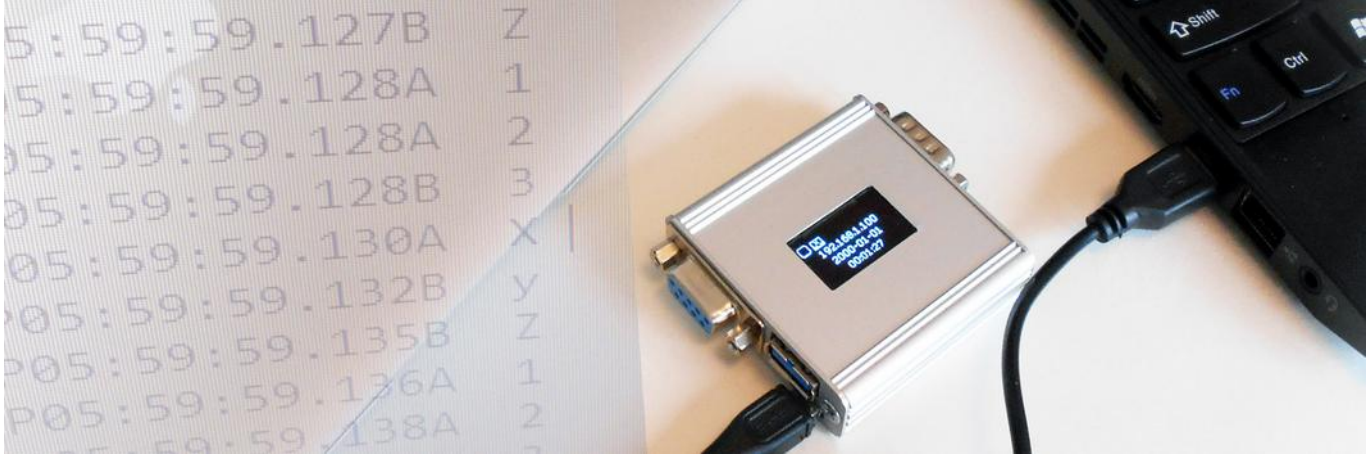
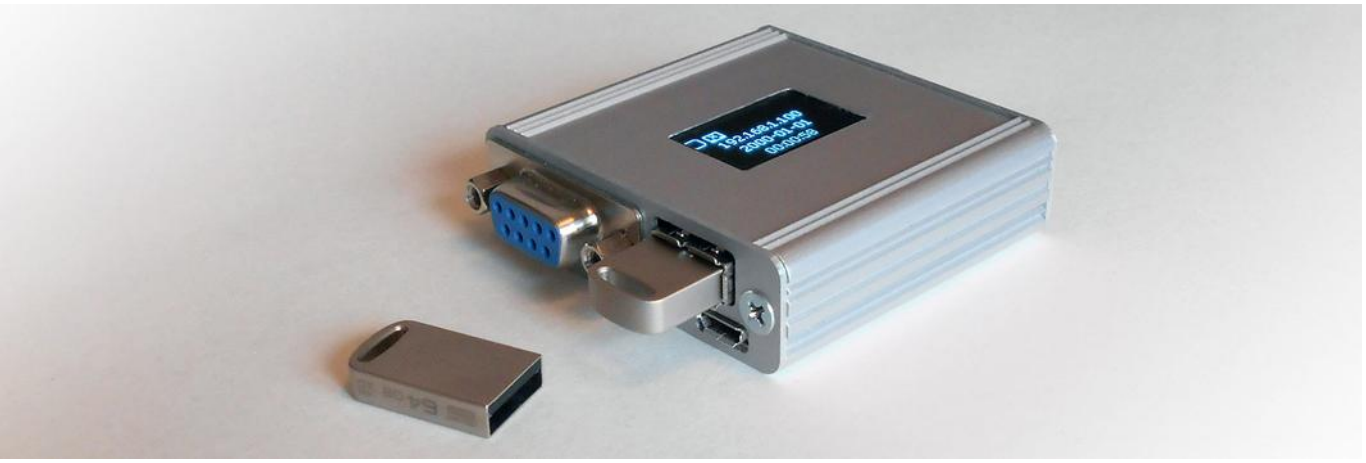


RSLogger

Users Manual

Ver. 1.30 © Electroware 2018 www.rslogger.com



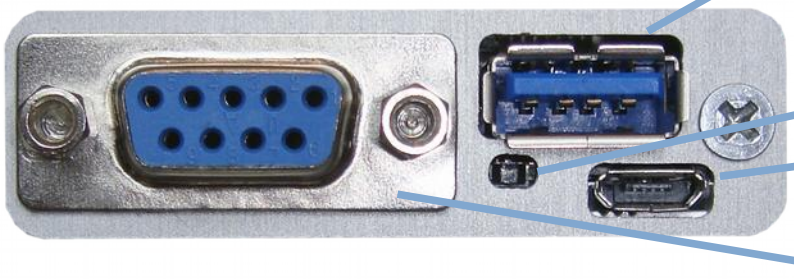
Introducing *RSLogger*: the ultimate RS232 recorder.

Main features:

- 512 MB fast internal FLASH ECC memory for data circular buffer
- asynchronous RS232, two channel (RX & TX) recording up to 230400 bps baud rate
- independent RS232 port settings (baud rate, etc) for each RX, TX channel
- simultaneous recording of RX/TX channels to one common or two separated output TXT files
- external USB Mass Storage support (FAT32) with intelligent automatic data backups
- Ethernet connectivity (RJ45) with HTTP control panel (WWW access)
- Virtual COM port on USB (FTDI chip standard, multiple OS support and SDK)
- Virtual COM Text Menu mode for settings and AT commands for data download & remote control.
- RS232 ports forwarding/splitting mode to Virtual COM (USB) or Ethernet TCP output
- Advanced Date&Time stamping (date as a file name, 1ms precision time stamps, rechargeable RTC backup battery, built-in smart recharge circuit)
- OLED 128x64 display for actual device status
- Always up-to-date with secure firmware update feature and back to the factory defaults option
- Micro USB connector for standard external 5VDC power adapters or USB port
- Wide range of standard supply voltages (5 -12VDC), Industrial and Module versions with extended supply voltage range (5-24VDC).
- ESD safe and EMI enhanced, ultra compact, durable aluminum enclosure
- programmable GPIO pins enabling additional features for industrial and module version

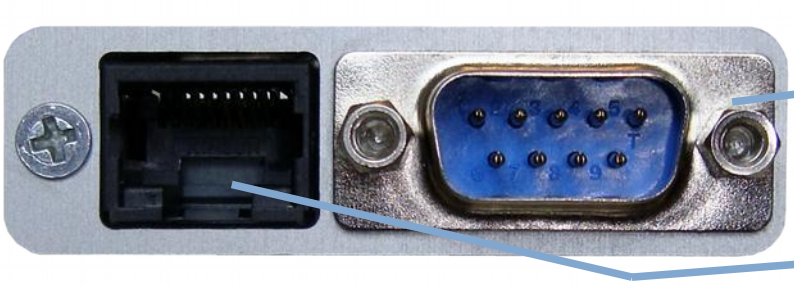
© *Electroware* 2018

RSLogger front panel



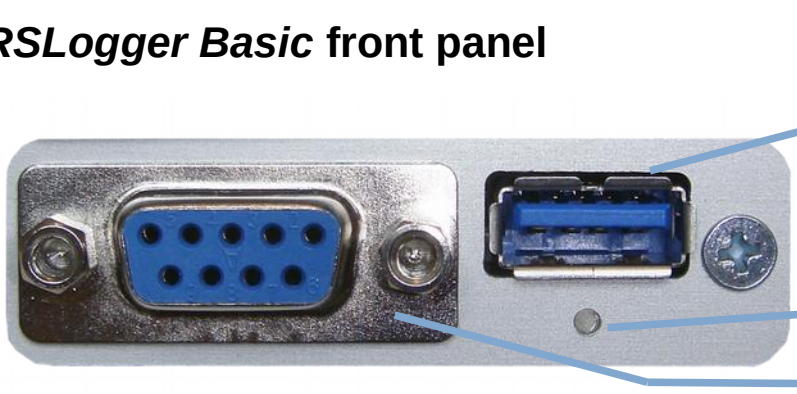
USB-A: connect your USB Mass Storage here for on-demand data download, automatic backups or left unconnected.
Push Button: hidden button allows to switch display page (short) or back to factory firmware option.
Micro USB: main power supply port (5VDC) and USB Virtual COM mode communication via Micro USB cable.
RS232 DSub: RS232 channel A (RX) input (all 9 pins are 1:1 connected with channel B port at the rear panel)

RSLogger rear panel



RS232 DSub: RS232 channel B (TX) input (all 9 pins are 1:1 connected with channel A port at the front panel)
RJ-45: connect LAN/WAN cable to enable the Ethernet functions. Find the actual connection status on OLED display.

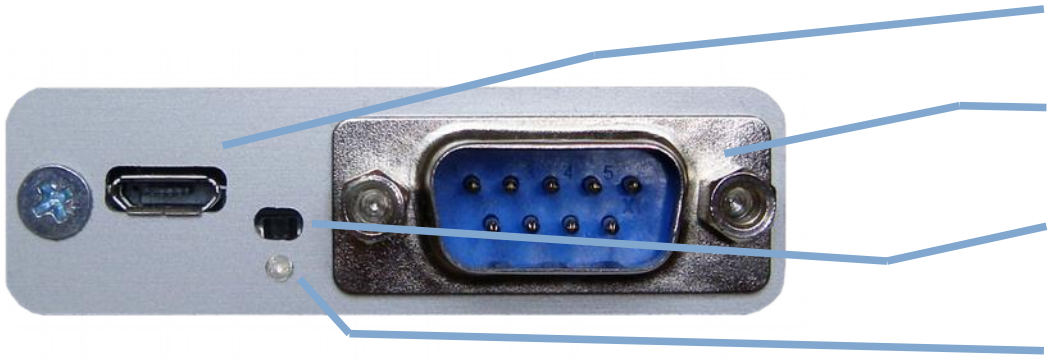
RSLogger Basic front panel



USB-A: connect your USB Mass Storage here for on-demand data download, automatic backups or left unconnected.
LED1: USB Mass Storage status indicator. Avoid disconnecting of the Mass Storage during read/write operation! LED1 on when Mass storage connected, LED1 blinks while read/write, LED1 off when no drive detected.
RS232 DSub: RS232 channel A (RX) input (all 9 pins are 1:1 connected with channel B port at the rear panel)

© *Electroware* 2018

RSLogger Basic rear panel



Micro USB
5VDC power supply and Virtual COMoperations

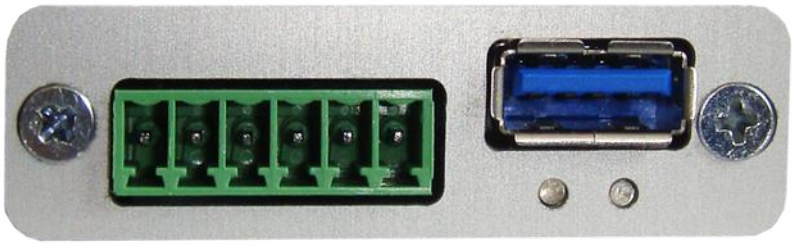
RS232 DSub 9-pin
input channel B

Push Button
device control and back to default function

LED 2
system status indicator

Micro USB: main power supply port (5VDC) and USB Virtual COM mode communication via Micro USB cable.
RS232 DSub: RS232 channel B (TX) input (all 9 pins are 1:1 connected with channel A port at the front panel)
Push Button: hidden button enables the one-click back to factory firmware (when pressed at power up for 5s).
LED 2: system status indicator for devices without OLED display. LED2 is on as power-on indicator, LED2 blinks fast while actual RS232 data recording to internal flash memory, LED2 blinks during back to default reset procedure.

RSLogger Industrial front panel



USB-A
USB Mass Storage

RS232: RX, TX, GND GND, Vcc, Aux

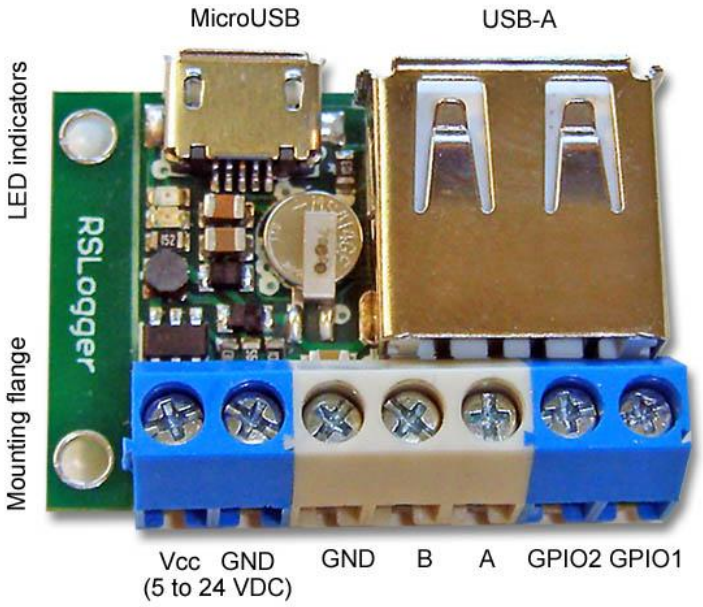
LED 1 & 2
system status indicators

Detachable, 6-ways, 3.5mm pitch standard industrial terminal connector for direct and secure wires connection:

A: RS232 link channel A (TX) input
B: RS232 link channel B (RX) input
Vcc: main power supply port (accepts 5 to 28V DC)
GND: RS232 ground and power ground (both GND pins are shorted)
AUX: auxiliary line/GPIO port for additional control (connected with push button feature)
LED 1: USB Mass Storage status indicator. Avoid disconnecting of the Mass Storage during read/write operation! LED1 on when Mass storage connected, LED1 blinks while read/write, LED1 off when no drive detected.
LED 2: system status indicator for devices without OLED display. LED2 is on as power-on indicator, LED2 blinks fast while actual RS232 data recording to internal flash memory, LED2 blinks during back to default reset procedure.
USB-A: connect your USB Mass Storage here for on-demand data download, automatic backups or left unconnected.

© *Electroware* 2018

RSLogger Module



Micro USB: for 5VDC power (from USB port or charger) and USB VirtualCOM port modes

USB-A: USB Mass Storage port

Vcc – main supply voltage input (5-24VDC accepted for Industrial version, 5VDC only for Eco version)

GND, GND – main ground and RS232 ground terminals (notice the both ground pins are internally shorted)

B - RS232 link channel B (RX) input

A - RS232 link channel A (TX) input

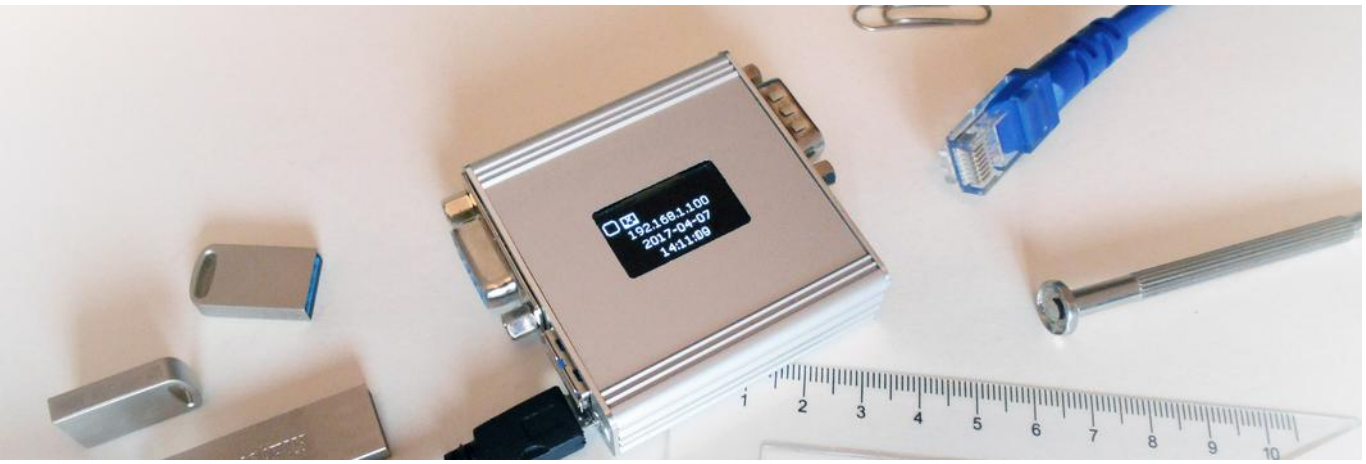
GPIO1,2 – general purpose input/output ports (to be adapted to the target implementation if more features or control are needed, please contact us in case of any requirements). The default GPIO1 and GPIO2 functionality for RSLogger module is the "back to factory default" function described in the "*Back to factory default firmware & settings*" chapter.

RED LED indicator: USB Mass Storage status indicator. Avoid disconnecting of the Mass Storage during read/write operation! LED ON: when Mass storage connected, LED blinks while read/write. LED OFF when no drive detected.

GREEN LED indicator: system status indicator for devices without OLED display. LED ON as a power-on indicator, LED blinks fast while actual RS232 data recording to internal flash memory, LED blinks slow during back to factory default reset procedure.

Mouting flange – use this flange and M2 screws to fix your RSLogger module on target position, if not needed it's possible to breake-out the flange as it's v-scored.

© *Electroware* 2018



1-2-3 Quick Start & Configuration

1. Power it on.

Connect the DC power supply to the MicroUSB port or dedicated terminal. External 5VDC adapter or USB port is fine (up to 24VDC for industrial or module version). Device is ready to be configured. For OLED featured version the display will show the welcome screen and some basic information. Models without OLED display will show actual status using system LEDs.

2. Set it up.

- By CONFIG.INI text configuration file
- Via Virtual COM Text Menu (USB)
- With Ethernet LAN connection

Please refer to **"*Configuration*"** chapter for detailed configuration information.

3. Record!

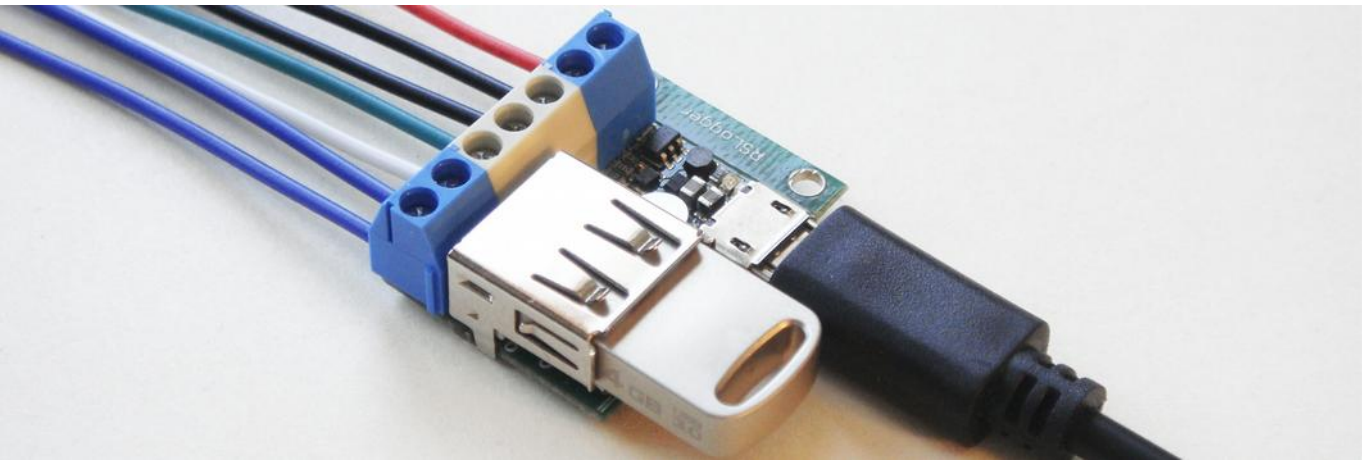
Once the device is already set up to meet your installation requirements it's ready to record RS232 data. Connect the RS232 cable (or cables) and power it on! RSLogger may act as an end-point device to record the RX stream from one RS232 device only or can be installed in between RS232 transmitter and receiver devices to record both side communications (RX, TX).

Please refer to **"*Extracting the data*"** chapter for available data download options explained.

4. Customization ideas?

Please contact our Team to discuss any firmware and hardware customization request. The RSLogger project is designed as a flexible platform capable to be modified or extended at low customization cost. Make your own, dedicated, stand-alone solution or integral part of a larger system capable to fit your project without changing it's entire architecture.

© *Electroware* 2018

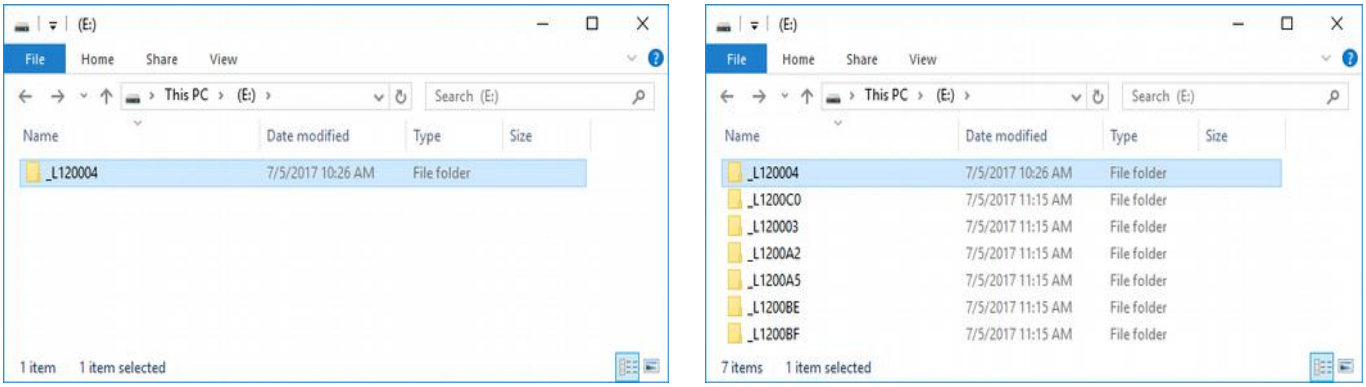


Settings & configuration options

Settings with USB Mass Storage

Simply connect the **FAT32** formatted (**default allocation unit size**) USB Mass Storage to the USB port located on the RSLogger device. By default the password protection is disabled so all USB Mass Storage devices would be accepted. You can protect the access by setting your own password latter.

Connect the USB mass storage to the USB port located on the RSLogger device. The logger's folder would be created on your USB Mass Storage according the device's uinque serial number like: **_L120004** (where **_L** header is constant for all devices, allowing easy folders sorting. Header is followed by 6 bytes long serial number in HEX format like: **120004**). From now on, it's the logger's main operating folder. It's the home for iall data, settings and system configuration files. Unique folder name for every single RSLogger allows to use one Mass Storage drive to handle several different RSLoggers. Each device has it's own directory named by it's unique serial number.



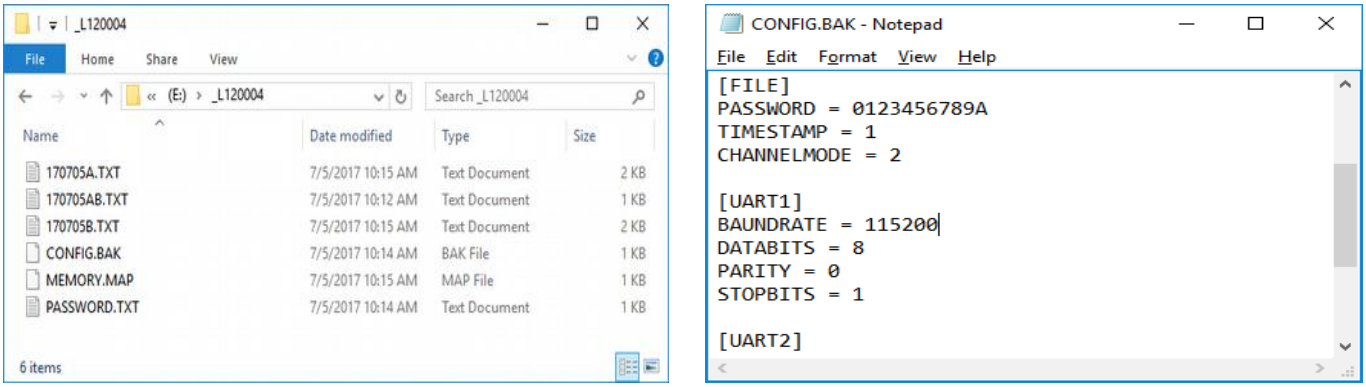
RSLogger main folder on single device and multi-device shared USB Mass Storage.

In above example the device's serial number is 120004. Current configuration would be stored into the **_L120004** folder as a CONFIG.BAK text file. Current device's internal memory map file would be stored in MEMORY.MAP system file.

You can edit the CONFIG.BAK configuration file with any text editor (like Windows Notepad).

© *Electroware* 2018

MEMORY.MAP is non editable binary file and should not be modified nor deleted. Please refer to the **"MEMORY.MAP explained"** chapter for detailed information.



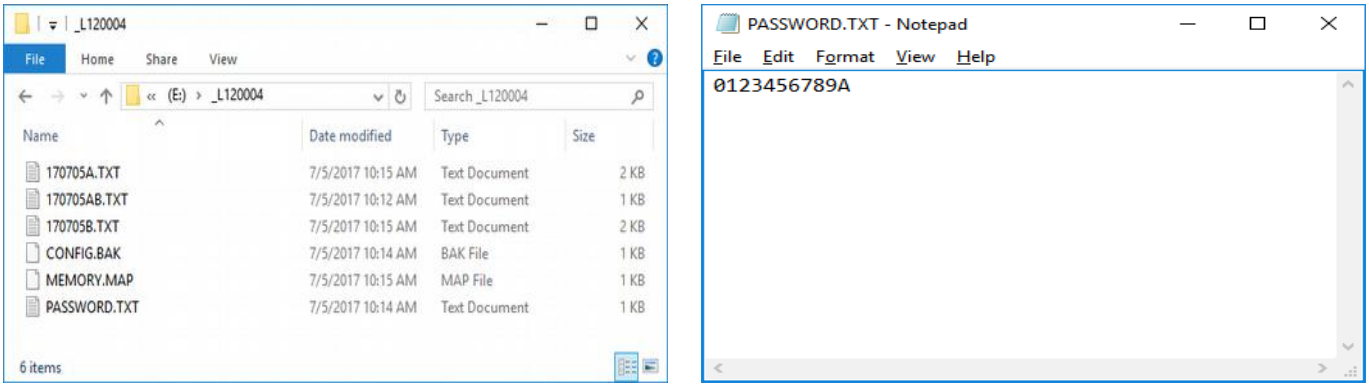
Device's main folder containing data and configuration files ans the CONFIG.BAK file content.

Changes would be applied after renaming the CONFIG.BAK file to **CONFIG.INI** and connecting the USB mass storage back to the RSLogger device to read and apply the changes. CONFIG.INI file would be read and applied, then file name would be changed to CONFIG.BAK accordingly.

Please refer to "**Advanced configuration options with CONFIG.INI file**" table located at the end of this Manual explaining all configuration parameters, values and syntax accepted by CONFIG.INI file.

If password parameter has been changed during CONFIG.INI configuration, the **PASSWORD.TXT** text file would be created in devices folder after new setting applying. Please keep the password file on the USB mass storage (inside the device folder) as long as you want to get access to your protected device with this particular Mass Storage device (it's containing the password string).

USB mass storage devices without actual PASSWORD.TXT file located in the device folder, would not be allowed by RSLogger for data and settings access. It would be not allowed for changing settings as well. An **ERROR.TXT** file with actual status log like *"Incorrect password"* would be created in the device's folder.



PASSWORD.TXT file created on USB Mass Storage is keeping the device's access password.

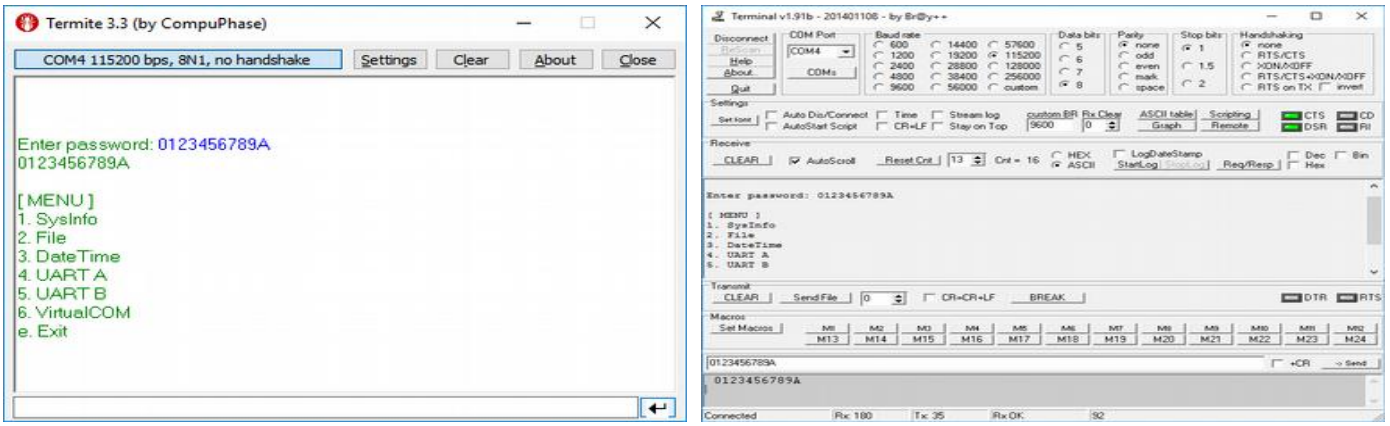
If you want to disable the password please left it blank in CONFIG.INI and the PASSWORD.TXT file would be automatically deleted after password change applying during next USB mass storage connection to RSLogger (or power up). All USB mas storage drives would be accepted from now on.

© *Electroware* 2018

Settings by Virtual COM port (USB) menu

In the Virtual COM mode it's possible to use any serial port terminal application to configure the device with intuitive and self-explanatory text based menu. Connect the Micro USB cable to the RSLogger and your PC. Please refer to the **"FTDI chip installation guide"** chapter if FTDI virtual COM drivers are not yet installed/configured in your system.

Use your own or any third party COM port terminal application to access text menu of the RSLogger device and download the data directly to terminal.



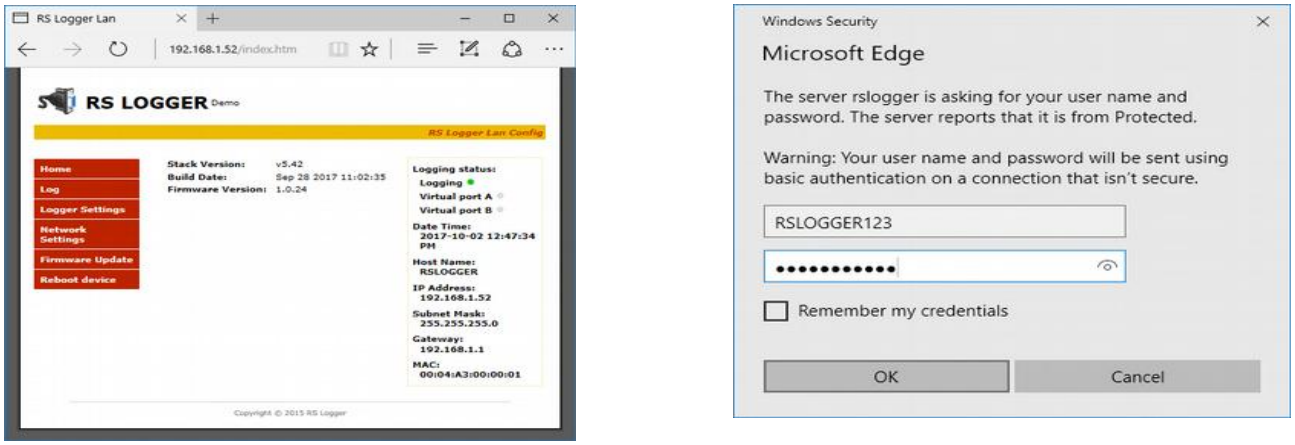
Termite (by CompuPhase) and Terminal (by Br@y++) applications with RSLogger's text menu.

Please refer to the **"Virtual COM terminal text menu mode"** chapter for more details about using the text menu.

Via Ethernet

On Ethernet enabled devices it's possible to download data from WWW interface. Any Internet browser can be used (PC or mobile device connected to the same LAN). Enter the "**http://rslogger/**" as a domain or it's **IP** number directly if known (method recommended on Apple Safari and mobile browsers).

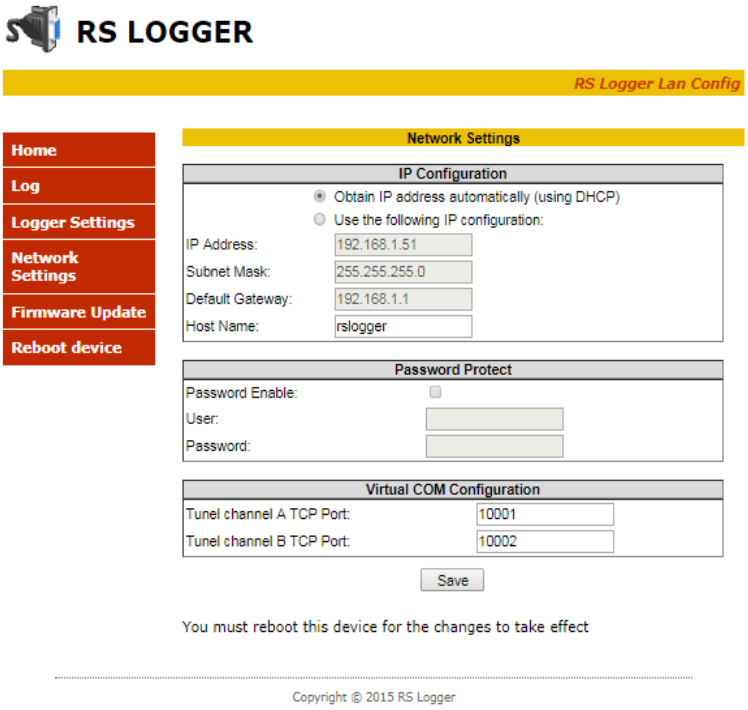
WWW page would be displayed with simple menu, allowing you to configure the device, view and download the data kept in the internal buffer (ASCII representation). It's all available directly from the Internet browser level with use of an intuitive menu.



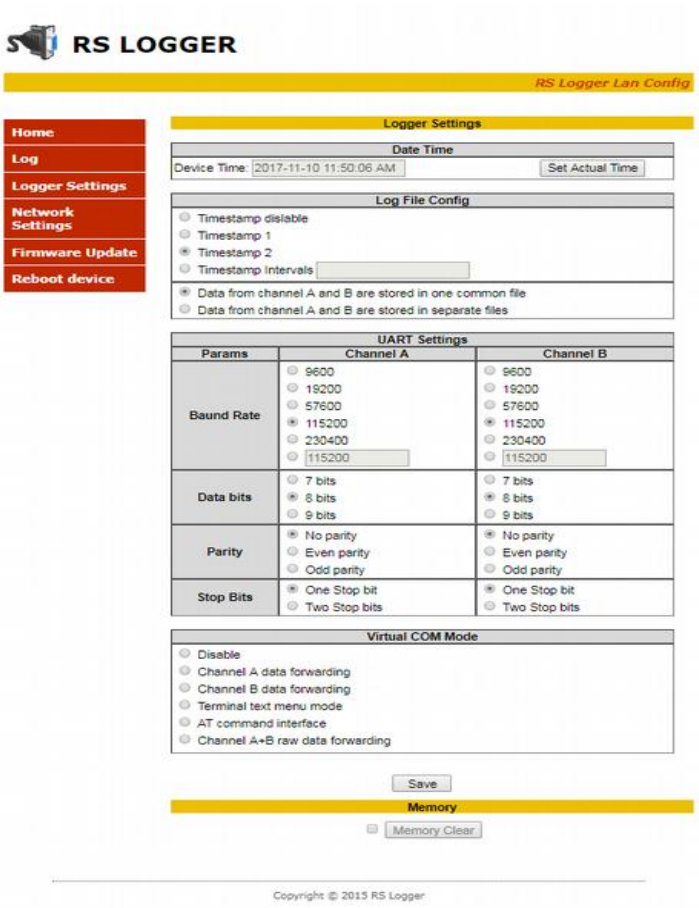
RSLogger's WWW welcome page on Microsoft Edge browser and the login window.

© *Electroware* 2018

By default the WWW interface password is disabled. You can set your own user and password in the Network Settings menu but we recommend to use your Mass Storage password as both user and password to not to mismatch the device's passwords. All device's settings may be managed by two menu optios: Logger Settings and Network Settings.



Network Settings menu: IP address settings, DHCP enable, WWW Password, TCP Virtual COM ports settings.



Logger Settings menu: time set, timestamp, log file mode, RS232 log port A & B settings (baud, parity, etc.)

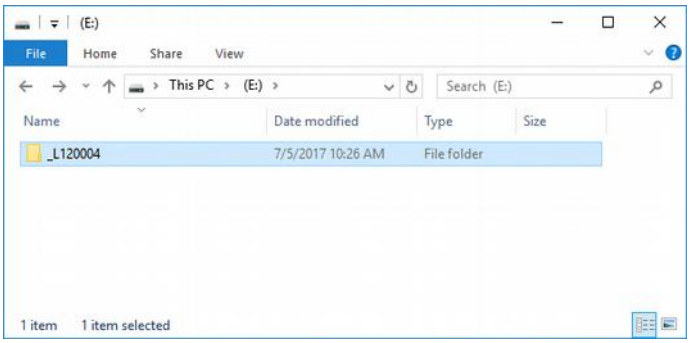
© *Electroware* 2018

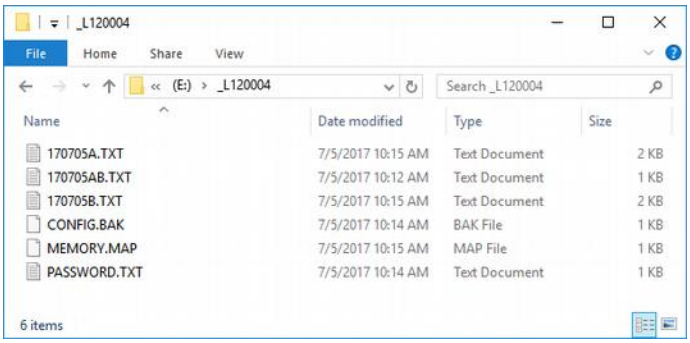
Extracting the data

With USB Mass Storage

Simply connect the **FAT32** formatted (**default allocation unit size**) USB Mass Storage to the USB port located on the RSLogger device.

Notice that if the device was protected by users password, the folder should include the PASSWORD.TXT text file with actual password.





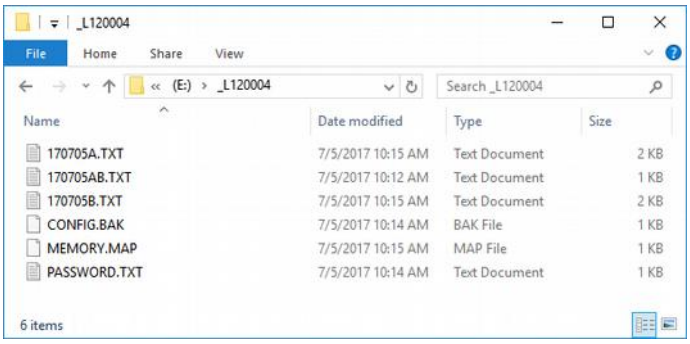
RSLogger's main folder USB Mass Storage device.

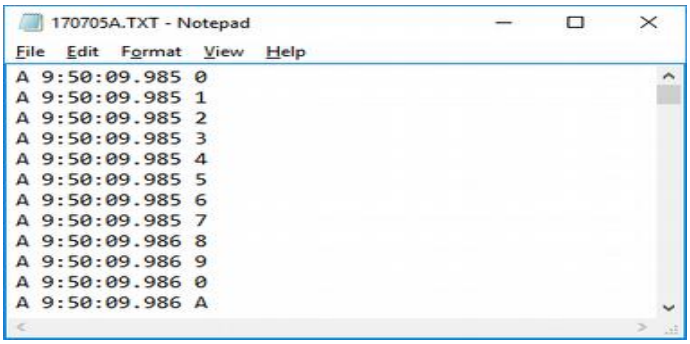
The copy of used internal memory would be created as a set of text files named with the actual logging date in the YYMMDDCC.TXT format (Y-ear, M-onth, D-ay, C-hannel) just like following example: 170626AB.TXT

MEMORY.MAP system file is keeping the actual index information which internal circular-buffer sectors, currently in use, were already downloaded to the backup USB Mass Storage. It's to avoid data duplication or data overlapping in the downloaded text files. Deleting the MEMORY.MAP file will result in index reset. Entire buffer memory would be downloaded to the Mass Storage and some previously downloaded data files existing on the USB Mass Storage may get overwritten (data with same logging date). New actual MEMORY.MAP would be created after each backup to the text file operation.

© *Electroware* 2018

The text files are created day-by-day according the recording date and each day is stored into separated text file. Depending on your preferences it's possible to store the data into one common file like 170626**AB**.TXT or to two separated files for A and B (TX and RX) channels to separate the data accordingly (170626**A**.TXT and 170626**B**.TXT).





RSLogger (S/N: 120004) main folder on USB Mass Storage content after downloading the recorded data.

The internal buffer memory is running in circular mode. Oldest data would be replaced by the new ones if buffer gets full. You can backup the buffer memory any time by connecting the external Mass Storage device. New data from internal memory, would be written (append) to the Mass Storage immediately after USB Mass Storage connection.

You can take the USB Mass Storage out to work with downloaded text files (RSLogger will still record to the internal buffer memory and backup the data on the next USB Mass storage connection) or left it permanently connected.

If the Mass Storage it connected permanently, the buffer memory dumps to USB Mass Storage would be executed automatically in the background task, every minute creating the intelligent up-to-date continuous backups.

Buffer memory itself is capable to keep up to 128MB of formatted data, including precise time stamps, channel indicators etc... Although the internal memory itself is capable to hold months of regular RS232 communication, permanent (or at least periodical) connection of the external USB Mass Storage would make the system's memory practically unlimited.

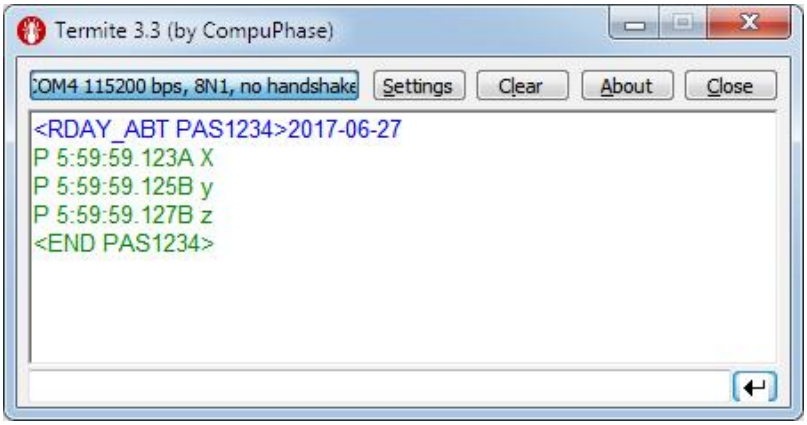
Please refer to **"Memory.MAP explained"** chapter for more details of managing the system's memory.

<AT> command interface (USB Virtual COM)

In the Virtual COM mode set to AT commands it's possible to use any serial port terminal application to download the data to the RS232 Terminal application. Connect the Micro USB cable to the RSLogger and your PC. FTDI chip driver would install automatically enabling the Virtual COM serial port in your system. Please refer to the **"FTDI chip installation guide"** chapter if FTDI virtual COM drivers are not yet installed/configured in your system.

Use your own application or any third party COM port terminal application to access AT commands interface and download the selected data range directly to your terminal.

© *Electroware* 2018

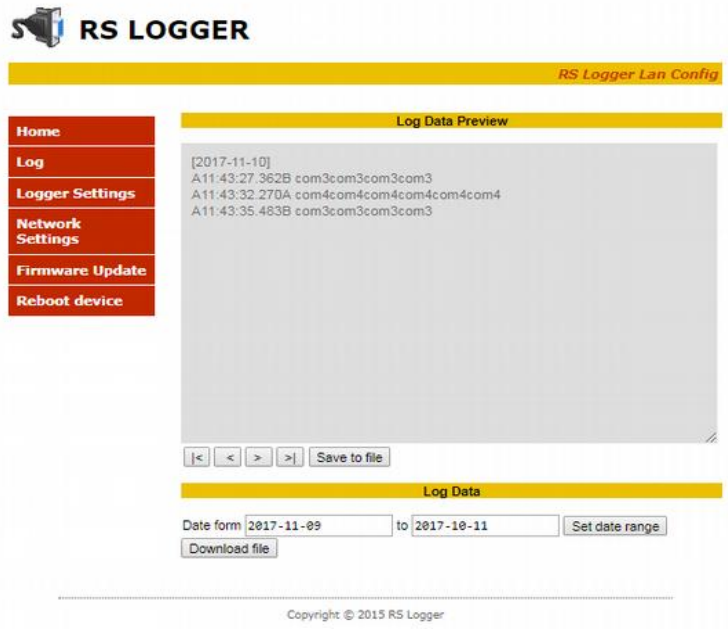


Termite (by CompuPhase) application with RSLogger's AT commands interface (Windows 7)

For more information about AT command interface usage please refer to AT command list, parameters and syntax explained in the "***AT commands (via USB Virtual COM)"*** chapter.

Via Ethernet

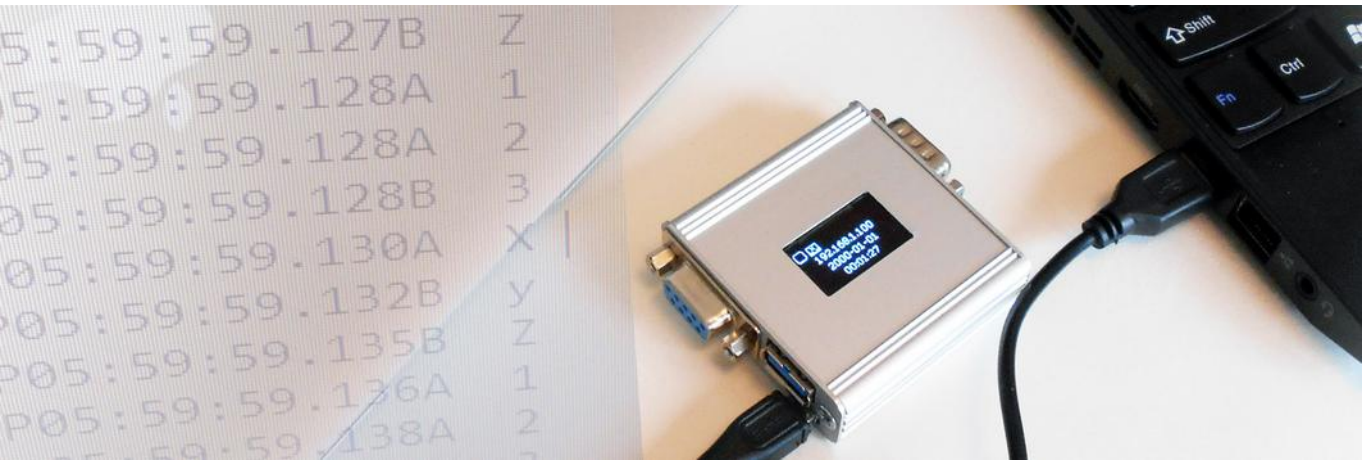
On Ethernet enabled RSLogger versions it's possible to download data from WWW interface. Any Internet browser can be used (PC or mobile device connected to the same LAN). Enter the "**http://rslogger/**" as a domain or it's IP number directly if known like 192.168.1.52 assigned by DHCP in the examples below (method recommended on Apple Safari and mobile browsers).



RSLogger's WWW data download page.

Navigate to the LOG menu allowing you to download the data from the internal memory buffer. Select the request date range and apply it by "Set Data Range" button. Read Log button would be activated then – press it to search and download the data from requested date range. It would take a while to dowlnoad the data, please wait. The current status would be displayed with percentage indicator below text area window. Notice that all non printable characters logged (controll ASCII characters) logged, would be displayed as converted to text (del character or space) depending your WWW browser. Please use the Mass Storage data download mode to get ASCII binary text file stored in the format exactly like it was logged (including all non printable control characters).

© *Electroware* 2018



Data format

Data are recorded into the ECC protected Flash Memory first. Then can be saved as a backup copy to an external USB Mass Storage (ASCII or HEX text format) or served via Ethernet in binary ASCII text format. Text format offers easy data import into any text editor or spreadsheet application if needed. Depending on settings data can be stored into TXT file in four main modes:

- ASCII/binary in common channel file AB: raw data from both A and B channels are stored in one common linear text file like for the example data bytes “XyZ123” below (Channelmode=1, Timestamp=0):

File 170626AB.TXT
XyZ123

- Time Stamped ASCII/binary in common channel file AB: raw data form both A and B channels are stored in one common file but each data byte X is preceded by header containing precise time stamp and channel indicator, tabulation and then followed by new line (/n). Same example bytes “XyZ123” recorded below (Channelmode=1, Timestamp=1):

File 170626AB.TXT
P 5:59:59.123A X P 5:59:59.125B y P 5:59:59.127B Z P 5:59:59.128A 1 P 5:59:59.129A 2 P 5:59:59.130B 3

- ASCII/binary in separated channel files A and B: raw data form both A and B channels are stored in two linear text files, each for one logging channel A & B (Channelmode=2, Timestamp=0):

File 170626A.TXT	File 170626B.TXT
X12	yZ3

- Time Stamped ASCII/binary in separated channel files A and B: data from both A and B channels are stored into two separated TXT files each for one logging channel (A & B). Each data byte X is preceded by header containing precise time stamp, tabulation and then followed by new line byte “/n” (Channelmode=2, Timestamp=1)

File 170626A.TXT	File 170626B.TXT
P 5:59:59.123 X P 5:59:59.124 1 P 5:59:59.129 2	P 5:59:59.125 y P 5:59:59.127 Z P 5:59:59.128 3

© *Electroware* 2018

Please refer to **DATAFORMAT=1** parameter description (page 23) to set the output files into the **HEX mode** where every byte is represented by pair of HEX digits instead of one binary ASCII symbol. Timestamps and channel marks would be still written in ASCII. The HEX bytes are separated with SPACE. Here is the example:

File 170626AB.TXT
P12:30:00.732A 47 56 37 47 F5 16 47 56 37 47 F5 16 P12:30:26.723B 37 56 47 4B F5 A6 47 56 C7 47

Time stamp format

The time stamp format is A/PHH.MM.SS.mmm (A/P – AM/PM, H-hour, M-minute, S-second, m-milisecond) like shown on following examples:

A11:59:59.123 for 11:59:59.123 AM
P 5:59:59.123 for 5:59:59.123PM

Option Timestamp = 1 (available for CHANNELMODE=1 and CHANNELMODE=2)

Every byte has it's own time stamp.

File 170626A.TXT	File 170626B.TXT
A11:59:59.123 X P 5:59:59.124 1 P 5:59:59.129 2	A11:59:59.125 y P 5:59:59.127 Z P 5:59:59.128 3

Option Timestamp = 2 (available only for CHANNELMODE=1 only)

Timestamps are available on active channel switch (common A+B file mode only).

File 170626AB.TXT
A1:02:34.851A Channel A data A1:02:45.440B Channel B data A1:02:47.431A Channel A data

Option Timestamp = delay (available for CHANNELMODE=1 and CHANNELMODE=2)

Delay = 1000 + S, where S is delay expressed in seconds [range of 1 – 43200]. Time stamps marks maximum frequency is set by delay parameter. Timestamp would appear every S seconds, corrected by the actual data byte [ms] precision timestamp. Timestamps are asynchronous for channel A and B (CHANNELMODE=2). Time stamp mark would be stamped on every active channel switch (CHANNELMODE=1). Timestamp with programmed delay would not appear in the log file if there is no data flow. It would appear for new data only if delay between previous and current data is greater than programmed delay = S. New timestamp delay settings can be also applied to the data already stored in internal memory by changing the delay parameter and deleting the Memory.map file.

File 170626AB.TXT for TimeStamp delay set to 2s
A1:02:32.851A Channel A dataChannel A data A1:02:34.440A Channel A dataChannel A data A1:02:36.645A Channel A dataChannel A data A1:02:37.431B Channel B data

© *Electroware* 2018

Time stamping can be disabled by CONFIG.INI option. Both the data representation and time stamp format may be changed on demand by customized firmware release. Please contact RSLogger Team or distributor for pricing details.

Virtual COM modes & COM operations

This chapter is explaining the available modes of the USB Virtual COM interface like Text Menu mode or AT commands mode. Some sample RS232 COM operations like port's splitting, forwarding or repeating options (to Virtual Com or TCP/IP) are described as well.

Virtual COM Text Menu mode.

If Virtual COM interface is enabled and set for text menu mode in config.ini file there would be possible to configure the device with intuitive and self-explanatory text menu displayed in real time in any RS232 text terminal. Standard terminal settings for menu are (115200,8,N1) means 115200 baud rate, 8 data bits, no parity, 1 stop bit - but any other settings should also be accepted as the COM port is virtual. Please send any data to Virtual COM port via terminal to open the menu mode. It will prompt you with "`Enter password:`" message. Type the password and it confirm with [Enter] key. Main Menu should be displayed as follows:

```
[ MENU ]
1. SysInfo
2. File
3. DateTime
4. UART A
5. UART B
6. VirtualCOM
e. Exit
```

```
[ MENU / File ]
Q. <- Back
1. Password
2. Timestamp
3. Channel Mode

[ MENU / File / Channel Mode ]
Q. <- Back
1. One Common file for Both A and B channels
2. Two separated files for A and B channels

ChannelMode = 1
```

Navigate the menu options as you like using your 0-9 keys, Enter, Backspace and "e" key for exit.

The current value of specific parameter would be displayed under the menu options. The [q] key would take you back to upper level menu. If any data needs to be entered manually, please type it in with your terminal keyboard, keeping same data format (date change in the following example). Confirm new value with [Enter] key.

```
[ MENU / DateTime / Date ]
Q. <- Back

Actual Date = 2017-07-07
   New Date = 2017-08-01
```


© *Electroware* 2018

Please notice that disabling the Virtual COM interface or it's text menu mode by changing settings during text menu operations will effect in disabling the menu functionality and end your current terminal menu session!

RS232 COM ports forwarding option

RSLogger offers the "real time" 1:1 RS232 port forwarding to Virtual COM port (1 channel A or B) or Ethernet TCP port (2 channels A and B simultaneously to two different TCP ports).

- **USB Virtual COM**: set the Virtual COM port into the forwarding mode for one of the real RS232 input channels: A or B. Edit the CONFIG.INI file section [VIRTUALCOM] to set parameter

  VIRTUALMODE = 1 (for channel A)
  VIRTUALMODE = 2 (for channel B)
  VIRTUALMODE = 5 (for hannels A+B forwarding, raw mixed)

  Example below:

  [VIRTUALCOM]
  VIRUALCOMMODE = 1

  The selected RS232 channel data stream would apper on the USB Virtual COM port served 1:1 in the real time (like for the hardware RS232 spliter).

- **Ethernet TCP Virtual COM**: set the TCP port for one or botch RS232 input channels to forward the data 1:1 to the TCP Virtual COM port. Notice that forwarded data may not always be served in the real time as it's depending on the RS232 data rate and actual LAN traffic and topology.

  TCP Virtual COM Ports may be set with WWW interface or CONFIG.INI file be editing the [ETHERNET] section parameters **TCOM1MODE** and **TCOM2MODE** to one of the proper values:

  0 - to disable the Virtual COM port (default)
  1 - for RS232 channel A data forwarding
  2 - for RS232 channel B data forwarding

  and the corresponding **TCOM1PORT** or **TCOM2PORT** parameters with requested TCP port number.

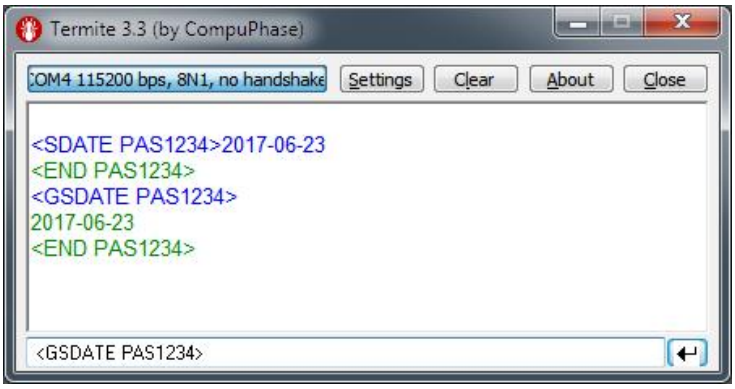
  Here is the CONFIG.INI [ETHERNET] section example:

  [ETHERNET]
  HOSTNAME=rslogger
  IP=192.168.1.100
  MASK=255.255.255.0
  GATEWAY=192.168.1.1
  TCOM1MODE=1
  TCOM1PORT=22334
  TCOM2MODE=2
  TCOM2PORT=22335

The selected RS232 channel data streams would appear on the selected TCP Virtual COM port served 1:1 in the near to real time (like for the hardware RS232 to Ethernet LAN port server/repeater). Create the TCP Virtual COM port in your operating system (with use of one of available third-party software solutions) and use the COM port terminal client software to get connected and receive the forwarded data.

© *Electroware* 2018

AT commands (via USB Virtual COM)



Use any RS232 terminal application or RS232 interfaced hardware to send the AT text commands to the RSLogger device in Virtual COM mode set to AT commands interface mode (VIRTUALCOMMODE=4). Each command to get accepted needs to carry the actual device password in the command syntax. Each command executed will result in <END password> footer returned by RSLogger, indicating the command execution or data flow end.

Free Termite terminal (by Compu Serve) in the example shown on the left.

AT command	Parameter	Description
<GTIME password>	none	Get the actual device time.
<GDATE password>	none	Get the actual device date.
<STIME password>	A/Phh:mm:ss	Set the actual device time.
<SDATE password>	yyyy-mm-dd	Set the actual device date.
<RALL_ABT password>	none	Read all the data from A & B channels with TimeStamp.
<RALL_AT password>	none	Read all the data from A channel with TimeStamp.
<RALL_BT password>	none	Read all the data from B channel with TimeStamp.
<RALL_AB password>	none	Read all the data from A & B channels no TimeStamp.
<RALL_A password>	none	Read all the data from channel, no TimeStamp.
<RALL_B password>	none	Read all the data from B channel, no TimeStamp.
<RDAY_ABT password>	yyyy-mm-dd	Read data from specific day, both channels A&B with TimeStamp.
<RDAY_AT password>	yyyy-mm-dd	Read data from specific day, both channel A with TimeStamp.
<RDAY_BT password>	yyyy-mm-dd	Read data from specific day, both channel B with TimeStamp.
<RDAY_AB password>	yyyy-mm-dd	Read data from specific day, both channels A&B, no TimeStamp.
<RDAY_A password>	yyyy-mm-dd	Read data from specific day, channel A, no TimeStamp.
<RDAY_B password>	yyyy-mm-dd	Read data from specific day, channel B, no TimeStamp.
<MCLR password?	none	Internal memory clear

Examples:

Device password is PAS1234 in the examples below. To process AT command please send the following ASCII string with your terminal application. Enn the command with carriage return CR (0x0D ASCII) or line feed “LF” (0x0A ASCII) control byte. Commands are marked green in the examples, parameters are blue and control bytes like “LF” or “CR” are red.

Set date to 2017-06-23: **<SDATE PAS1234>17-06-23”CR”**

Set time to 06:12:23:PM: **<STIME PAS1234>P06:12:23”CR”**

Read data recorded on 2017-06-23 channel B with TimeStamp: **<RDAY_B PAS1234>17:06:23”CR”**

As an effect of any AT command executed the RSLogger will respond with specific, unique end of data marker: **<END password>** allowing to determine the data flow end or command execution status within binary data stream.

© *Electroware* 2018

MEMORY.MAP & memory usage explained

The MEMORY.MAP file is an index file keeping the information about current and historical internal memory usage. It means it keeps the information of what data sectors were already saved to the actually connected USB Mass Storage. The MEMORY.MAP file is related to the particular USB Mass Storage device considered as a backup storage - not the RSLogger itself. According to MAP structure the RSLogger device is capable to backup the new data to USB Mass Storage without duplication, skipping or overlapping the data in text files created.

If data gets recorded they are stored in internal memory first. The internal memory cell is marked as “occupied” by new data. If the USB Mass Storage is connected the intelligent backup procedure would search for MEMORY.MAP file first to obtain information about last index of the internal buffer already saved to the text file. The background task will download the new data to USB Mass Storage and MEMORY.MAP would be updated accordingly. The information what indexes of internal memory sectors were stored to the USB Mass Storage would be written to MEMORY.MAP.

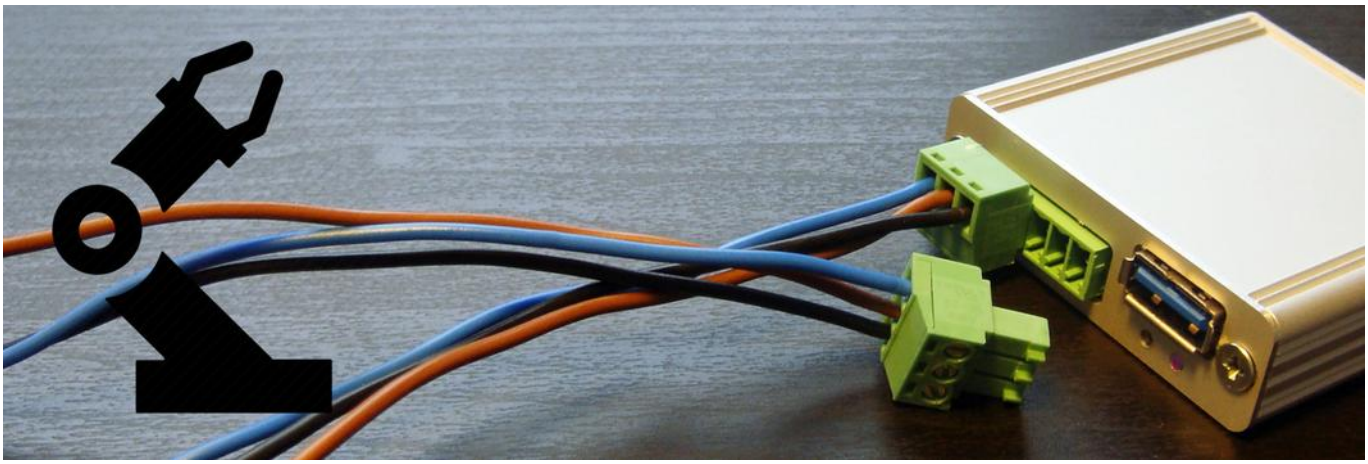
MEMORY.MAP can be unique/different for every USB Mass Storage device used with same RSLogger. You can use sevral different USB Mass Storage devices to download the data from same logger, connected in different time.

Deleting the MEMORY.MAP file will result in backup map erasing and all available data from internal memory would be considered as new data, ready to be saved in the text file form again. Please notice that without MEMORY.MAP file some logged data may get permanently lost (would not be backed up) if the internal memory buffer overlaps already resulting in erasing the oldest data.

Please notice that in case of high RS232 data rates and when using the RSLogger without connecting the USB Mass Storage at all (or when the USB Mass Storage is connected very rarely), some oldest data in internal memory circular buffer may get overwritten. In such a case the MEMORY.MAP would still be updated and all new data available would be saved to TXT files upon the next USB Mass Storage connection. If old data was saved to this Mass Storage already before, some data gap may occur in saved backup – adequate to overwritten data size when internal circular buffer overlaps.

It’s not recommended to delete the MEMORY.MAP file manually, but it can be done if user needs to save all internal memory data available in the RSLogger buffer. It can be useful for changing the saved files data format (for example to download same already saved data but in different format, for example with TimeStamps). Please proceed with care.

© *Electroware* 2018



Handling the mulitple-device systems

With data backup system applied in RSLogger it's possible to handle several different RSLogger devices with one common USB Mass Storage. It's handy and intuitive as every single RSLogger will create it's own folder, named with unique serial number and filled with it's unique configuration, password, data and MEMORY.MAP files.

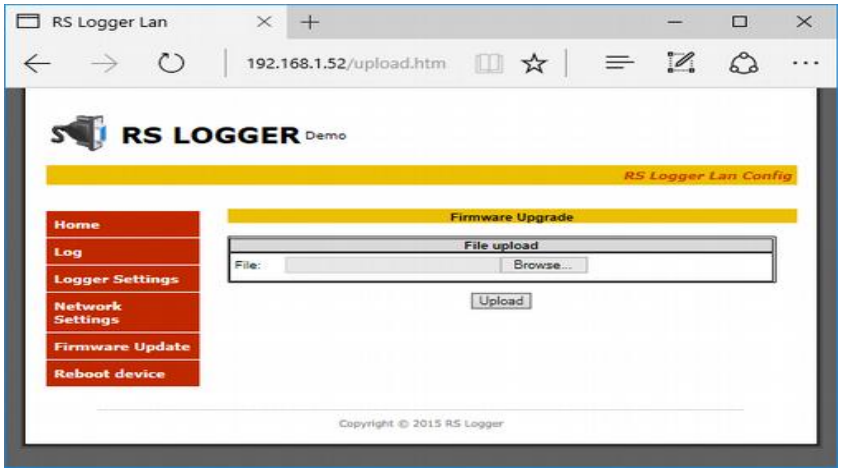
You can take the USB Mass Storage and have a walk from one RSLogger unit to another (when planted in your project) to collect the data to common USB Mass Storage device for further analysis.

Firmware update

RSLogger offers the firmware update option allowing to update the current firmware to any new released version. The firmware update is protected with several security features to avoid wrong firmware uploading or device disabling if update procedure has not been accomplished successfully (for example due to the power failure). There are three ways for firmware update and two of them are available for the end-users:

- Update with USB Mass Storage: simply store the firmware file on the USB Mass Storage device used for data download (password labeled) and connect it to the RSLogger device. If the FIRMWARE.BIN file has been found on the connected USB mass storage drive and the firmware file format is supported for this particular RSLogger model, it would be flashed into the internal memory. The firmware file would be renamed to FIRMWARE.BAK and left on the USB Mass Storage device as a proof of firmware update. RSLogger will restart automatically to finish the update procedure and run new firmware.
- Update with Ethernet connection: for devices equipped with Ethernet interface there is a special Menu (Firmware Backup) option on the WWW maintenance page allowing to remotely update the firmware version by browsing the firmware file on the local PC, just by clicking from the Web browser level.

© *Electroware* 2018



Firmware Update option of the WWW interface

- Update by VirtualCOM connection and external maintenance application (method reserved for manufacturer's service staff only)

Back to factory default firmware & settings

It's a rare case but no matter what went wrong with firmware update or settings (or the password was forgotten) it's always possible to return to original factory firmware and settings. There is a small black push-button located on the RSLogger panel. Press it down (with pencil or match) and turn on the device's power with button pressed. Hold the button for 5 seconds and release when LED2 (green) stops to blink (see OLED screen if available).

The back to factory default firmware procedure would be initiated (both LEDs will blink for a while) and device restored to default, factory state.

Notice that for security reasons all internal buffer memory will be erased with back to default function. Access password would be disabled (no password protection).

For RSLogger Module devices there is no push-button installed but the "back to default firmware" feature is still available with GPIO1 and GPIO2 ports located on the device's main terminal. Pelase short the GPIO1 to GPIO2 when the device is powered off. Turn on the RSLogger Module power and then after 5 seconds remove the GPIO1/GPIO2 jumper. The back to factory default firmware procedure would be initiated (both LEDs will blink for a while) and device restored to default, factory state.

Internal memory erase

If the total erase of internal memory is required it's possible with MEMCLEAR = TRUE parameter of the CONFIG.INI file, [SYSINFO] section. Edit the CONFIG.OLD file as shown on example below, rename itto CONFIG.INI and restart your device (or re-connect the USB Mass Storage containing CONFIG.INI file).

```
[SYSINFO]
FW_VER 1.0.31
HW_VER 1
MEMCLEAR = TRUE
```

Before memory clear operation, to avoid confusion, it's recommended to manually delete all data TXT files and MEMORY.MAP file that already exists in RSLogger working folder.

Notice that all data from internal memory would be permanently lost!

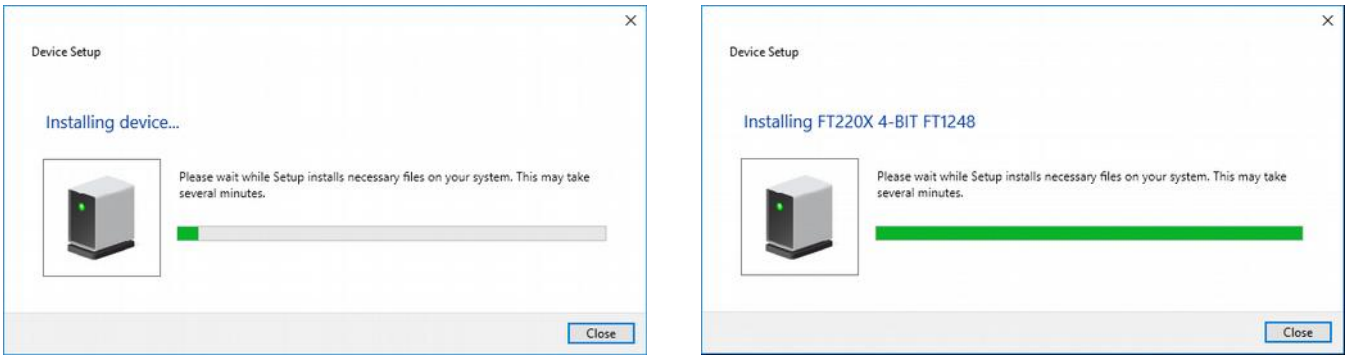
© *Electroware* 2018

FTDI chip Virtual COM drivers installation

To use the Virtual COM it's necessary to install the FTDI chip drivers. Most of the modern systems is capable to install the drivers automatically (like MS Windows 7, 8 & 10). FTDI devices will work with the following operating systems:

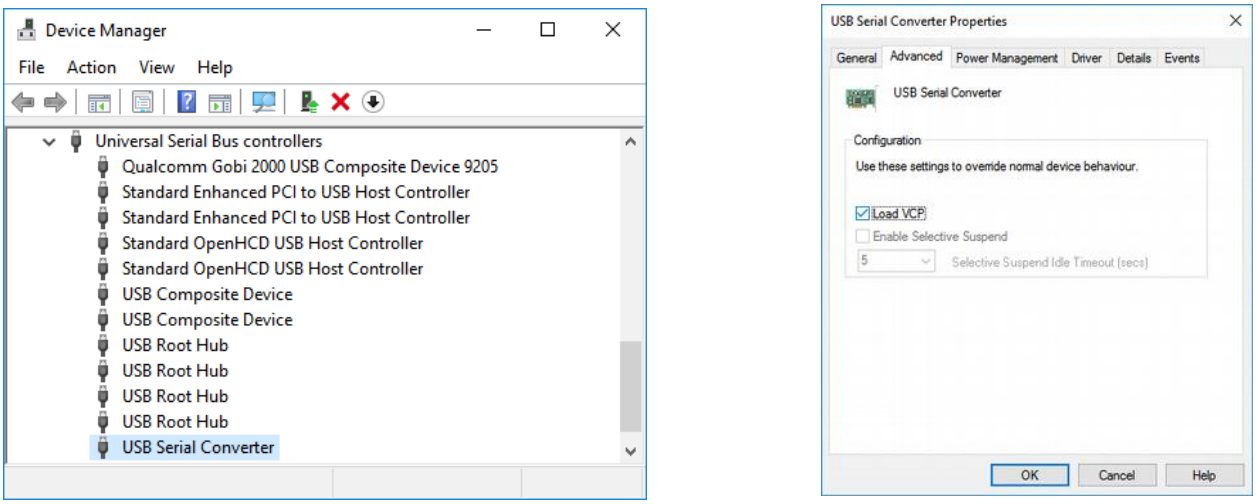
Windows Certified	Others	Legacy
Windows 10 (32/64)	Linux	Windows Vista (32/64)
Windows 8.1 (32/64)	MAC OSX	Windows XP (32/64)
Windows 8 (32/64)	Windows CE (Version 4.2 and greater)	Windows 2000
Windows 7 (32/64)	Windows RT	Windows ME
Windows Server 2012 R2 x64	Android	Windows 98
Windows Server 2008 R2 x64		MAC OS9
		MAC OS8

Here is a sample installation process on Microsoft Windows 10. Connect the new device to your PC's USB port. You should get the new hardware found dialog and setup as follows:



FTDI automatic driver installation process on Windows 10

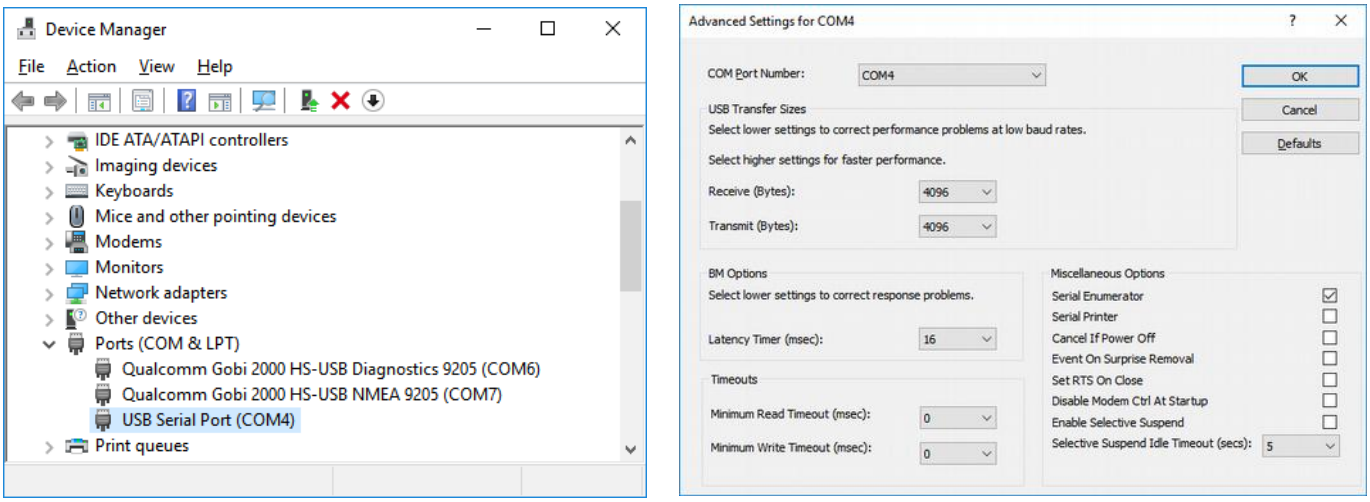
Important: make sure to enable VCP support in the USB Serial Converter device driver properties (if disabled) otherwise FTDI chip would not be visible as a standard serial COM Port but as a USB device only.



FTDI USB Serial Converter device on Windows Device Manager list and it's advanced properties for VCP module loading.

Restart the device after changing the VCP mode in driver settings to load the virtual com port module. Additional new device "USB Serial Port" should be found in ports. It's possible to change the COM port number in advanced settings.

© *Electroware* 2018



FTDI Virtual COM Port (VCP) module loaded and it's advanced settings.

In case the OS would not load the drivers automatically, please choose the "update driver" option manually or visit the http://www.ftdichip.com/ page for actual driver updates and installation support for your operating system.

Advanced configuration options with CONFIG.INI file

INI Section	Parameter	Example Value (default)	Description
[SYSINFO]	FW_VER	1.1.51	Firmware version. Read only
	HW_VER	1.1	Hardware version. Read only.
	MEMCLEAR	TRUE	Erases the internal memory upon next power up or USB Flash Mass storage connection.
[FILE]	PASSWORD	RSLOGGER123	USB Flash drive password label. Blank by default.
	TIMESTAMP	0	0-disable, 1- Enabled for every byte, 2- Enabled for active channel switch only
	CHANNELMODE	1	1 - one common file for both A and B channels 2 - two separated files for A and B channels
	DATAFORMAT	0	0 - ASCII / binary 1 – HEX
[DATETIME]	SETDATE	;SETDATE = 2018-01-01	Date setting. Applicable upon next restart or USB Flash drive connection (with config.ini). Last set time would be stored as a comment ";" in config.txt file.
	SETTIME	;SETTIME = 12:00:00PM	Time setting. Applicable upon next restart or USB Flash drive connection (with config.ini). Last set time would be stored as a comment ";" in config.txt file.

© *Electroware* 2018

[UART_A]	BAUDRATE_A	115200	Channel A UART port baudrate setting. Range [128 – 230400]
	DATABITS_A	8	Channel A UART port data bits setting. Values: 7, 8, 9 Please refer to comments at the bottom of this page!
	PARITY_A	0	Channel A UART port parity setting. 0 – no parity 1 – even parity 2 – odd parity. Please refer to comments!
	STOPBITS_A	1	Channel A UART port stop bits setting. 1 - (one stop bits) 2 - (two stop bits)
[UART_B]	BAUDRATE_B	115200	Channel B UART port baudrate setting. Range [128 – 230400]
	DATABITS_B	8	Channel B UART port data bits setting. Accepted values: 7, 8, 9 Please refer to comments!
	PARITY_B	0	Channel B UART port parity setting. 0 – no parity 1 – even parity 2 – odd parity. Please refer to comments!
	STOPBITS_B	1	Channel B UART port stop bits setting. 1 - (one stop bits) 2 - (two stop bits)
[VIRTUALCOM]	VIRTUALCOMMODE	0	Virtual COM feature mode. 0 – disabled 1 – Channel A data forwarding 2 – Channel B data forwarding 3 – Terminal text menu mode 4 – AT commands interface 5 – Channel A+B data forwarding (raw mixed)
[ETHERNET]	HOSTNAME	rslogger	Device Host name for LAN.
	IP	192.168.1.100	Device's IP address.
	MASK	255.255.255.0	IP address mask.
	GATE	192.168.1.1	LAN Gateway IP address.
	TCOM1MODE	0	TCP COM1 port mode: 0 – disabled 1 – Channel A data forwarding 2 – Channel B data forwarding
	TCOM1PORT	22334	TCP port number for TCP COM1
	TCOM2MODE	0	TCP COM2 port mode: 0 – disabled 1 – Channel A data forwarding 2 – Channel B data forwarding
	TCOM2PORT	22335	TCP port number for TCP COM2

Comments: according to RS232 data frame format the databits and parity settings are related. Some abnormal combinations of these parameters are forced or prohibited. For example: for databits = 7 the parity have to be always enabled (odd or even) and for databits =9 the parity has to be always disabled (parity = 0).

© *Electroware* 2018

Technical specifications

	RSLogger PRO	RSLogger Basic	RSLogger ECO	RSLogger Module
Net dimensions W x H x L [mm] (incl. connectors)	55 x 16 x 66	55 x 16 x 66	31 x 16 x 60	22 x 11 x 24.5 22 x 11 x 28.5 (with mounting flange)
Net weight [g]	62	57	57	7
Power supply	5 VDC	5 VDC	5 VDC	5 VDC (ECO) 5-24 VDC
Internal memory	512 MB Nand Flash	512 MB Nand Flash	512 MB Nand Flash	512 MB Nand Flash
USB Mass Storage Support	USB 2.0 FAT32	USB 2.0, FAT32	USB 2.0, FAT32	USB 2.0, FAT32
USB data download speed	1.2 Mb/s	1.2 Mb/s	1.2 Mb/s	1.2 Mb/s
Virtual COM (USB)	FTDI, 12 Mb/s	FTDI, 12 Mb/s	-	-
Ethernet (RJ45)	10BASE-T	-	-	-
Connectors	Dsub-9M (Ch A) Dsub-9F (Ch B) USB-A, MicroUSB RJ45 (Ethernet) Push-button	Dsub-9M (Ch A) Dsub-9F (Ch B) USB-A, MicroUSB Push-button, 2 status LEDs	Dsub-9M (Ch A) Dsub-9F (Ch B) USB-A MicroUSB	7-line terminal USB-A MicroUSB
Display Unit	OLED 128x64	Status LEDs	-	Status LEDs
Real Time Clock unit on board	Yes, 1ms time stamp precision	Yes, 1ms time stamp precision	Yes, 1ms time stamp precision	Yes, 1ms time stamp precision
RTC backup battery on board	Rechargeable 2mAh	Rechargeable 2mAh	Rechargeable 2mAh	Rechargeable 2mAh

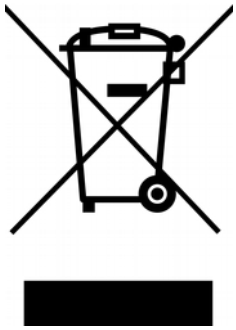
	RSLogger PRO Industrial	RSLogger Basic Industrial	RSLogger Module Industrial
Net dimensions W x H x L [mm] (incl. connectors)	22 x 11 x 24.5 22 x 11 x 28.5 (with mounting flange)	55x16 x55	22 x 11 x 24.5 22 x 11 x 28.5 (with mounting flange)
Net weight [g]	70	65	10
Power supply	5-24 VDC	5-24 VDC	5-24 VDC
Internal memory	512 MB Nand Flash	512 MB Nand Flash	512 MB Nand Flash
USB Mass Storage Support[1)]	USB 2.0, FAT32	USB 2.0, FAT32	USB 2.0, FAT32
USB data download speed	1.2 Mb/s	1.2 Mb/s	1.2 Mb/s
Virtual COM (USB)	FTDI, 12 Mb/s	FTDI, 12 Mb/s	FTDI, 12 Mb/s
Ethernet (RJ45)	10BASE-T	-	-
Connectors	7-line terminal USB-A, MicroUSB, Push-button	7-line terminal USB-A, MicroUSB, Push-button	7-line terminal USB-A, MicroUSB
Display Unit	OLED 128x64	Status LEDs	Status LEDs
Real Time Clock unit on board	Yes, 1ms time stamp precision	Yes, 1ms time stamp precision	Yes, 1ms time stamp precision
RTC backup battery on board	Rechargeable 2mAh	Rechargeable 2mAh	Rechargeable 2mAh

[1)]Please notice that only FAT32 Mass Storage (default allocation unit size) formatted drives are supported.

© *Electroware* 2018

See more on ***www.rslogger.com*** page!

DEALER CONTACT:



© *Electroware* 2018