with a faulty time pulse. In this case, power should be disconnected for a period of at least 5 minutes. Please note that this may affect the accuracy of your recorded data, since it is not possible to determine the skew once synchronization has been lost.

10 Support and Service

If any problem with GEOLON-MLS should arise, or if you need any support for operating it, please contact SEND GmbH via

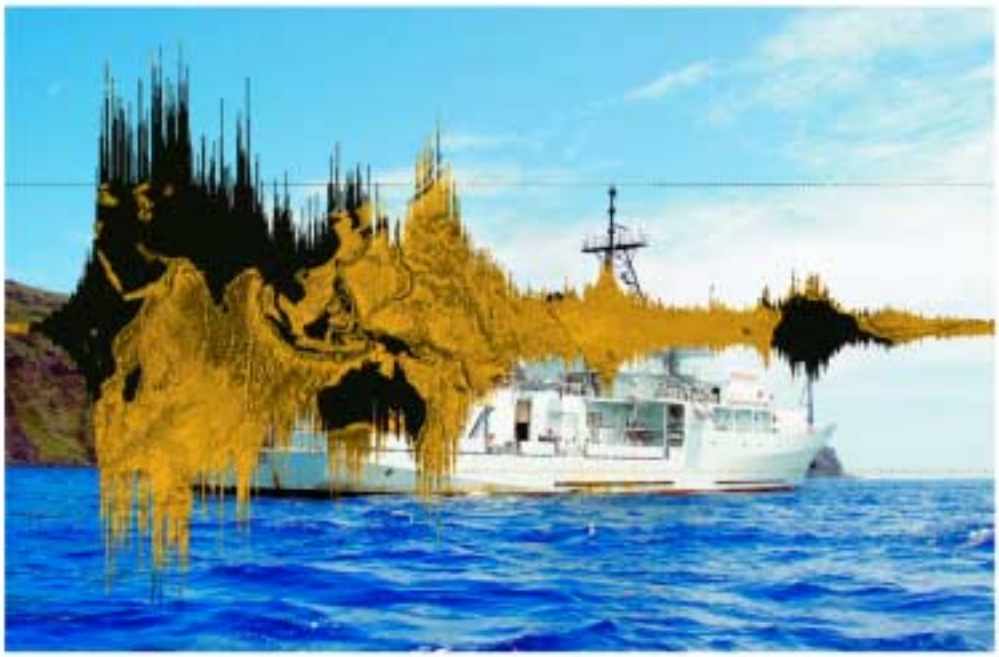
e-mail: support@send.de

fon: +49 40 375 008 03

fax: +49 40 375 008 93

11 Appendix: SEND2X manual

SEND2X



Data handling and conversion suite for Geolon Recorders

SEND Signal Elektronik GmbH

Rostocker Str. 20
D-20099 Hamburg
e-mail: office@send.de
Tel: +49 40 375008-03
Fax: +49 40 375008-93

Version 2.61

This program-package converts the compressed recordings of the SEND data loggers into different formats. SEND2X **version 2.60** is available for the Linux operating system.

This version allows the conversion of raw data into a binary file, an audio-wave file, a special Reftek SEG-Y format for seismological data and into the standard SEG-Y format if an appropriate shot file is available.

1.1 Description of individual programs 5
1.1.1 mcscopy 5
1.1.2 mescopy 5
1.1.3 mcslog 6
1.1.4 meslog 6
1.1.5 mcsread 6
1.1.6 mesread 8
1.1.7 mbsread 10
1.1.8 mlsread 12
1.1.9 mtsread 14
1.1.10 smtsread 16
1.1.11 mcsupload 16
1.1.12 mesupload 17
1.1.13 binwrite 17
1.1.14 wavewrite 18
1.1.15 seg-ywrite 18
1.1.16 gsewrite 21
1.1.17 paswrite 23
1.1.18 sacwrite 25
1.1.19 resample 27
1.1.20 pressure2ascii 28
1.2 Combining the programs 28
1.2.1 Structure of the shot file 29
1.3 Files 32
1.3.1 Log - File of Geolon-MES and Geolon MCS 32
1.3.2 Engineering data 32
1.3.3 Data Files 32

This manual includes the description of utilities, which are required to save and process the data of SEND's data loggers Methusalem-MBS, Geolon-MCS, Geolon-MES, Geolon-MLS and Geolon-MTS.

On the delivered CD-R you will find a program-library and a script, which includes an example of combining the different programs. Each of the programs can be used separately.

The library consists of the device specific files

for the Geolon-**MCS**:

mcscopy, mcslog, mcsread and mcsupload

for the Geolon-**MES**:

mescopy, meslog and mesread

for the Geolon-**MLS**:

mlsread

for the Geolon-**MTS/MTS-M**:

mtsread, smtsread and pressure2ascii

for the Geolon-**MBS**:

mbsread

The device-dependent programs can only be used while working with the corresponding device and the data retrieved can then be further processed with the following device independent programs:

binwrite

wavewrite

seg-ywrite

gsewrite

paswrite

sacwrite

resample

All programs use the standard input as input device, and the standard output as output device most of the time. This allows an easy combination of the different programs by 'pipeing'.

Specific Linux issues

To allow handling of files larger than 20 Gbytes and more, please format the harddisk of your Linux computer with the maximum cluster and sector size. Also, depending on your Linux distribution, it might be necessary to work with root-privileges when performing hardware related tasks, e.g. retrieving data from IEEE1394 drives utilized by the MES data recorder.

Installing the programs

The programs can simply be copied from the distribution CD-ROM to the harddrive of your Linux system, e.g. to /usr/send2x/bin. It is recommended that you append your path-variable with the path to the send2x files. Example: ***PATH = $PATH:/usr/send2x/bin***

1.1 Description of individual programs

In the following description, the symbols

< > will be used to include variables which should be entered.

< > are also used as symbols for standard in- and output as part of the command.

The option **`--help`** can be used for all listed programs to display a brief description of the program on the screen.

1.1.1 mcscopy

This program module copies all recorded data from Geolon-MCS's harddisk via standard input and standard output to the harddisk of your computer. For the execution of mcscopy, the standard input has to be assigned to the device path and file of the Firewire interface, which is directly connected to the MCS's harddisk (e.g. /dev/*devicefile*). Also, the standard output has to be assigned to the destination directory and file on your computer. We recommend '.raw' as extension for the data file on your harddisk. For example:

`mcscopy < /dev/sda > <`*`working_directory/data_file_name.raw`*`>`

In the case of a corrupted data file (e.g. due to an unclosed file, system crash or reboot of the MCS while recording data), mcscopy will not be able to determine the correct size of the data file. Therefore, mcscopy displays a comment, requesting to enter an estimated file size and to restart mcscopy using the option:

`--save` *`<file size in Mbytes>`*

The file size can be estimated using the following calculation:

<File size in MBytes> =<sample rate in Hz> * 0.33 * <recording days> * <active channels> +50

1.1.2 mescopy

This program module copies all recorded data from Geolon-MES's harddisk via standard input and standard output to the harddisk of your computer. For the execution of mescopy, the standard input has to be assigned to the device path and file of the Firewire interface, which is directly connected to the MES's harddisk (e.g. /dev/*devicefile*). Also, the standard output has to be assigned to the destination directory and file on your computer. We recommend '.raw' as extension for the data file on your harddisk. For example:

`mescopy < /dev/sda >` *`<working_directory/data_file_name.raw>`*

In the case of a corrupted data file (e.g. due to an unclosed file, system crash or reboot of the MES while recording data), mescopy will not be able to determine the correct size of the data file. Therefore, mescopy displays a comment, requesting to enter an estimated file size and to restart mescopy using the option:

`--save` *`<file size in Mbytes>`*

The file size can be estimated using the following calculation:

<File size in MBytes> =<sample rate in Hz> * 0.33 * <recording days> * <active channels> +50

1.1.3 mcslog

Mcslog displays the MCS-log file on the screen of your PC. The log file includes the settings, control, status, and identification information of all experiments, which were carried out since the last disc format (synchronization). To copy the log file to your computer instead of displaying it on the screen, the standard output has to be redirected to a file on your PC. For executing mcslog, the standard input has to be assigned to the device path and file of the Firewire interface, which is connected to the MCS.

For example:

`mcslog < /dev/sda`

displays the log-file on the screen and

`mcslog < /dev/sda >` *<directory/log_file_name.txt>*

copies the log file to the harddisk of your computer.

1.1.4 meslog

Meslog displays the MES-log file on the screen of your PC. The log file includes the settings, control, status, and identification information of all experiments, which were carried out since the last disc format (synchronization). To copy the log file to your computer instead of displaying it on the screen, the standard output has to be redirected to a file on your PC. For executing meslog, the standard input has to be assigned to the device path and file of the Firewire interface which is connected to the MES.

For example:

`meslog < /dev/sda`

displays the log-file on the screen and

`meslog < /dev/sda >` *<directory/log_file_name.txt>*

copies the log file to the harddisk of your computer.

1.1.5 mcsread

This program module reads, decompresses and converts the raw data (recorded by Geolon-MCS) into the internal send2x format. In addition, it generates an ASCII file, containing the acquired engineering data (for a detailed description see: "1.3 Files").

Again, the program can read the compressed raw data via standard input through the Firewire interface from the MCS's harddisk. To do this, the standard input has to be assigned to the Firewire device connecting the MCS to the PC. Alternatively, the standard input can be assigned to the data file, which was stored by mcscopy on the harddisk of your computer. The converted data will be transferred to the standard output. Again, the standard output can be assigned to a destination file.

For example:

`mcsread < <raw_data_ file> >` *<directory/file_name.s2x>*

Because the decompression and conversion of the raw data needs more processing time than the simple copy procedure, we recommend to use mcscopy at first and then to use mcsread as the next processing step.

mcsread includes the following options (values in <> are variables which must be entered):

--begin_sec *<integer in GPS seconds>*

or

--begin_date *<formatted_time_string_yyyy.mm.dd_hh:mm:ss>*

together with

--end_sec *<integer in GPS seconds>*

or

--end_date *<formatted_time_string_yyyy.mm.dd_hh:mm:ss>*

Using these options restricts the data conversion to the time-window of interest. The resulting file will contain only data within this time window. We recommend to use the time of the first and the last shot of a survey-line.

GPS seconds means the number of seconds since GPS start on January 6th 1980, 00:00 o'clock.

Examples:

--begin_sec 708669934 --end_sec 708685565

or

--begin_date 2002.06.21_04:45:34 --end 2002.06.21_09:06:05

--skew *<+/-_skew_time_in_µs>*

This option allows to store a manually entered skew value in the send2x file instead of the skew value from the raw data file.

Example: **--skew -1000**

Instead of the skew value of the raw data file, a skew value of -1 ms will be recorded in the send2x file.

--timeshift *<+/-value_in_seconds>*

The use of this option allows to shift all data by a constant time. The value has to be in full seconds (integer value). Use this command for the correction of differences in time, if different time standards have been used for synchronising the data logger and for synchronising the airgun controller on the shooting vessel.

Example: GPS-RTS delivers the time in UTC, the shooting vessel provides the shot time as GPS time. The difference between these two time standards is 14 seconds and therefore, you have to enter:

`--timeshift 14`

`--no_eng`

The engineering data will be stored automatically in a file in the current working directory with a filename according to chapter 1.3.2.

The option **`--no_eng`** suppresses the automatic file generation.

`-- log` *<filename>*

This option generates a log-file of the current mcsread-run in the current working directory. Without using the option, the log-file will be written automatically to the file "mcsread.log".

`--no_log`

No log file will be generated.

1.1.6 mesread

This program module reads, decompresses and converts the raw data (recorded by Geolon-MES) into the internal send2x format. In addition, it generates an ASCII file, containing the acquired engineering data (for a detailed description see: "1.3 Files").

Again, the program can read the compressed raw data via standard input through the Firewire interface from the MES's harddisk. To do this, the standard input has to be assigned to the Firewire device connecting the MES to the PC. Alternatively, the standard input can be assigned to the data file, which was stored by mescopy on the harddisk of your computer. The converted data will be transferred to the standard output. Again, the standard output can be assigned to a destination file.

For example:

`mesread <` *<raw_data_ file>* **`>`** *<directory/file_name.s2x>*

Because the decompression and conversion of the raw data needs more processing time than the simple copy procedure, we recommend to use mescopy at first and then to use mesread as the next processing step.

mesread includes the following options (values in <> are variables which must be entered):

`--begin_sec` *<integer in GPS seconds>*

or

`--begin_date` *<formatted_time_string_yyyy.mm.dd_hh:mm:ss>*

together with

`--end_sec` *<integer in GPS seconds>*

or

--end_date *<formatted_time_string_yyyy.mm.dd_hh:mm:ss>*

Using these options restricts the data conversion to the time-window of interest. The resulting file will contain only data within this time window. We recommend to use the time of the first and the last shot of a survey-line. Mesread internally subtracts 60 seconds from 'begin time' and adds 60 seconds to 'end time' (the time window thus becomes: begin-60secs till end+60secs).

GPS seconds means the number of seconds since GPS start on January 6th 1980, 00:00 o'clock.

Examples:

--begin_sec 708669934 --end_sec 708685565

or

--begin_date 2002.06.21_04:45:34 --end 2002.06.21_09:06:05

--skew *<+/-_skew_time_in_µs>*

This option allows to store a manually entered skew value in the send2x file instead of the skew value from the raw data file.

Example: **--skew -1000**

Instead of the skew value of the raw data file, a skew value of -1 ms will be recorded in the send2x file.

--timeshift *<+/-value_in_seconds>*

The use of this option allows to shift all data by a constant time. The value has to be in full seconds (integer value). Use this command for the correction of differences in time, if different time standards have been used for synchronising the data logger and for synchronising the airgun controller on the shooting vessel.

Example: GPS77 delivers the time in UTC, the shooting vessel provides the shot time as GPS time. The difference between these two time standards is 14 seconds and therefore, you have to enter:

--timeshift 14

--no_eng

The engineering data will be stored automatically in a file in the current working directory with a filename according to chapter 1.3.2.

The option **--no_eng** suppresses the automatic file generation.

-- log *<filename>*

This option generates a log-file of the current mesread-run in the current working directory. Without using the option, the log-file will be written automatically to the file "mesread.log".

--no_log

No log file will be generated.

1.1.7 mbsread

Mbsread has to be used to convert raw data acquired by the Methusalem-MBS data logger. At first, the raw data of all PCMCIA cards of one recording session as well as one of the MBS.SYS files must be copied to a directory using e.g. the cp command. Then mbsread is used to decompress and convert the raw data into the internal send2x format.

Mbsread reads the raw data from either the current directory or from the directory specified by the --src option. The converted data will be transferred to the standard output, which usually will be re-directed to a destination file.

For example:

mbsread --src *<directory>* **>** *<directory/file.s2x>*

mbsread includes the following options (values in <> are required variables):

--src *<directory>*

specifies the directory, which is used as raw data input. --src may be omitted if the input comes from the current directory. The directory must contain all raw data files, which may have been copied from multiple PCMCIA cards belonging to one recording session as well as the MBS.SYS file from one of the PCMCIA cards.

--help displays a list of all options with brief explanations

--begin_sec *<integer_in_GPS_seconds>*

or

--begin_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

--end_sec *<integer_in_GPS_seconds>*

or

--end_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

Using these options restricts the data conversion to the time-window of interest. The resulting file will contain only data within this time window. We recommend to use the time of the first and the last shot of a survey-line. Mbsread internally subtracts 60 seconds from 'begin time' and adds 60 seconds to 'end time' (the time window thus becomes: begin-60secs till end+60secs).

GPS seconds means the number of seconds since GPS start on January 6th 1980, 00:00 o'clock.

Example:

--begin_sec 708669934 --end_sec 708685565

or

--begin_date 2002.06.21_04:45:34 --end 2002.06.21_09:06:05

--skew *<+/-_skew_in_µs>*

This option allows to store a manually entered skew value in the send2x file instead of the skew value from the raw data file(s).

Example: **`--skew -1000`**

Instead of the skew value of the raw data file, a skew value of -1 ms will be recorded in the send2x file.

`--timeshift` *<+/-value_in_seconds>*

The use of this option allows to shift all data by a constant time. The value has to be in full seconds (integer value). Use this command for the correction of differences in time, if different time standards have been used for synchronising the data logger and for synchronising the airgun controller on the shooting vessel.

Example: GPS77 delivers the time in UTC, the shooting vessel provides the shot time as GPS time. The difference between these two time standards is 14 seconds and therefore, you must enter:

`--timeshift 14`

`--log` *<filename>*

This option generates a log file of the current mbsread run in the current directory. Without using this option, the log file will be written "mbsread.log".

`--no_log`

No log file will be generated.

`--nc`

If this option is used, no messages will be produced when mbsread is executed.

1.1.8 mlsread

Mlsread has to be used to convert raw data acquired by the GEOLON-MLS data logger. At first, the raw data of all PCMCIA cards belonging to one recording session as well the corresponding MLS.SYS file must be copied to a directory using e.g. the cp command. Then mlsread is used to decompress and convert the raw data into the internal send2x format.

Mlsread reads the raw data from either the current directory or from the directory specified by the --src option. The converted data will be transferred to the standard output, which usually will be re-directed to a destination file.

For example:

mlsread --src *<directory>* **>** *<directory/file.s2x>*

mlsread includes the following options (values in <> are required variables):

--src *<directory>*

specifies the directory, which is used as raw data input. --src may be omitted if the input comes from the current directory. The directory must contain all raw data files, which may have been copied from multiple PCMCIA cards belonging to one recording session as well as the MLS.SYS file from one of the PCMCIA cards.

--help or **-h** displays a list of all options with brief explanations

--begin_sec *<integer_in_GPS_seconds>*

or

--begin_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

--end_sec *<integer_in_GPS_seconds>*

or

--end_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

Using these options restricts the data conversion to the time-window of interest. The resulting file will contain only data within this time window. We recommend to use the time of the first and the last shot of a survey-line. Mlsread internally subtracts 60 seconds from 'begin time' and adds 60 seconds to 'end time' (the time window thus becomes: begin-60secs till end+60secs).

GPS seconds means the number of seconds since GPS start on January 6th 1980, 00:00 o'clock.

Example:

--begin_sec 708669934 --end_sec 708685565

or

--begin_date 2002.06.21_04:45:34 --end 2002.06.21_09:06:05

--skew *<+/-_skew_in_µs>*

This option allows to store a manually entered skew value in the send2x file instead of the skew value from the raw data file(s).

Example: **`--skew -1000`**

Instead of the skew value of the raw data file, a skew value of -1 ms will be recorded in the send2x file.

`--timeshift` *<+/-value_in_seconds>*

The use of this option allows to shift all data by a constant time. The value has to be in full seconds (integer value). Use this command for the correction of differences in time, if different time standards have been used for synchronising the data logger and for synchronising the airgun controller on the shooting vessel.

Example: GPS77 delivers the time in UTC, the shooting vessel provides the shot time as GPS time. The difference between these two time standards is 14 seconds and therefore, you must enter:

`--timeshift 14`

`--log` *<filename>*

This option generates a log file of the current mlsread run in the current directory. Without using this option, the log file will be saved as "mlsread.log".

`--no_log`

No log file will be generated.

`--nc`

If this option is used, no messages will be produced when mlsread is executed.

1.1.9 mtsread

Mtsread has to be used to convert raw data acquired by the GEOLON-MTS/MTS-M data logger. At first, the raw data of all PCMCIA cards belonging to one recording session as well as the corresponding MLS.SYS file must be copied to a directory using e.g. the cp command. Then mtsread is used to decompress and convert the raw seismic data into the internal send2x format.

The raw data of the absolute pressure gauge will be converted into ASCII format and stored in a separate file. The name of this file will be generated automatically with the channel number 5 and the extension **.pressure** (see also chap. 1.3.3).

The resulting ASCII file looks like this:

```
************************************************************
time                    pressure (mBar)
------------------------------------------------------------
13:30:13 16.08.2005      1088.6
13:30:28 16.08.2005      1088.7
13:30:43 16.08.2005      1088.8
13:30:58 16.08.2005      1088.9
.................          ...
************************************************************
```

Mtsread reads the raw data from either the current directory or from the directory specified by the --src option. The converted data will be transferred to the standard output, which usually will be re-directed to a destination file.

For example:

mtsread --src *<directory>* **>** *<directory/file.s2x>*

mtsread includes the following options (values in <> are required variables):

--src *<directory>*

specifies the directory, which is used as raw data input. --src may be omitted if the input comes from the current directory. The directory must contain all raw data files, which may have been copied from multiple PCMCIA cards belonging to one recording session as well as the MLS.SYS file from one of the PCMCIA cards.

--help displays a list of all options with brief explanations

--begin_sec *<integer_in_GPS_seconds>*

or

--begin_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

--end_sec *<integer_in_GPS_seconds>*

or

--end_date *<formatted_string_yyyy.mm.dd_hh:mm:ss>*

Using these options restricts the data conversion to the time-window of interest. The resulting file will contain only data within this time window. We recommend to use the time of the first and the last shot of a survey-line.

GPS seconds means the number of seconds since GPS start on January 6th 1980, 00:00 o'clock.

Example:

--begin_sec 708669934 --end_sec 708685565

or

--begin_date 2002.06.21_04:45:34 --end 2002.06.21_09:06:05

--skew *<+/-_skew_in_µs>*

This option allows to store a manually entered skew value in the send2x file instead of the skew value from the raw data file(s).

Example: **--skew -1000**

Instead of the skew value of the raw data file, a skew value of -1 ms will be recorded in the send2x file.

--timeshift *<+/-value_in_seconds>*

The use of this option allows to shift all data by a constant time. The value has to be in full seconds (integer value). Use this command for the correction of differences in time, if different time standards have been used for synchronising the data logger and for synchronising the airgun controller on the shooting vessel.

Example: Your GPS receiver delivers the time in UTC, the shooting vessel provides the shot time as GPS time. The difference between these two time standards is 14 seconds and therefore, you must enter:

--timeshift 14

--log *<filename>*

This option generates a log file of the current mtsread run in the current directory. Without using this option, the log file will be stored in a file with the name "mtsread.log".

--no_log

No log file will be generated.

--nc

If this option is used, no messages will be produced when mtsread is executed.

`--report_slips`

This option starts a reporting of time slip messages on the screen during the conversion of the data.

For a description of time slips, please see the chap. "Time Slips" in the MTS Manual.

1.1.10 smtsread

This programs converts data sequences that have been requested from a MTS-M tsunameter seismocorder in an ocean bottom unit (OBU) by the buoy computer´s OBUD program. It works similar to mtsread, just without mls.sys file information, which is replaced by the header information in the data sequence file.

Data in- and output can be pipes or files only, with no additional parameters:

EXAMPLE:

smtsread </data/inputfilename >/data/outputfilename

1.1.11 mcsupload

This program allows uploading of user programs to the Geolon-MCS via the IEEE1394-interface. The order of options is not important. Options are.

--usp <*filename*>

This option defines the filename of the user program if StdIn is not used.

--log <*filename*>

This option defines the name of the logfile. If this option is not given, “mcsupload.log” will be used by default.

EXAMPLES:

mcsupload </data/user.mcs >/dev/*mcsdevice*

mcsupload –usp /data/user.mcs >dev/*mcsdevice*

mcsupload –p /var/log/mcs/mcsupload.log –usp /data/user.mcs >dev/*mcsdevice*

1.1.12 mesupload

This program allows uploading of user programs, new operating system software and new firmware versions to the Geolon-MES via the IEEE1394-interface. Options are.

--usp <*filename*>

This option defines the filename of the user program if StdIn is not used.

--rom <*filename*>

This option defines the filename of the operating system software to be loaded.

--fw <*filename*>

This option defines the filename of the firmware to be loaded.

EXAMPLES:

mesupload </data/user.mes >/dev/*mesdevice*

mesupload --usp /data/user.mes >dev/*mesdevice*

mesupload --fw /data/mes113a_fw.mes >dev/*mesdevice*

Please note, that new operating system software and new firmware will not be installed automatically by the MES recorder. Please consult the MES manual for details, how to store these files from the internal harddrive to nonvolatile memory.

1.1.13 binwrite

This program has to be used to convert the data from internal send2x format into 32 bit binary format. It reads the data via standard input and writes for each activated channel and each recording period (sequence) a file into a destination directory. The filenames are generated automatically. Corresponding to chapter 5.3.3, they consist of the recorder's serial number, channel no, sequence no and the date and time of recording-start. Furthermore, each file will be marked by the extension '.bin'. Following options are important:

`--dest` *`<path/destination directory/>`*

This option defines the destination directory.

`--no_head`

The application of no_head suppresses the text header in the output file. Otherwise, a header of 1024 Byte will be written.

1.1.14 wavewrite

As binwrite, wavewrite also reads the data from standard input. But now, the data will be converted into a 16 bit PCM wave (audio wave) format and stored in a separate file for each channel. The filenames will be generated automatically. Corresponding to chapter 5.3.3, the filename consists of the recorder's serial number, channel no, sequence no and the date and time of recording-start. In addition, each file will be marked by the extension '.wav'.

Options:

`--dest` *<path/destination directory/>*

This option defines the directory in which the converted data files will be stored.

`--lower`

Please use this option, if you would like to preserve the 16 least significant bits of the 24 bit data. Without using this option, wavewrite neglects the 8 least significant bits (as default).

1.1.15 seg-ywrite

This program module converts the internal send2x format into SEG-Y format. The program requires a shot file for cutting the long raw data stream into single traces. Please find a description of the shot file structure at the end of this chapter.

If no shot file is available, a single and long SEG-Y trace can be generated using the option --reftek.

As well as the other writing programs, seg-ywrite reads the data via standard input and writes for each activated channel and for each recording period (sequence) a file into a destination directory. The filenames will be generated automatically and marked by the extension '.segy'. Each filename consists of the recorder's serial number, channel no, sequence no and the date and time of recording-start, using the nomenclature as described in chapter 5.3.3.

Options:

`--dest` *<path/directory/>*

Please enter the complete path to and the name of the destination directory where the resulting (converted) files should be stored.

`--shot` *<path/directory/filename>*

Please enter the directory and filename where the shot file is located.

`--tracelen` *<number of samples>*

The option --tracelen allows to cut all traces to a common length.

Example: --tracelen 2000

will write traces consisting of 2000 samples.

The resulting trace length in seconds can be calculated by:

trace length in ms = (number of samples) * (sample period in ms)

e.g.: (2000)*(4ms)=8000 ms trace length

`--fill_null`

If the entered trace length in option --tracelen is longer than the time period between two shots, the option --fill_null can be used to fill the traces with zeros (0) up to the defined trace length.

Following the SEG-Y standard, the trace length can not exceed 32767 samples.

`--reftek`

This option stores the complete data in one single trace. As this file can be longer than 32767 samples, it is not written in standard SEG-Y. Instead of this, the file is compatible to the file format provided by the program REF2SEGY of REFTEK. Especially the trace headers are modified. The option --reftek can not be used in combination with --shot. Please note, that only the start-time of the first sample of the one single trace will be corrected by the automatic skew correction.

`--msb`

converts the data in 2byte, two's complement integer format, whereas only the most significant 16 bits will be copied.

`--lsb`

converts the data in 2byte, two's complement integer format, whereas only the least significant 16 bits will be copied.

`--trace_cut` *<value in seconds>*

If no shot file is available and the raw data should be cut into traces to display the data as 'seismic section', then the option --trace_cut can be used. Thereby the long raw data stream will be cut into traces of equal length, given by the entered number of seconds. The value has to be in full seconds (integer value). The option --trace_cut can not be used in combination with --reftek or --shot.

`--skew` *<+/- skew in µs>*

If you enter --skew 0, the program will apply no skew correction and ignores the skew-value written in the raw data file.

If you don't use this option, the skew value written in the raw data will automatically be used to correct the start time of each trace.

Example: --skew -1000

Instead of processing the skew value supplied with the raw data, the skew correction will now be performed using a skew value of -1 ms. Please note, that with the –reftek option, the skew is treated the same way as when using traces, no matter whether the skew is entered manually oder taken

from the raw data automatically: The proportional skew per second is calculated and used to determine the start sample for the output file.

--gcx *<integer value>*

Please enter the x coordinate for the geophone group (here for the OBS location), if available.

--gcy *<integer value>*

Please enter the y coordinate for the geophone group (here for the OBS location). if available

Following the SEG-Y standard, the x and y coordinates have to be in seconds of arc, if spherical coordinates are used. Then, the x value should represent longitude and the y-value latitude. A positive value designates the number of seconds east of Greenwich Meridian or north of the equator and a negative value designates the number of seconds south or west.

--data_le

This option controls the byte order in the data area of the resulting SEG-Y file:

Without using the option, seg-ywrite stores the trace header and the data in "Big Endian" format (recommended for PC's).

By using --data_le, the data will be written in "Little Endian" format.

--le

Both, data and header will be written in "Little Endian" format.

--out_int

If this option will be used, then the data in the SEGY- file will be written in 4-byte, two's complement integer format. Without using the option, the data will be written in "IEEE floating point" format (as default).

--log *<filename>*

This option generates a log file of the current seg-ywrite run in the current directory. Without using this option, the log file will be stored in a file with the name "seg-ywrite.log".

1.1.16 gsewrite

The gsewrite module converts data from the send2x format to GSE (Group of Seismic Experts) standard file format. It reads data from standard input and writes the output into files. For each active channel in the raw data, a separate GSE file will be written, using the standard nomenclature for file names as described in chapter 5.3.3. Optionally, a customized file name can be created using the - - nameform option. Target files will be written to the current directory unless a target directory is spezified using the - - dest option.

Usage:

```
gsewrite [options] <rawdata.s2x
```

Options:

-- dest PATH or -d PATH	Sets output for target files to given path/directory
-- nameform FORM or –n FORM	Sets file name format according to format string FORM.
	FORM is a string literal which may not contain spaces or whitespace. It can take up any ASCII characters and the following placeholder symbols:
	%n device serial number
	%c channel number
	%q record sequence number
	%v event number
	%e experiment number
	%E experiment name
	%t station number
	%T station name
	%Y year of the first sample
	%J julian day of the first sample
	%D day of month of the first sample
	%M month of the first sample
	%h hour of the first sample
	%m minute of the first sample
	%s seconds of the first sample
	%g gain value
	%r sample rate of recording
	%p sample period (in micro seconds) of recording
	The default setting to form filenames according to chapter 5.3.3. thus is: %n.%c.%q.%v.%Y%J.%h.%m.%s.gse
--split SEC or –t SEC	split the output files into segments of <SEC> seconds worth of samples each

--slice SIZE or -c SIZE	slice the output files into segments of <SIZE> megabytes size each
--verbose or -v	verbose mode with additional information for debugging
--version or -V	print version information and exit
--silent or -s	suppresses all regular message output
--help or -h	display this help and exit

1.1.17 paswrite

The paswrite module converts data from the send2x format to PASSCAL data format. It reads data from standard input and writes the output into files. For each active channel in the raw data, a separate PASSCAL file will be written, using the standard nomenclature for file names as described in chapter 5.3.3. Optionally, a customized file name can be created using the - - nameform option. Target files will be written to the current directory unless a target directory is spezified using the - - dest option. Finally, a multiplexed target file, containing interleaved data from all channels, can be written using the –multiplexed option. By default, the 16 most significant bits from each sample will be written to the target file, unless the --lsb option is used to switch to the 16 least significant bits from each sample.

Usage:

paswrite [options]<*infile*

or

paswrite [-h] [-V]

Options:

-- dest PATH or -d PATH	Sets output for target files to given path/directory
-- nameform FORM or –n FORM	Sets file name format according to format string FORM.

FORM is a string literal which may not contain spaces or whitespace. It can take up any ASCII characters and the following placeholder symbols:

%n	device serial number
%c	channel number
%q	record sequence number
%v	event number
%e	experiment number
%E	experiment name
%t	station number
%T	station name
%Y	year of the first sample
%J	julian day of the first sample
%D	day of month of the first sample
%M	month of the first sample
%h	hour of the first sample
%m	minute of the first sample
%s	seconds of the first sample

	%g gain value %r sample rate of recording %p sample period (in micro seconds) of recording The default setting to form filenames according to chapter 5.3.3. thus is: %n.%c.%q.%v.%Y%J.%h.%m.%s.passcal
--split SEC or –t SEC	split the output files into segments of <SEC> seconds worth of samples each
--slice SIZE or -c SIZE	slice the output files into segments of <SIZE> megabytes size each
--lsb or –l	only write the 16 least significant bits of samples to target file
--msb or –m	only write the 16 most significant bits of samples to target file
--multiplexed or –x	write data from all channels into a single target file
--verbose or -v	verbose mode with additional information for debugging
--version or -V	print version information and exit
--silent or -s	suppresses all regular message output
--help or -h	display this help and exit

1.1.18 sacwrite

The sacwrite module converts data from the send2x format to SAC (Seismic Analysis Code) data format. It reads data from standard input and writes the output into files. For each active channel in the raw data, a separate SAC file will be written, using the standard nomenclature for file names as described in chapter 5.3.3. Optionally, a customized file name can be created using the - - nameform option. Target files will be written to the current directory unless a target directory is spezified using the - - dest option. Optionally, the geographic location of the station may be provided as input parameter.

Usage:

sacwrite [options]<*infile*

or

sacwrite [-h] [-V]

Options:

-- dest PATH or -d PATH	Sets output for target files to given path/directory
-- nameform FORM or –n FORM	Sets file name format according to format string FORM.

FORM is a string literal which may not contain spaces or whitespace. It can take up any ASCII characters and the following placeholder symbols:

%n	device serial number
%c	channel number
%q	record sequence number
%v	event number
%e	experiment number
%E	experiment name
%t	station number
%T	station name
%Y	year of the first sample
%J	julian day of the first sample
%D	day of month of the first sample
%M	month of the first sample
%h	hour of the first sample
%m	minute of the first sample
%s	seconds of the first sample
%g	gain value

	%r sample rate of recording %p sample period (in micro seconds) of recording The default setting to form filenames according to chapter 5.3.3. thus is: %n.%c.%q.%v.%Y%J.%h.%m.%s.sac
--split SEC or -t SEC	split the output files into segments of <SEC> seconds worth of samples each
--slice SIZE or -c SIZE	slice the output files into segments of <SIZE> megabytes size each
--latitude <deg> or -a <deg>	set latitude of station to <deg> degrees, where <deg> is a floating point number with north being a positive value
--longitude <deg> or -o <deg>	set longitude of station to <deg> degrees, where <deg> is a floating point number with east being a positive value
--verbose or -v	verbose mode with additional information for debugging
--version or -V	print version information and exit
--silent or -s	suppresses all regular message output
--help or -h	display this help and exit

1.1.19 resample

Data recorded with the GEOLON-MLS and GEOLONON-MTS or MTS-M may show so-called "time slips", due to differences between the long term stabilization of the internal clock and the sample frequency clock. Thus, within a given sample period, less or more samples than required are recorded. e.g. 99 samples instead of 100. If this misalignment of sample periods and samples actually collected is influencing the precision of your experiment, you can correct the data using RESAMPLE. RESAMPLE does not correct time slips for other GEOLON Data Loggers, but may be helpful for correction of filter influences and adjusting skew times.

Usage:

resample [-v] [-s *size*] [-c *seconds*] <*infile* >*outfile*

or

resample [-h] [-V]

Options:

-h Help. Displays a list of all available options.

-V Version. Displays the version number of resample

-v Verbose. Displays process information on stderr

-s Sets the size of the internal data buffer to *size* MB. Default value is 16MB.

-c Sets the time in *seconds*, which must pass between consecutive time slips before re-interpolation starts

Remarks:

With the GEOLON-MLS, -MTS or -MTS-M, either a –c value or a –s value, or both, must be given or the command will abort indicating an error. With the GEOLON MES and MCS, normally no parameters should be given, since by design they do not show time slips and you are likely to run into buffer memory problems when setting parameters.

RESAMPLE can react to three special situations:

1. Data gaps. In this case, RESAMPLE will handle the data before and after the gap as separate streams of data.
2. If the time between two time slips is larger than the time covered by the data in the internal buffer of RESAMPLE, an error message will be displayed. In this case, the –s parameter can

be used to increase the buffer size accordingly. Please note that there is one buffer for each channel of data, so the value for this parameter is limited by the total amount of real and virtual memory of the computer, divided by the number of channels recorded. It also means, that for large files with great distances between timeslips, the memory requirement can exceed the limits of any state of the art PC. In this case we suggest splitting the input file and trying to correct smaller snippets, which can later be reassembled by concatenation.

3. Frequent discrepancies. If time slips occur in rapid succession, typically with alternating signs (+1 sample, -1 sample, etc.), the number of samples between slips is insufficient for proper interpolation. In this case, the –s parameter be set to a threshold in seconds, that has to pass without a new time slip, before a new interpolation is done.

1.1.20 pressure2ascii

This programs converts sequences of pressure data that have been retrieved from a MTS-M tsunameter seismocorder in an ocean bottom unit (OBU) by the buoy computer´s OBUD program.

It writes the data as ascii-text into standard output or into a file with a filename of your choice.

The input file has to be generated by the OBUD software. These data files can be read by pressure2ascii using the --file option.

EXAMPLE:

pressure2ascii --file 765432_10.obud.2 >/data/outputtextfile

-whereas '765432_IO.obud.2' is a file stored by the OBUD software on the buoy computer after retrieving pressure data from the OBU.

- whereas ' /data/outputtextfile ' is a path and filename of your choice. If you leave this string in your command, the ASCII data will be piped into standard output.

The resulting ASCII file looks like:

```
date & time of first sample;06.02.2006 07:28:16
sample time;pressure (mBar)
0;1088,6
15;1088,7
30;1088,7
..;......
```

The start time of the measurement is noted in the first line of the file. The second line explains the sorting of the following data. The first value in the data lines stands for the seconds gone since the measurement was started.

1.2 Combining the programs

The concept to use standard input and output allows to hand over the data easily from program to program and to combine the program modules in 'pipes'.

Example 1:

$ mesread --begin 708669934 --end 708685565 < /dev/sdc | seg-ywrite --dest ~/segy_data/

--shot ~/shots/shotfile.txt --tracelen 2000

In this example, mesread cut out data from the raw data file on GEOLON's harddisk between the GPS time 708669934 seconds and 708685565 seconds and passes them over to seg-ywrite via standard output. Seg-ywrite cuts the data stream into traces of 2000 samples regarding the shot times in the shot-time file shotfile.txt. In the example this file is located in the subdirectory 'shots/' of your home directory. For each activated channel during recording, a file with the extension '.segy' will be stored in the directory ~/segy_data/. The complete sequence of commands could also be arranged in a script.

Example 2:

$ mesread --begin 708669934 --end 708685565 </dev/*devicefile* > ~/temp.s2x
$ seg-ywrite < ~/temp.s2x --dest ~/segy_data/ --shot ~/shots/shotfile.txt --tracelen 2000
$ wavewrite < ~/temp.s2x --dest ~/wave_data/

Here, mesread cut out data from the raw data file on GEOLON's harddisk (via Firewire connection, established through the *devicefile*) between the GPS time 708669934 seconds and 708685565 seconds and writes standard output into the file 'temp.s2x' in your home directory. In the second line you find the command to start seg-ywrite, which cuts the data stream into traces of 2000 samples regarding the shot times in the shot-time file shotfile.txt. The third line shows how the same file 'temp.s2x' can be used for data conversion into audio wave format. If you would use pipes to get the same result, mesread would be started twice: The first time in the pipe combining mesread with seg-ywrite and the second time combining mesread with wavewrite.

1.2.1 Structure of the shot file

Normally, the program uses a shot file for converting the raw data file in a file of single traces. The shot file has to contain the name of shooting-line (or profile), the shot point number, and shot time at minimum. Also the entry of the geographical coordinates of the shot points are possible. Together with the coordinates for the receiver, provided by the options --gcx and --gcy of seg-ywrite, the complete geographical positioning information of shot and receiver can be stored in the resulting SEG-Y file.

Shotfiles generally consist of a header line containing field descriptions and parameter options, followed by lines of data and terminated by a blank line containing a single carriage return.

Seg-ywrite can read following shot file formats:

1: GPS time in seconds.microseconds

LINENAME	SHOTPOINT	GPS-TIME:SEC
abcd-1234	1234	1234567.123456
abcd-1237	1237	1234597.123456
abcd-1314	1314	1234607.123456

2: GPS time in seconds.microseconds + coordinates of shotpoint

LINENAME	SHOTPOINT	GPS-TIME:SEC	X-COORDINATE	Y-COORDINATE
abcd-1234	1234	1234567.123456	374136	6456332
abcd-1237	1237	1234597.123456	374136	6456332
abcd-1314	1314	1234607.123456	374136	6456332

3: GPS time in year.month.day_hour:minute:second.microsecond

e.g.: 2003.05.07_06:14:49.123456

LINENAME	SHOTPOINT	GPS-TIME:DATE
abcd-1234	1234	2003.05.07_06:14:49.123456
abcd-1237	1237	2003.05.07_06:14:50.123456
abcd-1314	1314	2003.05.07_06:14:51.123456

4: GPS time in year.month.day_hour:minute:second.microsecond + coordinates of shotpoint

LINENAME	SHOTPOINT	GPS-TIME:DATE	X-COORDINATE	Y-COORDINATE
abcd-1234	1234	2003.05.07_06:14:49.123456	374136	6456332
abcd-1237	1237	2003.05.07_06:14:50.123456	374136	6456332
abcd-1314	1314	2003.05.07_06:14:51.123456	374136	6456332

The header line contains the following field discriptors:

`LINENAME:` The corresponding fields contain descriptive string up to 16 characters long.

`SHOTPOINT:` The corresponding fields contain an ASCII coded number up to 2147482648.

GPS-TIME: The corresponding fields contain ASCII coded time information. The colon-separated parameter options may be either **DATE** for a human-readable timeformat or **SEC** for time values in GPS seconds since start of service on January 5th, 1980.

The data format for the **SEC** option is SECONDS.MICROSECONDS.

The data format for the **DATE** option is YYYY.MM.DD_HH:MM:SS.ssssss, where

YYYY	holds the year as a four digit number
MM	holds the month as a two digit number with leading zero
DD	holds the day as two digit number with leading zero
HH	holds the hour as two digit number with leading zero
MM	holds the minute as two digit number with leading zero
SS	holds the seconds as two digit number with leading zero
ssssss	holds the microseconds as six digit number

Please note, that all GPS time formats have a common requirement: The microseconds must consist of figures with 6 numbers after the decimal point.

The program expects pure GPS-time, which is counted in seconds after 6th January 1980, 00:00 o'clock (start of GPS by USA). Often UTC is used instead of the GPS-time. UTC time is currently (in 2006) 14 seconds behind GPS Time, that means:

GPS-time = UTC + 14 seconds

When you use UTC-time for the synchronisation of the recorder and the GPS of the gun-boat delivers GPS-time, then 14 sec have to be added to your recorder-time (see option -timeshift) to get a comparable time basis.

X-COORDINATE: The corresponding fields optionally contain the x-coordinate of the shot position.

Y-COORDINATE: The corresponding fields optionally contain the x-coordinate of the shot position.

Following the SEG-Y standard, the x and y coordinates have to be in seconds of arc, if spherical coordinates are used. Then, the x value should represent longitude and the y-value latitude. A positive value designates the number of seconds east of Greenwich Meridian or north of the equator and a negative value designates the number of seconds south or west.

Seg-ywrite automatically recognises if your shot file contains only 3 columns instead of 5 (if no coordinates are available). The columns have to be separated by empty spaces or tabs.

Seg-ywrite writes four SEG-Y files, each containing the data of one channel.

If you have no shot file, please use the program option wavewrite. The resulting '.wav' file can be displayed using a sound-display program as for example COOLEDIT.

1.3 Files

The nomenclature of the automatically generated files will be described hereafter.

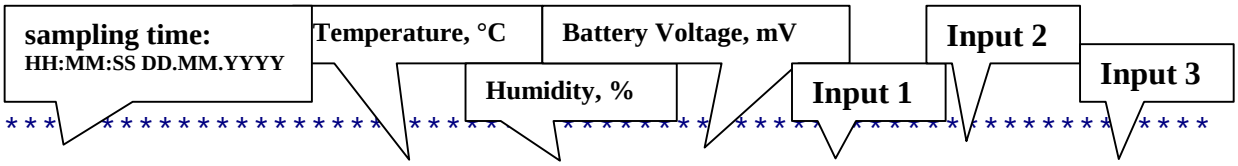
1.3.1 Log - File of Geolon-MES and Geolon MCS

All control, status, and identification information of the current experiment are stored on GEOLON's harddisk. Furthermore, the log data includes the parameter settings such as the sample rate and the amplifier gain for each channel. Normally the program meslog displays these data on the screen. Through the assignment of the standard output to the harddisk on the connected external PC, these data can alternatively be stored in a text-file. The user has to define the filename and the destination directory in the kind of *log file_ name*.txt.

1.3.2 Engineering data

The internal temperature and humidity as well as the battery voltage will be stored together with the recorded seismic data on the internal disk. The programs mesread and mcsread provide the extraction of the engineering data out of the recorded data stream. The file will automatically be written in the current working directory. The corresponding file name is described in chapter 1.3.3.

The engineering data file looks like:




```
***************************************************************************
time                    temp  hydr   ubat   ain1   ain2   ain3
06:14:49 07.05.2003      27    49   13623
06:15:49 07.05.2003      27    49   13623
 ..    ..    ....       ...   ...   ...     ...    ...     ...
***************************************************************************
```

Input 1-3: free channels for user defined input. For the GEOLON-MTS and MTS-M variants, the engineering data files contain the pressure values recorded for the corresponding pressure sensor.

1.3.3 Data Files

The programs binwrite, wavewrite and seg-ywrite generate automatically files with names of following nomenclature:

020806.01.00.01.2002.168.07.09.57.extension

serial number of GEOLON	channel no	sequence no	event no	start time of recording at activation of REC command	
					.bin for binary files **.wav** for audio wave files **.segy** for files in SEG-Y format **.eng** for engineering data of MES and MCS recorder **.pressure** for pressure data of MTS and MTS-M recorder

The start time of recording consists of:

the year 2002

number of days since the beginning of the year	168
the hour	07
minutes	09
seconds	57

User Manual

Installation and Operation

Table of contents

1 General Information 3
1.1 Operating Systems and related Issues 3
2 Installation 3
2.1 Obtaining the Software 3
2.2 Installing on a Linux System 3
2.3 Installing on a Windows System (2000 or XP) 5
3 Using Sendcom 7
3.1 The Menu Bar 7
3.2 The Command Window 9
3.3 The Show Window 9
3.4 Function Keys 11
3.5 Exiting the SHOW window and SENDCOM 11
4 Troubleshooting 11
4.1 Error: No Class Definition for SendCom 11
4.2 Error: Illegal use of nonvirtual function call 12
4.3 Error: Class Cast Exception thrown 12
4.4 Error: No Class Definition for EventListener 13
4.5 SendCom will not start for a normal user 13

SendCom User Manual

Document No. SendCom2.20a.mnl.doc,
May 2005

Manufactured by:

Signal Elektronik GmbH

Rostocker Str. 20
D-20099 Hamburg

Tel: +49 40 375008 03
Fax:+49 40 375008 93

URL: http://www.send.de
e-mail: office@send.de

1 General Information

This program is used to communicate with GEOLON-Dataloggers via the RS232 interface for interactive configuration and for displaying recorded data on the PC screen.

1.1 Operating Systems and related Issues

SendCom is a Java-application that works on both Linux and Windows operating systems. The only prerequisite is that the Java Runtime Engine (JRE) is installed on the PC. We recommend using our customized KNOPPIX-CD, which allows running Linux and SendCom on most PC-configurations without the need to install Linux or SendCom on that PC. However, if you want a permanent installation of SendCom, or want to use MS Windows, just follow the steps below.

2 Installation

2.1 Obtaining the Software

SendCom can either be found on the Utility-CD supplied with our data loggers, or it can be downloaded from our website. Please copy the files to a folder on your local hardrive.

2.2 Installing on a Linux System

2.2.1 Unpacking the files

Unzip the file sendcom_ver_x.xx.zip (if you already have installed java)
or sendcom_ver_x.xx_java.zip (if you do not have installed java already).

This will produce a temporary hierarchy with a top level directory sendcom.
(E.g.: /tmp/sendcom)

2.2.2 Java Installation

In order to use the SendCom software, J2RE (Java run time engine only) or J2SDK (complete system development kit) must be installed on your computer. Please follow the installations instructions in either 1.2.2.1. OR 1.2.2.2. according to your needs.

2.2.2.1 Installing for JRE, if not already installed

Please check whether a JRE is installed on your computer. If not, you can get the JRE for your version of Linux from Sun Microsystems at:

http://java.sun.com/products/plugin/downloads/index.html

To install and configure Java in Linux, please follow the instructions provided at:

http://java.sun.com/j2se/1.4.2/jre/install.html

For the purposes of this paper, it will be assumed that you have installed a JRE, which by default is located in:

/usr/java/j2re1.4.2_04/

And java is then located in:

/usr/java/j2re1.4.2_04/bin/

Check the path to your java-interpreter with "which java". If your setup is different, please adjust accordingly. Please note, that newer versions of Java might have become available meanwhile, which will also work with SendCom.

2.2.2.2 Installing for Java SDK, if not already installed

Please check, if a SDK is installed on your computer. If not, you can get the SDK for your version of linux from Sun Microsystems at:

http://java.sun.com/j2se/1.4.2/download.html

If you cannot get access to the sun site, you will find the corresponding file in the folder:

<sendcom>/drivers_4_linux/java/j2sdk-1_4_2_04-linux-i586-rpm.bin

To install and configure Java in Linux, please follow the instructions provided at:

http://java.sun.com/j2se/1.4.2/install.html

For the purposes of this paper, it will be assumed that you have installed the SDK, which by default is located in:

/usr/java/j2sdk1.4.2_04/

And java then is located in:

/usr/java/j2sdk1.4.2_04/bin/

Check the path to your java-interpreter with "which java". If your setup is different, please adjust accordingly.

2.2.3 Installing the drivers

2.2.3.1 If you use JRE

Copy librxtxSerial.so to your <JRE>/lib/ext directory.

Example:
cp /tmp/sendcom/drivers_4_linux/librxtxSerial.so /usr/java/j2re1.4.2_04/lib/ext

Copy RXTXcomm.jar to your <JRE>/lib/ext directory.

Example:
cp /tmp/sendcom/drivers_4_linux/RXTXcom.jar /usr/java/j2re1.4.2_04/jre/lib/ext

2.2.3.2 If you use J2SDK:

Copy librxtxSerial.so to your <SDK>/jre/lib/ext directory.

Example:
cp /tmp/sendcom/drivers_4_linux/librxtxSerial.so /usr/java/j2sdk1.4.2_04/jre/lib/ext
Copy RXTXcomm.jar to your <SDK>/jre/lib/ext directory.

Example:
cp /tmp/sendcom/drivers_4_linux/RXTXcom.jar /usr/java/j2sdk1.4.2_04/jre/lib/ext

2.2.3.3 Clean up installation files

You may now remove the installation files from the temporary folder

Example:
rm -R /tmp/sendcom

2.3 Installing on a Windows System (2000 or XP)

2.3.1 Unpacking the files

Unzip the file *sendcom_ver_x.xx.zip* (if you already have installed java) or *sendcom_ver_x.xx_java.zip* (if you do not have installed java already). This will produce a temporary hierarchy with a top level directory ***sendcom.*** (E.g.: ***C:\sendcom***)

2.3.2 Java Installation

In order to use the SendCom software, a J2RE (Java runtime engine only) must be installed on your computer. Please follow the instructions in 1.3.3 if you don´t have a JRE installed.

2.3.2.1 Installing JRE, if not installed already

Please check, if a JRE is installed on your computer. In the case it is not, then you can obtain a JRE for your version of Windows from **Sun Microsystems** at

http://java.sun.com/products/plugin/downloads/index.html

If you cannot get access to the sun site, you will find the corresponding file in the folder:
*<sendcom>/*drivers_4_windows/java/ *j2re-1_4_2_03-windows-i586-p.exe*

(E.g.: C:\sendcom\drivers_4_windows\java\j2re-1_4_2_03-windows-i586-p.exe) Please note, that there might be newer versions of the Java Runtime Engine available from Sun Microsystems. These will also work with SendCom.

2.3.3 Copying the drivers

Copy rxtxSerial.dll to your <JRE>\bin directory.

Example:
(E.g.: if you extracted the files into the folder C:\sendcom and java has been installed to the default location)

copy c:\sendcom\drivers_4_windows\rxtxSerial.dll c:\Programme\Java\jre1.4.2_03\bin
(for German Windows)

copy c:\sendcom\drivers_4_windows\rxtxSerial.dll c:\Program Files\Java\jre1.4.2_03\bin
(for English Windows)

2.3.4 Copy RXTXcomm.jar to your <JRE>\lib\ext directory

Example:
(E.g.: if you extracted the files into the folder C:\sendcom and java has been installed to the default location)

copy c:\sendcom\drivers_4_windows\RXTXcomm.jar c:\Programme\Java\jre1.4.2_03\lib\ext
(for German Windows)

copy c:\sendcom\drivers_4_windows\RXTXcomm.jar c:\Program Files\Java\jre1.4.2_03\lib\ext
(for English Windows)

2.3.5 Copy sendcom.jar to your <sendcom> directory

If you already have a previously installed version of sendcom, just overwrite the **SendCom.jar** file in your **<sendcom>** folder. Otherwise, please create a new folder, e.g. c:\Programme\sendcom

Example:
(E.g.: if you extracted the files into the folder C:\sendcom)

copy c:\sendcom\SendCom.jar c:\Programme\sendcom
(for German Windows)

copy c:\sendcom\SendCom.jar c:\Program Files\sendcom
(for English Windows)

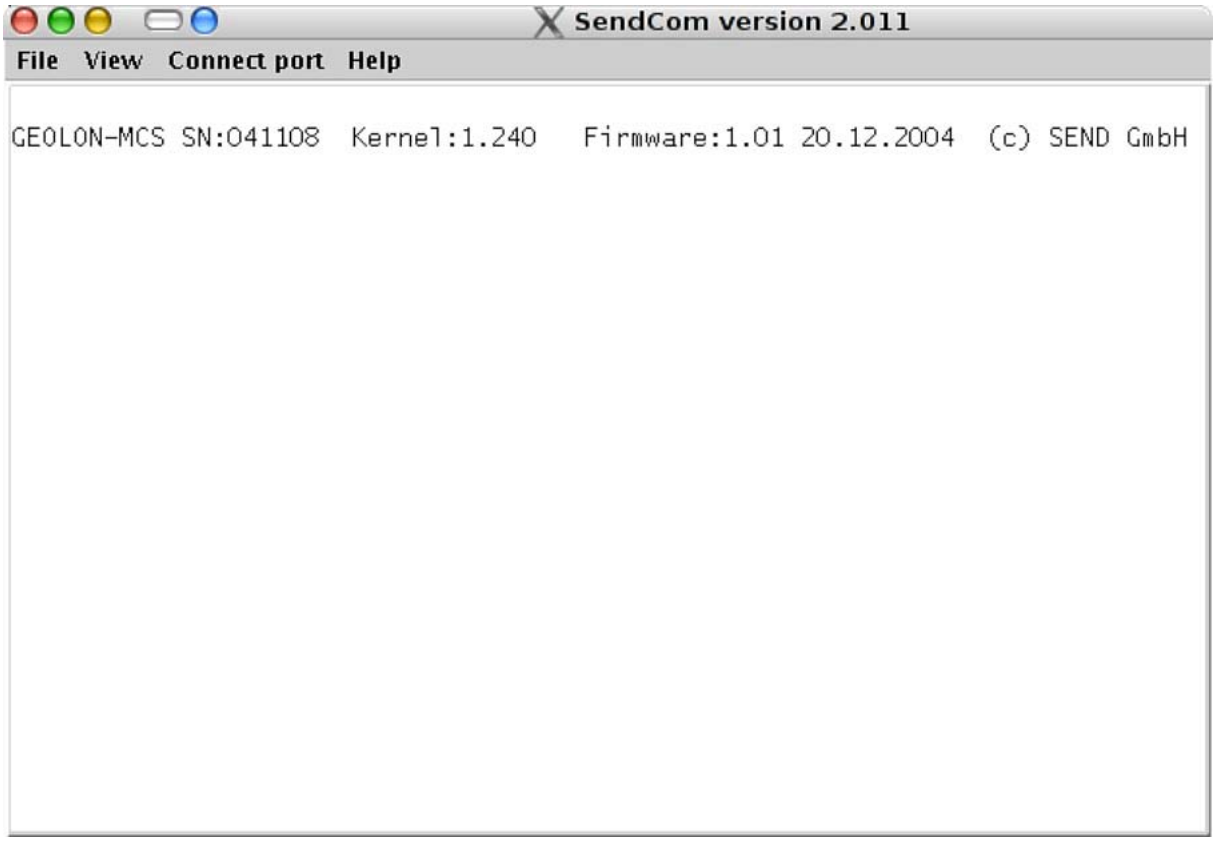
2.3.6 Delete temporary directory *sendcom*

2.3.7 Optionally, you can add SendCom.jar to your classpath:

set CLASSPATH=c:\Programme\sendcom\sendcom.jar;%classpath%

3 Using Sendcom

SendCom starts up with a window like this:



The window frame and menu bar on your PC may look different from this, depending on the settings of the desktop you are using, e.g. Windows or Linux with Gnome, KDE, etc.

3.1 The Menu Bar

3.1.1 The File menu

The file menu allows uploading of system files to the GEOLON loggers, saving the logfile to the local harddisk and leaving the programm.

3.1.1.1 Load

This function brings up a file selection window, where you can navigate to the location of a firmware update, user programm or other system files for your GEOLON logger. The file will automatically be transferred from the local drive to the logger´s storage device. You can also use the drag & drop feature of your operating system by simply dragging the desired file into the SendCom window.

3.1.1.2 Save logfile

This functions brings up another file selection window, where you can select a destination folder and filename for the SendCom logfile of your session on your local drives. The Logfile will contain all commands and responses which have been issued so far.

3.1.1.3 Exit

This function terminates SendCom and clears the DTR signal for the GEOLON logger. The same is achieved by clicking on the “close” box in the upper right corner of the window.

3.1.2 The View menu

The view menu allows the selection of additional display areas and windows, which are used less frequently in typical mission preparation sessions.

3.1.2.1 Show window

This is the most important additional window, since it displays the aquisition data in realtime from either of the recorders channels. The functions of the show window are described in an own section of this manual.

3.1.2.2 System messages

When this option is selected, a new window area is displayed at the bottom of the main window. Here you can find the current communication parameters and other system information.

3.1.2.3 Status

When this option is selected, a new window area is displayed at the bottom of the main window. If the system messages also are activated, the status window area will be placed underneath the system message area. The information displayed in this area reflects internal parameters of the GEOLON logger and is usually only of interest for factory maintenance and debugging purposes.

3.1.3 The Connect Port menu

3.1.3.1 COM1..[COMn] or /dev/ttyS0..[/dev/ttySn]

By default, SENDCOM uses COM1 (on Windows) or /dev/ttyS0 (on Linux) as its communication port. If you want to connect to a different port, you can select the appropriate port here. Sendcom lists only those ports which are not currently occupied by other applications, so your desired port may not show up at all. You may want to close interferring applications to free the port and then use the “Rescan ports” to refresh the port list.

3.1.3.2 Rescan Ports

If you are missing COM-ports in the list of available ports, they might be occupied by other applications. SendCom can only detect and list free ports, which are not occupied by other programs. If you decide to free an occupied port, you must use the Rescan Ports function to refresh the list of available ports to make the freed port available for SendCom.

3.1.4 The Help menu

The Help menu gives you information how to contact SEND Signal Elektronik GmbH for support and other issues, as well as information that may be useful for us to solve these issues.

3.1.4.1 About SendCom

This opens a new window, which displays our contact information like phone numbers, email addresses on the “About” page. The “Details” page gives information about the version number of SendCom and the Java engine used for compilation.

3.2 The Command Window

The command window is the main part of SendCom. All commands send to the GEOLON device and its responses will be displayed here and written to the logfile when logging is activated. Please consult your GEOLON manual for details about the possible configuration procedures and commands. Commands are generally sent by typing them on the computer keyboard, like it is done with a regular terminal program. Keystrokes will not appear on the screen unless they have been properly received and echoed by the connected device. Some functions and commands have been assigned to certain keys to make things a bit easier. Please refer to the “function keys” section for details.

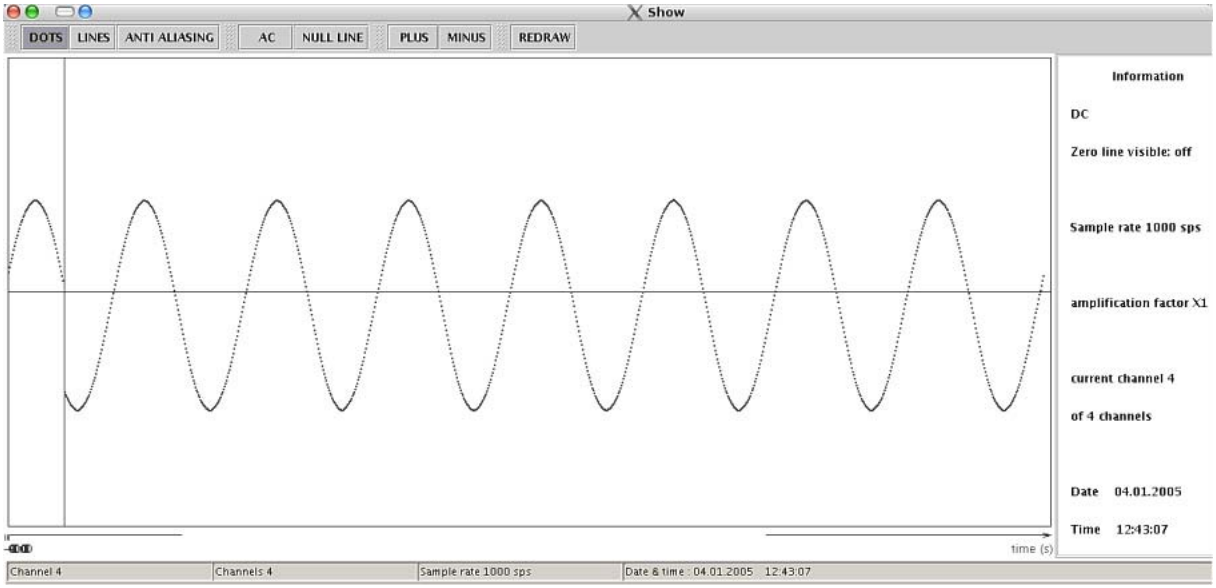
3.3 The Show Window

The show window displays the data of a single channel from a GEOLON recorder in realtime. Though the window can be opened anytime from the view menu, it can only display data after the appropriate command has been issued to the logging device. Consequently, just issuing this command will also open the show window automatically.

The **SHOW** command can be used to display the current signals of a selected channel. SHOW may be used to check the proper operation of the sensor electronics prior to starting an experiment.

Usage:
***<string>* SHOW**
displays data of channel <string>, so <string> selects the channel to be displayed.
^C (ctrl-C) will terminate the command.

Example:
4 SHOW <*enter*>
This opens the show window (if not open already) and displays the data from channel number four, as shown below.



On the top menu bar of the show window, activated options are indicated by highlighting the respective buttons. In the example displayed above, the option DOTS is activated. DOTS sets the display to dots only. Other options are:

LINES:	data points will be connected with lines
ANTIALIASING:	evens out steps which are generated due to insufficient screen resolution.
AC:	in AC mode, any DC offset will be removed
NULL LINE:	although the null line will be displayed as grey line already, this option activates a modus in which each second data dot will be set to null and marks the null line
PLUS:	Amplifies current signal by a factor of 2
MINUS:	Attenuates current signal by factor of 2
REDRAW:	Redraws the screen, use in case of display artifacts

Depending on the type of data logger connected, additional information will be displayed in the information window on the right.

Due to the limited transfer rate of the RS232 interface, a delay in displaying the signals can occur at sampling rates of 500 Hz and higher, also depending on PC properties. Not all data can be displayed for these frequencies, resulting in a small time gap between two redraws of the screen.

3.4 Function Keys

In order to make frequently used commands better accessible, some of them have been assigned to the function keys F1 to F12 on your PC keyboard. Again, some of these keys are only needed for debugging and support issues by the SEND staff, the others speed up typical mission preparation steps.

F1 Displays the list of function key assigned commands
F2 ?forth - For service purposes only
F3 1 load - For service purposes only
F4 application - For service purposes only
F5 hex - sets number display to hex
F6 decimal - sets number display to decimal
F7 boot - reboots data logger
F8 settings - displays the current settings of the data logger
F9 1 show - displays the current data from channel one in show window
F10 2 show - displays the current data from channel two in show window
F11 3 show - displays the current data from channel three in show window
F12 4 show - displays the current data from channel four in show window

3.5 Exiting the SHOW window and SENDCOM

Please click on the X button on the right top corner of the window for closing the SHOW window. You can also press CTRL-C in the Command Window to close the SHOW window.

To exit SENDCOM you can either click on the exit option in the file submenu or click on the X button on the right top corner of the SENDCOM window.

4 Troubleshooting

4.1 Error: No Class Definition for SendCom

If you get the following error:

java -jar SendCom.jar
Exception in thread "main" ***java.lang.NoClassDefFoundError:*** *SendCom*

then you have not added the SDK to the classpath. Just add the SDK to the classpath, or type on your command line:

<PATH_TO_J2SDK_BIN>/java -jar SendCom.jar

Example:
/usr/java/j2sdk1.4.2_04/bin/java -jar SendCom.jar

4.2 Error: Illegal use of nonvirtual function call

If you get the following error:

Exception in thread "main" java.lang.VerifyError: (class: gnu/io/RXTXPort$SerialOutputStream, method: write signature: ([BII)V) Illegal use of nonvirtual function call
at gnu.io.RXTXPort.<init>(RXTXPort.java)
at gnu.io.RXTXCommDriver.getCommPort(RXTXCommDriver.java)
at javax.comm.CommPortIdentifier.open(CommPortIdentifier.java:547)

or something like that, then you can work around it by adding **-noverify** to your command line.

Usage:
java -classpath <PATH_TO_SENDCOM> -noverify <PATH_TO_SENDCOM>/SendCom.jar

Example:
java -classpath /usr/sendcom -noverify /usr/sendcom/SendCom.jar

4.3 Error: Class Cast Exception thrown

If you get the following error:

java.lang.ClassCastException thrown while loading gnu.io.RXTXCommDriver

or a similiar error message, then you probably have to delete older versions of the com drivers from your computer:

Example for Linux:
rm -r /usr/java/j2sdk1.4.2_04/jre/lib/i386/libParallel.so

rm -r /usr/java/j2sdk1.4.2_04/jre/lib/i386/libSerial.so

rm -r /usr/java/j2sdk1.4.2_04/jre/lib/ext/jcl.jar

rm -r /usr/java/j2sdk1.4.2_04/jre/lib/javax.comm.properties

Example for Windows:
del <path to java>/jre/lib/i386/libParallel.so

del <path to java>/jre/lib/i386/libSerial.so

del <path to java>/jre/lib/ext/jcl.jar

del <path to java> /jre/lib/javax.comm.properties

4.4 Error: No Class Definition for EventListener

If you get this quite impressive error:

Exception in thread "main" java.lang.NoClassDefFoundError:
gnu/io/SerialPortEventListener
at java.lang.ClassLoader.defineClass0(Native Method)
at java.lang.ClassLoader.defineClass(ClassLoader.java:537)
at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:123)
at java.net.URLClassLoader.defineClass(URLClassLoader.java:251)
at java.net.URLClassLoader.access$100(URLClassLoader.java:55)
at java.net.URLClassLoader$1.run(URLClassLoader.java:194)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(URLClassLoader.java:187)
at java.lang.ClassLoader.loadClass(ClassLoader.java:289)
at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:274)
at java.lang.ClassLoader.loadClass(ClassLoader.java:235)
at java.lang.ClassLoader.loadClassInternal(ClassLoader.java:302)

or something similiar, the drivers probably have not been installed properly. Please check the *<SDK>/jre/lib/ext* directory for the presence of *librxtxSerial.so* and *RXTXcomm.jar* . If neccessary, repeat step 1.2.3 and, if required, 1.2.4.

4.5 SendCom will not start for a normal user

This applies to Linux users only:

- Change permissions for ***/dev/ttyS**** or ***/dev/tts/**** with the following commands:
  chmod g+rw /dev/ttyS or chmod g+rw /dev/tts/**
  chgrp tty /dev/ttyS or chgrp tty /dev/tts/**

- Add your user to a group like ***/dev/ttyS**** or ***/dev/tts/**** (tty) with the *kuser* utility.

- Change permissions for ***/var/lock*** with the follwoing command:
  chmod g+rw /var/lock

- Add your user to a group like */var/lock* (uucp) with the *kuser* utility.