# Rotel RT-11 V02
# RS232 ASCII Controller Command List

| Date | Version | Update Description |
|---|---|---|
| June 12, 2014 | 1.00 | Original Specification |
| August 14. 2015 | 1.10 | Updated with several new commands added with software V2.24 (New commands shown in red) |
| August 28, 2018 | 1.11 | Add missing "!" char to recall preset commands. |

The RT-11 V02 supports an ASCII based RS232 protocol. The RS232 hardware does not support flow control so care needs to be take when sending and receiving data to avoid packet loss.

The below protocol is effective starting with V02 units that have the rear panel RS232 connection. Earlier models with the rear panel mini USB connections do not support this protocol.

All commands sent to the attached Rotel device must have a terminating "!" character.

**Example Command:** power_on!

***Note:*** *Do not include any spaces in the command, and do not include a carriage return or line feed after the command, only the "!" terminating character.*

Status information from the attached Rotel product with either have a terminating "!" character or a byte count for variable length text data that may include a "!" in the returned message. It is up to the sending/receiving control application to properly parse and process the packets.

***Note:*** *The byte count only includes the text data and not the length or "," character.*

## Connection Settings

| Baud Rate | Parity | Valid Data Bits | Stop Bit Value | Handshaking | Data Type |
|---|---|---|---|---|---|
| 115200 | N | 8 | 1 | None | String |

## Communication Protocol

Command and response messages are included on the following pages. Automatic display update information can be enabled/disabled using the "display_update_auto" and "display_update_manual" commands.

In automatic mode each time the display changes the new display line(s) will be sent.

In manual mode the display updates must be requested each time a refresh of the display information is desired. This primarily applies to Front USB metadata information. Basic status information such as volume, power or source changes will still be provided automatically.

## Section 1: Control Command List

| RT-11 ASCII | Command Description | Unit Response |
|---|---|---|
| POWER COMMANDS | | |
| power_on! | Power On | power=on! |
| power_off! | Power Off | power=standby! |
| power_toggle! | Power Toggle | power=on/standby! |
| SOURCE SELECTION COMMANDS | | |
| fm! | Source FM | source=fm! |
| dab! | Source DAB | source=dab! |
| TUNER CONTROL COMMANDS | | |
| tune_up! | Tune Up | n/a |
| tune_down! | Tune Down | n/a |
| queue! | Access Preset List | n/a |
| preset_up! | Preset Up | n/a |
| preset_down! | Preset Down | n/a |
| fm_mono! | Toggle FM Mono/Stereo Mode | n/a |
| fm_rds_disp! | Toggle through FM RDS or DAB Station Information | n/a |
| MENU CONTROL COMMANDS | | |
| menu! | Display the System Menu | n/a |
| module_setup! | Display the FM/DAB Setup Menu | n/a |
| exit! | Exit Key | n/a |
| up! | Cursor Up | n/a |
| down! | Cursor Down | n/a |
| left! | Cursor Left | n/a |
| right! | Cursor Right | n/a |
| enter! | Enter Key | n/a |
| NUMERIC KEY COMMANDS | | |
| 1! | Number Key 1 | n/a |
| 2! | Number Key 2 | n/a |
| 3! | Number Key 3 | n/a |
| 4! | Number Key 4 | n/a |
| 5! | Number Key 5 | n/a |
| 6! | Number Key 6 | n/a |
| 7! | Number Key 7 | n/a |
| 8! | Number Key 8 | n/a |
| 9! | Number Key 9 | n/a |
| 0! | Number Key 0 | n/a |

| RT-11 ASCII | Command Description | Unit Response |
|---|---|---|
| FM / DAB PRESET COMMANDS | | |
| memory! | Select memory for saving presets | n/a |
| call_fm_preset_n! | Recall FM Preset n (n = 01 - 30) | fm_preset_n=##,text |
| call_dab_preset_n! | Recall DAB Preset n (n = 01 - 30) | dab_preset_n=##,text |
| OTHER COMMANDS | | |
| dimmer! | Toggle display dimmer | dimmer_#! |
| dimmer_0! | Set display to brightest setting | n/a |
| dimmer_1! | Set display to dimmer level 1 | n/a |
| dimmer_2! | Set display to dimmer level 2 | n/a |
| dimmer_3! | Set display to dimmer level 3 | n/a |
| dimmer_4! | Set display to dimmer level 4 | n/a |
| dimmer_5! | Set display to dimmer level 5 | n/a |
| dimmer_6! | Set display to dimmest setting | n/a |
| DISPLAY REFRESH COMMANDS | | |
| display_update_auto! | Set Display Update to Auto | display_update=auto! |
| display_update_manual! | Set Display Update to Manual | display_update=manual! |

## Section 2: Feedback Request Command List

| | |
|---|---|
| **Command:** | get_display! |
| **Description:** | Request the entire display to be sent |
| **Return String:** | display=###,text |
| **Return Description:** | Current display data; must include 3 digit length of text string at beginning followed by "," and text string (no terminating character) |
| **Example:** | display=020, Sample Text |

| | |
|---|---|
| **Command:** | get_display1! |
| **Description:** | Request display line #1 to be sent |
| **Return String:** | display1=##,text |
| **Return Description:** | Current display line 1, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| **Example:** | display1=20, Sample Text |

| | |
|---|---|
| **Command:** | get_display2! |
| **Description:** | Request display line #2 to be sent |
| **Return String:** | display2=##,text |
| **Return Description:** | Current display line 2, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| **Example:** | display2=20, Sample Text |

| Command: | get_product_type! |
|---|---|
| Description: | Request the product type |
| Return String: | product_type=##,text |
| Return Description: | Rotel product type name, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | product_type=05,RT-11 |

| Command: | get_product_version! |
|---|---|
| Description: | Request the main CPU software version |
| Return String: | product_version=##,text |
| Return Description: | Rotel main CPU software version, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | product_version=06,V2.1.6 |

| Command: | get_display_size! |
|---|---|
| Description: | Request display size |
| Return String: | display_size=##,##! |
| Return Description: | Columns and rows on current display |
| Example: | display_size=20,02! |

| Command: | get_display_update! |
|---|---|
| Description: | Request display update |
| Return String(s): | display_update=auto! / display_update=manual! |
| Return Description: | Status of if the display refresh is automatic or manual |
| Example: | display_update=auto! |

| Command: | get_current_power! |
|---|---|
| Description: | Request current power status |
| Return String(s): | power=on! / power=standby! |
| Return Description: | Current power status |
| Example: | power=on! |

| Command: | get_current_source! |
|---|---|
| Description: | Request current source |
| Return String(s): | source=fm! / source=dab! |
| Return Description: | Current source |
| Example: | source=fm! |

| Command: | get_current_station! |
|---|---|
| Description: | Request current station |
| Return String(s): | **FM format:** current_station=modulated frequency [9 characters]+!<br>**DAB format:** current_station=station name[16 characters]+! |
| Return Description: | Current FM or DAB station |
| Example: | current_station= 99.55MHz! / current_station=6N 64-3_1KHZ_Mo! |

| Command: | get_current_preset! |
|---|---|
| Description: | Request current preset |
| Return String(s): | preset_fm=##! / preset_dab=##! |
| Return Description: | Current FM or DAB preset. If currently tuned station is not a stored preset will return 00 for preset number. |
| Example: | preset_fm=03! / preset_dab=01! / preset_dab=00! |

| Command: | get_current_dimmer! |
|---|---|
| Description: | Request current front display dimmer status |
| Return String(s): | dimmer=#! (0 (brightest) – 6 (dimmest)) |
| Return Description: | Current front display dimmer status |
| Example: | dimmer=3! |

| Command: | get_fm_preset_n! |
|---|---|
| Description: | Request saved FM station info for present n (n = 1 – 30) |
| Return String(s): | fm_preset_n=##,text |
| Return Description: | Saved FM preset number, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | fm_preset_04=09,107.15MHz |

| Command: | get_dab_preset_n! |
|---|---|
| Description: | Request saved DAB station info for present n (n = 1 – 30) |
| Return String(s): | dab_preset_n=##,text |
| Return Description: | Saved DAB preset number, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | dab_preset_01=##,text |

| Command: | get_allpreset_fm! |
|---|---|
| Description: | Request all saved FM station info [ 1..30 ] |
| Return String(s): | fm_preset_n=##,text |
| Return Description: | Saved FM preset number, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | fm_allpreset_01=##,text .. fm_allpreset_30=##,text |

| Command: | get_allpreset_dab! |
|---|---|
| Description: | Request all saved DAB station info [ 1..30 ] |
| Return String(s): | dab_preset_n=##,text |
| Return Description: | Saved DAB preset number, must include 2 digit length of text string at beginning followed by "," and text string (no terminating character) |
| Example: | dab_allpreset_01=##,text .. dab_allpreset_30=##,text |