ELIUM DAB+/FM Radio RS-232/Network Remote Control Description

1. General

Date: 20.01.2016
Revision: 0.01
Scope: The goal of this document is to describe how ELIUM Radio Device can be controlled through RS232 connector (RS232-RC mode) or via network TCP connection (NET-RC mode).

2. The RS232/Network attachment

One of the many features implemented in ELIUM Radio Device application is the possibility of bidirectional controlling the device through RS232 connection or via network TCP connection according to this Remote Control description.

3. Example application

ELIUM Radio Device can be controlled from your PC. Be aware that only two wires of nine are used (RX and TX) in the case of RS232-RC mode. The TCP port 26 is the default communication port in the case of NET-RC mode.

4. Working conditions

The communication can work correctly only if the following conditions are fulfilled.

For the RS232-RC mode (via RS232 connection):

- Baud: 115.200 (default)
- Parity: none
- Data Bits: 8
- Stop Bits: 1
- Flow Control: none

For the NET-RC mode (via network TCP connection):

The TCP communication port is configurable by the server side (ELIUM Radio Device) but selected port number should not be used by other applications. The client (PC) connects to the server (ELIUM Radio Device) with its network address and port via TCP and fulfils the commands described below to control the device.

5. Attention:

Please mention that after switching on the unit by pushing the Power Switch, the unit is starting and during this procedure should not be disturbed. If you send anything during the starting procedure, the unit can go to Firmware update procedure. So it is recommended waiting until receive text information from application part - "#READY: mode1,mode2". This information depends on last mode the device was switched to. (IDLE, DAB+, FM)

6. Note:

In certain moments ELIUM Radio Device sent other "#" lines too.

They give information about: Boot, Application Version etc. These lines should not be taken into account.

7. Commands without additional return value

Each command starts with "<" char and ends with ">". Immediately after ">" sign is received, command will be performed.

If command is not recognized (for example, if <ABC> command is sent), the following text should appear on your terminal window:

#COMMAND: <ABC>
#ERROR: Command not supported

If command is supported and was received correctly you should get something like:

#COMMAND : <ON>
#OK

The line "#COMMAND:" is sent before command is performed. It only indicates that certain string of chars was received by Receiver. After that, command is performed and, if this action is finished, the line "#OK" should be sent.

In order to simplify (from programmer point of view) the reception of responses (so called confirmations) the first sign sent from Receiver is always "#". So, host should wait for "#", the next letter should indicate whether everything was all right or not (#C, #E or #C, #O)

Command	**Description**
<ON>	Turn on device (doesn´t work in normal mode) Examples: #COMMAND: <ON> #OK or #COMMAND: <ON> #ERROR: Not in standby
<OFF>	Turn off device (doesn´t work in Standby mode) Examples: #COMMAND: <OFF> #OK or #COMMAND: <OFF> #ERROR: Already in standby
<REB>	Reboot device

<SSS n>	Set RS232 Baud Rate / Speed Example values are: 9600, 19200, 38400, 115200 but user can set any speed. However, Boot Loader will not change its Baud Rate which is 115.200. Only application may work with the new speed. Baud Rate will be changed immediately. No #OK: confirmation is given but if an error occur, #ERROR: line is sent. Example: #COMMAND: <SSS 9600>
<SELECT n>	Select tuner. n = Tuner 0 – Tuner 1 1 – Tuner 2 Example: #COMMAND: <SELECT 0> #OK
<IDLE MODE>	Switch radio mode to IDLE. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Examples: #COMMAND: <IDLE MODE> #OK or #COMMAND: <IDLE MODE> #ERROR: Device is busy
<FM MODE>	Switch radio mode to FM and enable FM Radio functions. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Examples: #COMMAND: <FM MODE> #OK

	or #COMMAND: <FM MODE> #ERROR: Device is busy
<DAB MODE>	Switch radio mode to DAB / DAB+ and enable DAB / DAB+ Radio functions. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Examples: #COMMAND: <DAB MODE> #OK or #COMMAND: <DAB MODE> #ERROR: Device is busy
<DAB SEARCH>	Start DAB / DAB+ automatic channel search. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Examples: #COMMAND: <DAB SEARCH> #OK or #COMMAND: <DAB SEARCH> #ERROR: Already searching or #COMMAND: <DAB SEARCH> #ERROR: Device is busy
<FM SEARCH>	Start FM automatic channel search. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command

	fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM SEARCH> #OK or #COMMAND: <FM SEARCH> #ERR0R: Already searching or #COMMAND: <FM SEARCH> #ERROR: Device is busy
<FM TUNE F n>	Tune FM Radio to frequency. n = frequency in 10 kHz (102.4 MHz = 10240) Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM TUNE F 10240> #OK
<RMC n>	Simulates an input via remote control. n = remote control keycode for the given key Possible keycodes (for device remote control) are: 22 Key ON/OFF 2 Key '1' 5 Key '2' 6 Key '3' 9 Key '4' 10 Key '5' 13 Key '6' 14 Key '7' 17 Key '8' 18 Key '9' 21 Key '0' 34 Key MODE 37 Key RADIO/TV 38 Key MUTE 41 Key LAST

	25 Key UP 26 Key DOWN 29 Key LEFT 33 Key RIGHT 30 Key OK 42 Key MENU 45 Key EXIT 58 Key Red (DVR / PVR) 61 Key Green (DVD / MOVIE) 62 Key Yellow (MP3/FPEG / MUSIC) 65 Key Blue (GAME / MEDIA) 46 Key << (rew. back) 49 Key PLAY/PAUSE 50 Key >> (rew. forward 53 Key |<< (go prev.) 54 Key REC/STOP 57 Key >>| (go next) 66 Key INFO 69 Key EPG 70 Key TIMER 73 Key TXT 74 Key PIP 77 Key SEARCH/FREEZE 78 Key TECH.INFO/ZOOM 81 Key AUDIO VIDEO It is also possible to emulate any other user defined keys with the keycodes in decimal form different from the above values. Example: #COMMAND: <RMC 30> #OK
<SIP s>	Change device network configuration s = list of string attributes separated with “;” delimiter in format: *ipaddr;netmask;gateway;dns* where *ipaddr* = device IP-address or “-” if no change; *netmask* = device network mask or “-” if no change; *gateway* = network gateway IP-address or “-” if no change; *dns* = DNS server IP-address or “-” if no change; Example: #COMMAND: <SIP 10.1.1.54;-;10.1.1.1;10.1.1.1> #OK

<SDM n>	Set HDMI video display mode n = decimal code for HDMI video display mode Possible decimal codes for HDMI video display mode are: 0 : 480i 1 : 576i 2 : 480p 3 : 576p 4 : 720p 50Hz 5 : 720p 60Hz 6 : 1080i 50Hz 7 : 1080i 60Hz 8 : 1080p 24Hz 9 : 1080p 50Hz 10 : 1080p 60Hz 11 : 1080p 25Hz 12 : 1080p 30HZ Command should be used only in Working Mode (out of Standby). Otherwise command fails with the following error message: #ERROR: Command not allowed Example: #COMMAND: <SDM 7> #OK
<WEBAPP SYNC n s>	Update ELIUM WebApp Engine GUI Application Please, refer ELIUM Radio App Development Guide for details. n = [NET | SDC | USB] – update container s = list of parameters depending of n (see below) Possible update containers are: 1. n = NET Update GUI WebApp from network share (file server). In this case '*s'* is the list of string attributes separated with “;” delimiter in format: *server_ip;share_name;webapp_path;username;password* where *server_ip* = IP-address of the network file server; *share_name* = network shared resource name; *webapp_path* = relative path to GUI WebApp folder in side shared folder;

	username = network file server user name or "-" if empty for anonymous (guest) login; *password* = network file server user password or "-" if empty (no password); 2. n = SDC Update GUI WebApp from SD card. In this case *'s'* is relative path to GUI WebApp folder inside the SD card root filesystem. 3. n = USB Update GUI WebApp from USD drive. In this case *'s'* is relative path to GUI WebApp folder inside the USB drive root filesystem. Device sends error message if update fails for some reasons: #ERROR: *error_description* Examples: To update GUI WebApp from network share #COMMAND: <WEBAPP SYNC NET 10.1.1.5;Public;webapp;-;-> #OK To update GUI WebApp from SD card #COMMAND: <WEBAPP SYNC SDC webapp> #OK To update GUI WebApp from USB drive #COMMAND: <WEBAPP SYNC USB webapp> #OK
<APPDATA SYNC n s>	(Re)write ELIUM WebApp Engine GUI Application data file (e.g. TV/Radio channellists, GUI WebApp configuration etc) Please, refer ELIUM Radio App Development Guide for details. n = [NET | SDC | USB] – data file container s = list of parameters depending of n (see below) Possible data file containers are: 1. n = NET (Re)write GUI WebApp data from network share (file server). In this case '*s'* is the list of string attributes separated with ";" de-

	limiter in format: *server_ip;share_name;file_path;username;password* where *server_ip* = IP-address of the network file server; *share_name* = network shared resource name; *file_path* = relative path to GUI WebApp data file inside shared folder; *username* = network file server user name or "-" if empty for anonymous (guest) login; *password* = network file server user password or "-" if empty (no password); 2. n = SDC (Re)write GUI WebApp data from SD card. In this case *'s'* is relative path to GUI WebApp data file inside the SD card root filesystem. 3. n = USB (Re)write GUI WebApp data from USD drive. In this case *'s'* is relative path to GUI WebApp data file inside the USB drive root filesystem. Device sends error message if (re)write fails for some reasons: #ERROR: *error_description* Examples: (Re)write GUI WebApp TV channellist from network share #COMMAND: <APPDATA SYNC NET 10.1.1.2;Public;webapp;-;-> #OK (Re)write GUI WebApp Radio channellist from SD card #COMMAND: <APPDATA SYNC SDC webapp> #OK (Re)write GUI WebApp TV channellist from USB drive #COMMAND: <APPDATA SYNC USB webapp> #OK

8. Commands with additional return value

Each command starts with "<" char and ends with ">". Immediately after ">" sign is received, command will be performed.

If command is not recognized (for example, if <ABC> command is sent), the following text should appear on your terminal window:

#COMMAND: <ABC>
#ERROR: Command not supported

If command is supported and was received correctly you should get something like:

#COMMAND: <GCS>
#RET: on
#OK

Command	**With Return Value**
<VER>	Get device firmware and hardware information. Example: #COMMAND: <VER> #Mainboard: Rev.01.00 #Firmware: Ver.01.00 #S/N: 01234567890123 #OK
<FWINFO>	Get extended device firmware information. Example: #COMMAND: <FWINFO> #Firmware: Ver.01.00 build 03 (12.05.2014 14:56) #OK
<IPC>	Get device network configuration. Example: #COMMAND: <IPC> #MACADDR: EA:E7:72:B3:0B:31 #IP: 10.1.1.52 #MASK: 255.255.255.0 #GW: 10.1.1.1 #DNS: 10.1.1.1 #OK
<GCS>	Get current device state (On or Standby). Examples: #COMMAND: <GCS>

	#RET: on #OK or #COMMAND: <GCS> #RET: off #OK
<GDM>	Get current HDMI video display mode. Possible decimal codes for HDMI video display mode are: 0 : 480i 1 : 576i 2 : 480p 3 : 576p 4 : 720p 50Hz 5 : 720p 60Hz 6 : 1080i 50Hz 7 : 1080i 60Hz 8 : 1080p 24Hz 9 : 1080p 50Hz 10 : 1080p 60Hz 11 : 1080p 25Hz 12 : 1080p 30HZ Example: #COMMAND: <GDM> #RET: 9;1080p 50Hz #OK
<SELECTED>	Get selected Tuner. 0 – Tuner 1 1 – Tuner 2 -1 – No tuner selected Example: #COMMAND: <SELECTED> #RET: 0 #OK
<GCV>	Get current volume (mute state and volume level). Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected

	Example: #COMMAND: <GCV> #RET: on;100 #OK
<VOL n>	Set/change volume (mute state or volume level). n = [+/-][0 .. 100] | [ON,OFF] n set to ON and n set to OFF turns mute on or off. n without a leading sign sets the volume absolute. n with a leading sign sets the volume relative to the current value. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Example: #COMMAND: <VOL -10> #RET: on;90 #OK
<MODE>	Get current device mode (1 = IDLE, 2 = FM, 3 = DAB). Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Example: #COMMAND: <MODE> #RET: 2 #OK
<DAB SEARCH S>	Get DAB channel search state (0 = not searching, 1 = searching). Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode

	Example: #COMMAND: <DAB SEARCH S> #RET: 0 #OK or #COMMAND: <DAB SEARCH S> #RET: 1 #OK
<DAB SERVICE C>	Get DAB channel / service count. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Example: #COMMAND: <DAB SERVICE C> #RET: 13 #OK
<DAB SERVICE I n>	Get DAB channel / service information. n = DAB channel index Return values are separated with ";" delimiter in format: *channel_index;service_index;pty;name* where *channel_index* = index of DAB channel *service_index* = index of DAB service *pty* = program type *name* = DAB channel / service name Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Example: #COMMAND: <DAB SERVICE I 3> #RET: 3;53792;7;DKultur #OK

<DAB SERVICE L>	List all DAB channels. Returns each channel in a separate line and values are separated with “;” delimiter in format: *channel_index;service_index;pty;name* where *channel_index* = index of DAB channel *service_index* = index of DAB service *pty* = program type *name* = DAB channel / service name Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Examples: #COMMAND: <DAB SERVICE L> #RET: 1;6138;10;Absolut relax #RET: 2;53776;3;Deutschlandfunk #RET: 3;53792;7;DKultur #OK or #COMMAND: <DAB SERVICE L> #ERR0R: Channel list is empty
<DAB TUNE I n>	Tune DAB Radio channel index. n = DAB channel index Return values are separated with “;” delimiter in format: *channel_index;return* where *channel_index* = index of DAB channel *return* = result (0 = successfully executed) Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode

	Example: #COMMAND: <DAB TUNE I 4> #RET: 4;0 #OK
<DAB STATE>	Get DAB channel and tune state. Return values are separated with ";" delimiter in format: *freq_idx;rssi;snr;valid;acq;fic;cnr;eid;chan_idx;bitrate;mode;return* where *freq_idx* = DAB channel frequency index *rssi* = RSSI value *snr* = SNR value *valid* = is channel valid *acq* = ACQ value *fic* = FIC Quality value *cnr* = CNR value *eid* = EID value *chan_idx* = DAB channel index *bitrate* = bitrate in kbps *mode* = sound mode (0 - Dual, 1 - Mono, 2 - Stereo, 3 - Joint Stereo) *return* = result (0 = successfully executed) Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Example: #COMMAND: <DAB STATE> #RET: 2;18;0;1;1;100;7;4284;5;62464;1;0 #OK
<DAB NAME>	Get name of current playing DAB channel. Return values are separated with ";" delimiter in format: *return;name*

	where *return* = result (0 = successfully executed) *name* = DAB channel name Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Example: #COMMAND: <DAB NAME> #RET: 0;ENERGY #OK
<DAB DLS>	Get DLS text of current playing DAB channel. Return values are separated with ";" delimiter in format: *return;length;text* where *return* = result (0 = successfully executed) *length* = text length *text* = DLS text Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Examples: #COMMAND: <DAB DLS> #RET: 0;38;Sie hoeren DRadio Dokumente & Debatten #OK or #COMMAND: <DAB DLS> #RET: 32;0; #OK

<DAB TIME>	Get current date and time from DAB tune. Return values are separated with ";" delimiter in format: *return;date_time* where *return* = result (0 = successfully executed) *date_time* = date and time information Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in DAB Mode. Otherwise command fails with the following error message: #ERROR: Not in DAB mode Examples: #COMMAND: <DAB TIME> #RET: 0;16.06.2015 09:02:03 #OK or #COMMAND: <DAB TIME> #RET: 32;00.00.0000 00:00:00 #OK
<FM SEARCH S>	Get FM frequency search state in % (-1 = not searching). Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM SEARCH S> #RET: 53 #OK or #COMMAND: <FM SEARCH S> #RET: -1 #OK

<FM FREQ A n>	Add frequency to list. n = list of atributes In this case is the list of attributes separated with ";" delimiter in format: *freq;name* where *freq* = FM frequency in 10kHz (102.70 MHz = 10270) *name* = FM frequency name (max. length 9) Return values are separated with ";" delimiter in format: *idx;freq;name* where *idx* = index of FM frequency *freq* = FM frequency in 10 kHz (10080 = 100.80 MHz) *name* = FM frequency name Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Examples: #COMMAND: <FM FREQ A 10910;Live News> #RET: 16;10910;Live News #OK or #COMMAND: <FM FREQ A 10910;Freq Name> #ERROR: frequency already in list or #COMMAND: <FM FREQ A 10310;OverflowExample> #RET: 17;10310;OverflowE #OK
<FM FREQ C>	Get FM frequency index count. Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected

	Example: #COMMAND: <FM FREQ C> #RET: 13 #OK
<FM FREQ I n>	Get FM frequency index information. n = FM frequency index Return values are separated with ";" delimiter in format: *idx;freq;pty;name* where *idx* = index of FM frequency *freq* = FM frequency in 10 kHz (10080 = 100.80 MHz) *pty* = program type *name* = FM frequency name Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <DAB SERVICE I 4> #RET: 4;10080;10;WDR 2 #OK
<FM FREQ L>	List all FM frequencies. Returns each frequency in a separate line and values are separated with ";" delimiter in format: *freq_index;freq;pty;name* where *freq_index* = FM frequency index *freq* = FM frequency in 10 kHz (8930 = 89.30 MHz) *pty* = program type *name* = FM frequency name Command should be used only with selected tuner. Otherwise command fails with the following error message:

	#ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Examples: #COMMAND: <FM FREQ L> #RET: 1;10190;3;WDR 5 #RET: 2;10240;10; 1LIVE #RET: 3;10580;0; ERFT #OK or #COMMAND: <DAB FREQ L> #ERR0R: Frequency list is empty
<FM TUNE I n>	Tune FM Radio frequency index. n = FM frequency index Return values are separated with ";" delimiter in format: *freq_index;freq;return* where freq_*index* = FM frequency freq = FM frequency in 10 kHz (10080 = 100.80 MHz) *return* = result (0 = successfully executed) Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM TUNE I 4> #RET: 4;10190;0 #OK
<FM STATE>	Get FM frequency and tune state. Return values are separated in two lines and with ";" delimiter in format:

	1. line (tune information): *return;valid;freq;rssi;snr;mpath* where *return* = result (0 = successfully executed) *valid* = is channel valid *freq* = FM frequency in 10 kHz (9040 = 90.40 MHz) *rssi* = RSSI value *snr* = SNR value *mpath* = multipath value 2. line (rds information): *return;rds_status;rds_pi;rds_pty* where *return* = result (0 = successfully executed) *rds_status* = RDS status *rds_pi* = RDS PI value *rds_pty* = RDS pogramm type information Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Examples: #COMMAND: <FM STATE> #RET: 0;TRUE;10240;28;28;3 #RET: 0;31;0xd391;10 #OK or #COMMAND: <FM STATE> #RET: 0;FALSE;10640;249;248;52 #RET: 0;31;0x0000;0 #OK
<FM NAME>	Get name of current playing FM frequency. Return values are separated with ";" delimiter in format: *return;name*

	where *return* = result (0 = successfully executed) *name* = FM frequency name Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM NAME> #RET: 0;1LIVE #OK
<FM RT>	Get RT text of current playing FM frequency. Return values are separated with ";" delimiter in format: *return;length;text* where *return* = result (0 = successfully executed) *length* = text length *text* = RT text (is filled to *length* characters with whitespaces) Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Example: #COMMAND: <FM RT> #RET: 0;64;Katy Perry mit Roar #OK or #COMMAND: <FM RT> #RET: 0;0; #OK
<FM TIME>	Get current date and time from FM tune. Return values are separated with ";" delimiter in format: *return;date_time*

	where *return* = result (0 = successfully executed) *date_time* = date and time information Command should be used only with selected tuner. Otherwise command fails with the following error message: #ERROR: No tuner selected Command should be used only in FM Mode. Otherwise command fails with the following error message: #ERROR: Not in FM mode Examples: #COMMAND: <FM TIME> #RET: 0;Tue 16.06.2015 10:02 #OK or #COMMAND: <FM TIME> #RET: 32; 00.00.0000 00:00 #OK
<UPDATE n s>	Update device firmware. n = [NET | SDC | USB] - update container (firmware image container) s = list of parameters depending of n (see below) Possible update containers are: 1. n = NET Update from network share (file server). In this case '*s'* is the list of string attributes separated with ";" delimiter in format: *image_file;server_ip;share_name;username;password* where *image_file* = relative path to firmware image file inside shared folder; *server_ip* = IP-address of the network file server; *share_name* = network shared resource name; *username* = network file server user name or "-" if empty for anonymous (guest) login; *password* = network file server user password or "-" if empty (no password);

	2. n = SDC Update from SD card. In this case *'s'* is relative path to firmware image file inside the SD card root filesystem. 3. n = USB Update from USD drive. In this case *'s'* is relative path to firmware image file inside the USB drive root filesystem. The operation progress can also be observed during the update process. Device sends progress information as: #RET: *progress_percent* Device also sends error message if the update process fails for some reason: #ERROR: *error_description* Examples: To update from network share #COMMAND: <UPDATE NET elium_radio_v01.00.img;10.1.1.5;Public;-;-> #RET: 20% #RET: 40% #RET: 60% #RET: 80% #RET: 100% #OK To update from SD card #COMMAND: <UPDATE SDC elium_radio_v01.00.img> #RET: 20% … #RET: 80% #RET: 100% #OK To update from USB drive #COMMAND: <UPDATE USB elium_radio_v01.00.img> … #OK