|  |  |
| --- | --- |
| Logo_HB9HSLU.pnghslu_ta_elektrotechnik_cmyk_0902.jpghslu@space_25x140mm.gif | **Astrocast Transceiver and Rotator Remote Control Base  Specifications and Interface Description** |
| Prepared by:  Martin Klaper, HB9ARK |  |
| Checked/reviewed by: Florian George & Group, Astrocast Marcel Joss, HB9TWM |  |
| Approved by: Federico Belloni |  |
| Amateurfunkverein der Hochschule Luzern HB9HSLU  Technikumstrasse 21, CH-6048 Horw |  |

**RECORD OF REVISIONS**

|  |  |  |  |
| --- | --- | --- | --- |
| ISS/REV | Date | Modifications | Created/modified by |
| 0/8 | 10/08/2018 | Initial release | Martin Klaper |
| 0/9 | 12/08/2018 | captions, config ctl, context diagram architecture & context | Martin Klaper |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**DISTRIBUTION LIST**

|  |  |  |  |
| --- | --- | --- | --- |
| **CONFIDENTIAL** |  |  |  |
| NO |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **RECIPIENT NAME** | **FUNCTION** | **AFFILIATION** | **COMMENT** |
| Federico Belloni and group | CTO and Co-founder | Astrocast, EPFL Innovation Park | process owner |
| Florian George | Software Systems   Engineer | Astrocast, EPFL Innovation Park |  |
| Marcel Joss, HB9TWM | Professor | HSLU |  |
| Remo Mattmann | Assistent | HSLU | informational |
| Armin Roesch, HB9MFL | Reviewer & Swisscube Operator | AMSAT-DL | informational |
| Michael Lipp, HB9WDF | Swiss Oscar Operator | AMSAT-DL, UK, NA | informational |
|  |  |  |  |
|  |  |  |  |

TABLE OF CONTENTS

[**RECORD OF REVISIONS** 2](#_Toc521792566)

[**DISTRIBUTION LIST** 3](#_Toc521792567)

[TABLE OF CONTENTS 4](#_Toc521792568)

[**REFERENCES** 5](#_Toc521792569)

[**Applicable Documents** 5](#_Toc521792570)

[**References Documents** 5](#_Toc521792571)

[**LIST OF ACRONYMS** 7](#_Toc521792572)

[1. Introduction 8](#_Toc521792573)

[2. Functionality 9](#_Toc521792574)

[3. Commands and Parameters for transceivers 11](#_Toc521792575)

[4. Initial Setup parameters for transceiver 14](#_Toc521792576)

[5. Commands and Parameters for rotators 15](#_Toc521792577)

[6. Initial Setup parameters for rotators 16](#_Toc521792578)

[7. Configuration of Transceiver and Antennae Base 17](#_Toc521792579)

[1. Context and Architecture 19](#_Toc521792580)

[2. Appendix: Supported transceivers 21](#_Toc521792581)

[3. Appendix: Supported rotators 24](#_Toc521792582)

**REFERENCES**

**Applicable Documents**

[A1] Groundstation HB9HSLU-V2.0, Martin Klaper, 07-SEP-2017, Astrocast meeting in Horw

[A2] 73-Astrocast-HSLU-Ground Station Description-2-3, 12-MAR-2018, Martin Klaper

[A3] Agenda Astrocast Meeting 19-JUNE-2018, Ecublens

[A4] Minutes of the meeting, 19-JUNE-2018, Florian George

**References Documents**

|  |  |  |
| --- | --- | --- |
| [R1] | N. N. | Ham Radio Control Libraries, version 3.0.1, 6 January 2016 [[1]](#footnote-1) |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Typesetting conventions:**

* Text is written in Calibri font.
* Commands and responses are written in Courier New.

**List of figures**

[Fig 1 virtualization / abstraction of transceivers and rotators 19](#_Toc521792583)

[Fig 2 Context and architecture diagram 20](#_Toc521792584)

**List of tables**

[Table 1 Commands for reservation of equipment 9](#_Toc521792585)

[Table 2 Commands for selection of transceiver or rotator (device selector) 10](#_Toc521792586)

[Table 3 description of Commands for transceivers 13](#_Toc521792587)

[Table 4 description of Commands for rotators 15](#_Toc521792588)

[Table 5 Parameters for hamlib invocation 17](#_Toc521792589)

[Table 3 Supported transceivers 23](#_Toc521792590)

[Table 3 Supported rotators 24](#_Toc521792591)

**LIST OF ACRONYMS**

|  |  |
| --- | --- |
| AZ | Azimuth |
| CAT | computer aided tuning |
| EGSE (router) | Electrical Ground Support Equipment |
| EL | Elevation |
| HSLU | Hochschule Luzern – Technik & Architektur |
| I/F | Interface |
| rig | transceiver |
| rot | Rot rotator (AZ/EL) |
| RX | Receiver |
| SCOE | Special Checkout Equipment |
| S Band | 2.4 GHz Band |
| TRX | Transceiver, i. e. TX and RX |
| TX | Transmitter |
| UHF | Ultra High Frequency |
| VHF | Very High Frequency |
| VFO | Variable Frequency Oscillator |
|  |  |
|  |  |
|  |  |
|  |  |

# Introduction

This document describes the Astrocast’s "Remote Ground Station Base" (RGSB). The Remote Ground Station Base (RGSB) constitutes a virtual **transceiver** and a virtual **rotator** **controller** with interlocking facilities. The current implementation instantiates Command and Control for

* One VHF/UHF transceiver[[2]](#footnote-2)
* One VHF/UHF rotator controller AZ/EL
* One S-Band transceiver
* One S-Band rotator controller AZ/EL

The entity for the interlocking is either the VHF/UHF transceiver and VHF/UHF antenna rotator or the S-Band transceiver and S-Band antenna rotator as a compound unit. If the VHF/UHF transceiver consists of separate Uplink und Downlink Transceivers, both belong to the VHF/UHF entity.

This solution encapsulates transceivers and rotators by providing virtual transceivers and rotators.

The entries in the configuration file named “GroundStation.exe.config”, which can be altered without recompiling the project, select the concrete transceiver and antenna rotator. After altering a transceiver or a rotator, a restart of the program “GroundStation.exe” is necessary. A list in the appendix shows all available concrete transceivers and rotators.

The “GroundStationTransceiverController” serves as a simple test environment and permits activation of all commands on a command line. A graphical user interface is not part of this project.  
  
GroundStationTransceiverController and GroundStation communicate via an EGSE router.

This document summarizes all available commands and responses, which closely follow the Hamlib definition.

For test purposes a "GroundStation.exe" is running at HB9HSLU via the EPFL EGSE router.

# Functionality

Full remote control via the Electrical Ground Support Router (EGSE) infrastructure of up to three transceivers and up to two AZ/EL antenna rotators is required. This control software supports all common brands of ham radio transceivers and antenna controllers. The caller gets a response to issued commands. Depending on the type of radio, the scope of Computer Aided Tuning (CAT) is different. [[3]](#footnote-3) The aim is to control all commands and responses that are possible for each specific type of radio. Changing a radio or a radio interface (RS-232, USB) requires a restart of the program at the premises of the ground station.

**Commands for Locking / Unlocking (reservation of equipment)**Relating to the interlock concept there are two units, a VHF/UHF unit and an S-Band unit. Each of the unit is in one of the states*free* (green) or *occupied* (red). It is therefore possible to operate VHF/UHF and S-Band independently, e.g. for student's project work. The current state is graphically displayed at the groundstation. Remote station can check the current state. It is therefore necessary to request (and release) access to the equipment.

**Commands to get access to the equipment (reservation of equipment)** [[4]](#footnote-4)

|  |
| --- |
| * "***requestVHFUHF***" |
| * "***releaseVFUHF***" |
| * "***requestSband***" |
| * "***releaseSband***" |
| * "***getReservationState***" |

Table 1 Commands for reservation of equipment

**Commands to select devices (Selector for transceivers, rotators)**

|  |
| --- |
| * ***"rigctlVHFUHF01:<command to transceiver[[5]](#footnote-5)>"*** |
| * ***"rigctlVHFUHF02:<command to transceiver>"*** |
| * ***"rigctlSband:<command to transceiver>"*** |
| * ***"rotctlVHFUHF:<command to rotator>"*** |
| * ***"rotctlS-Band:<command to rotator>"*** |

Table 2 Commands for selection of transceiver or rotator (device selector)

# Commands and Parameters for transceivers

Commands and parameters are according to the following definition.

* Command short name (usually a single letter) is followed by the long name which is followed by any variable names (parameters).
* Some short commands are noted as hexadecimal digits due to the limitation of upper and lower case letters available. Use the associated long command name instead.
* While a comma is used to separate variable names in this document, they are not part of the command syntax used by rigctl. Use a space to separate values.
* In the case of "set" commands the variable name is replaced by the value in the description.
* Commands in **green** colour are often used basic commands.
* This list contains only a selection of commands in order to keep this description short. The HAMLIB documentation lists all commands.
* **Capitalized letters are set commands. Lower case letters are get commands.**

|  |  |
| --- | --- |
| *Short name, long name parameter(s)* | comment |
| ***F, set\_freq Frequency*** | **Set Frequency, in Hertz.** |
| ***f, get\_freq*** | **Get Frequency, in Hertz.** |

|  |  |
| --- | --- |
| ***M, set\_mode Mode, Passband*** | * Set Mode to one of: USB, LSB, CW, CWR, RTTY, RTTYR, AM, FM, WFM, … * Set Passband frequency in Hertz, or 0 for the Hamlib backend default * Passing a "?" (query) as the first argument instead of Mode will return a space separated list of radio backend supported Modes. Use this to determine the supported Modes of a given radio backend. |
| ***m, get\_mode*** | Returns Mode as a string from set\_mode above and Passband frequency in Hertz. |

|  |  |
| --- | --- |
| **V, set\_vfo VFO** | Set VFO to one of: VFOA, VFOB, .... |
| **v, get\_vfo** | Get current VFO. |

|  |  |
| --- | --- |
| **T, set\_ptt PTT** | **Set PTT to one of: 0 (RX), 1 (TX), 2 (TX mic), 3 (TX data).** |
| **t, get\_ptt** | **Get PTT status.** |

|  |  |
| --- | --- |
| **I, set\_split\_freq Tx Frequency** | **Set TX Frequency, in Hertz for “split” frequency operation.** |
| **i, get\_split\_freq** | **Get TX Frequency, in Hertz for “split” frequency operation.** |

|  |  |
| --- | --- |
| N, set\_ts Tuning Step | Set Tuning Step, in Hertz. |
| n, get\_ts | Get Tuning Step, in Hertz. |

|  |  |
| --- | --- |
| U, set\_func Func, Func Status | Set Func, Func Status.   * Func is one of: FAGC, NB, NR, AFC, SATMODE, … * Func Status argument is 1 for "activate", 0 for "deactivate. * Passing a "?" (query) as the first argument instead of Func will return a space separated list of radio backend supported "set" functions. Use this to determine the supported functions of a given radio backend. |
| u, get\_func Func | Get Func Status. |

|  |  |
| --- | --- |
| L, set\_level Level, Level Value | Set Level, Level Value.  Level is one of: PREAMP, ATT, AF, RF, SQL, NR, RFPOWER, MICGAIN, AGC(0:OFF, 1:SUPERFAST, 2:FAST, 3:SLOW, 4:USER, 5:MEDIUM, 6:AUTO), SWR, ALC, …. |
| l, get\_level Level | Get Level Value. The Level Value can be a float or an integer. |

|  |  |
| --- | --- |
| P, set\_parm Parm, Parm Value | Set Parm, Parm Value   * Parm is one of: ANN, **APO**, BACKLIGHT, BEEP, …  Passing a "?" (query) as the first argument instead of Parm will return a space separated list of radio backend supported “set” parameters. Use this to determine the supported parameters of a given radio backend. |
| p, get\_parm Parm | Get Parm Value.  Returns Parm Value as a float or integer for the Parm passed. Parm is a token from the list in set\_parm above |

|  |  |
| --- | --- |
| **\*, reset Reset** | **Perform rig Reset.**  **0 = None, 1 = Software reset, 2 = VFO reset, 4 = Memory Clear reset, 8 = Master reset. Since these values are defined as a bitmask in rig.h, it should be possible to AND these values together to do multiple resets at once, if the backend supports it or supports a reset action via rig control at all.** |

|  |  |
| --- | --- |
| 0x87, set\_powerstat *Power Status* | Set power On/Off/Standby *Power Status*. 0 = Power Off, 1 = Power On, 2 = Power Standby. |
| 0x88, get\_powerstat | Get power On/Off/Standby *Power Status* as in set\_powerstat above. |

|  |  |
| --- | --- |
| \_, get\_info \_ is underscore | Get misc information about the rig |

|  |  |
| --- | --- |
| 1, dump\_caps | Not a real rig remote command, it just dumps capabilities, i.e. what the backend knows about this model, and what it can do.  This command will produce many lines of output so be very careful if using a fixed length array! For example, running this command against the Dummy backend results in over 5 kB of text output. |

Table 3 description of Commands for transceivers

# Initial Setup parameters for transceiver

Caution: This suggestion is tentative and is subject to verification. [[6]](#footnote-6)

|  |  |
| --- | --- |
| **Cold start procedure** | **Use of transceiver** |
| 1. Set power on 2. Reset all 3. Set VFO A 4. Set Mode to USB 2400 Hz 5. Set tuning step 10 Hz 6. Set Frequency 438abc MHz 7. Set VFO B (TRX 01) 8. Set Mode to USB 2400 Hz 9. Set tuning step 10 Hz 10. Set Frequency 145xyz MHz | 1. Use Set Frequency to adjust for Doppler 2. Use get Frequency (maybe) 3. Use Set PTT [Data=3] (Push to Talk) for transmission. |

These commands correspond to the above definitions and run with the virtual transceiver (m=1).

|  |  |
| --- | --- |
| **Cold start procedure, coded** | **Use of transceiver, coded** |
| 1. requestVHFUHF 2. rigctlVHFUHF1:set\_powerstat 3. rigctlVHFUHF1:\* 15 4. rigctlVHFUHF1:-V VFOA 5. rigctlVHFUHF1:-M USB 2400 6. rigctlVHFUHF1:-N 10 7. rigctlVHFUHF1:-F 438123456 8. rigctlVHFUHF1:-V VFOB 9. rigctlVHFUHF1:-M USB 2400 10. rigctlVHFUHF1:-N 10 11. rigctlVHFUHF1:-F 145876543 | 1. rigctlVHFUHF1:-F 438123987 2. rigctlVHFUHF1:-f 3. rigctlVHFUHF1:-T 3      1. releaseVHFUHF |

# Commands and Parameters for rotators

Commands and parameters are according to the following definition.

* Command short name is followed by the long name which is followed by any variable names.
* While a comma is used to separate variable names in this document, they are not part of the command syntax used by rotctl. Use a space to separate values.
* In the case of "set" commands the variable name is replaced by the value in the description.
* In the case of "get" commands the variable name is the key name of the value returned.
* Commands in green colour are often used basic commands.
* This list contains only a selection of commands in order to keep this description short. The HAMLIB documentation lists all commands.
* **Capitalized letters are setters. Lower case letters are getters.**

|  |  |
| --- | --- |
| P, set\_pos Azimuth, Elevation | Set position: Azimuth and Elevation, double precision floating |
| p, get\_pos | Get position: Azimuth and Elevation double precision floating |

|  |  |
| --- | --- |
| M, move Direction, Speed | * Move the rotator in a specific direction at the given rate. * Values are integers where Direction is defined as 2 = Up, 4 = Down, 8 = Left, and 16 = Right. Speed is an integer between 1 and 100. * Not all backends that implement the move command use the Speed [[7]](#footnote-7)value. |

|  |  |
| --- | --- |
| S, stop | Stop the rotator. |
| K, park | Park the antenna. |
| R, reset Reset | Reset the rotator. Integer value of 1 for Reset All. |
| \_, get\_info // \_ is underscore | Get misc information on the rotator. returns Model Name |
| w, send\_cmd Cmd | Send raw command string to the rotator. |

Table 4 description of Commands for rotators

# Initial Setup parameters for rotators

Caution: This suggestion is tentative and is subject to verification. [[8]](#footnote-8)

|  |  |
| --- | --- |
| **Cold start procedure** | **Use of rotator(s)** |
| 1. Reset 2. Park | 1. Set Position AZ / EL |

These commands correspond to the above definitions and run with the virtual rotator (m=1).

|  |  |
| --- | --- |
| **Cold start procedure, coded** | **Use of rotator(s), coded** |
| 1. requestVHFUHF // unless already done 2. rotctlVHFUHF:-R 3. rotctlVHFUHF:-K | 1. rotctlVHFUHF:-P 303 45 2. rotctlVHFUHF:-P 304 47 3. rotctlVHFUHF:-P 305 49 4. and so on ……. |

# Configuration of Transceiver and Antennae Base

* These parameters are relevant only for ground station clients, i. e. for the location where the physical antennas, transceivers and rotators are located.
* It deals with the physical connections of transceivers and rotators
* It deals with the concrete transceiver type.
* For a complete list see the HAMLIB documentation.

|  |  |  |
| --- | --- | --- |
| **Rig control** | | |
| **Short form** | **Long form** | **description** |
| -m | --model=id | Select radio model number. See model list (use rigctld –l) |
| -r | --rig-file=device | Often a serial port, but could be a USB to serial adapter. Typically /dev/ttyS0 or COM1 , COM2 , … |
| -s | --serial-speed=baud | Set serial speed to *baud* rate |
| -T | --listen-addr=IPADDR | Use *IPADDR* as the listening IP address / localhost |
| -t | --port=number | recommendation: even numbers for rig |

|  |  |  |
| --- | --- | --- |
| **Rot control** | | |
| **Short form** | **Long form** | **description** |
| -m | --model=id | Select rotator model number. |
| -r | --rig-file=device | Often a serial port, but could be a USB to serial adapter. Typically /dev/ttyS0 or COM1 , COM2 , … |
| -s | --serial-speed=baud | Set serial speed to *baud* rate |
| -T | --listen-addr=IPADDR | Use *IPADDR* as the listening IP address / localhost |
| -t | --port=number | recommendation: odd numbers for rot |

Table 5 Parameters for hamlib invocation

**Some examples of HAMLIB invocation parameters:**

HamlibInvocationParametersTRX1 = "-m 1 -t 4534" // dummy transceiver

HamlibInvocationParametersTRX2dummy = "-m 1 -t 4536"

HamlibInvocationParametersTRX2   
 = "-m 214 -r COM7 -s 9600 -C rts\_state=ON -t 4536" // TS-2000

HamlibInvocationParametersTRX\_Sband\_dummy = "-m 1 -t 4538"

HamlibInvocationParametersTRX\_Sband   
 = "-m 214 -r COM99 -s 9600 -C rts\_state=ON -t 4538"

HamlibInvocationParametersROT99 = "-m 1 -t 4535" // dummy rotator

HamlibInvocationParametersROT1 = "–m 901 –r COM4 –s 115200 -t 4535" // VHFUHF rotator

HamlibInvocationParametersROT\_Sband\_dummy = "–m 1 -t 4537"

HamlibInvocationParametersROT\_Sband = "–m 901 –r COMx –s 115200 -t 4537"

|  |  |  |
| --- | --- | --- |
|  | port number | COM port |
| VHFUHF transceiver 1 | 4534 | COM7 |
| VHFUHF transceiver 2 | 4536 | tbd |
| VHFUHF rotators AZ/EL | 4535 | COM4 |
| S Band transceiver [[9]](#footnote-9) | 4538 | tbd |
| S Band rotator (dish) AZ/EL | 4537 | tbd |

# Context and Architecture

HAMLIB offers an open source library for control of transceivers and rotators of almost any brand. The HAMLIB Core serves for transceivers (rigctl) or for rotators (rotctl) and is the base for remote controlling e. g. ground stations.

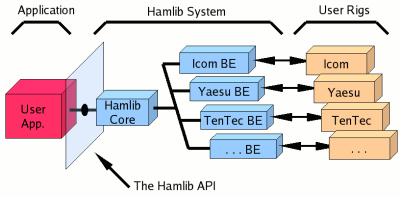


Fig 1 virtualization / abstraction of transceivers and rotators

COM 7

VHF/UHF TRX 1  
e. g. TS-2000

VHF/UHF TRX 2  
in case of separate   
TX & RX

Yagi Rotator VHF/UHF

S-Band TRX  
GNU/USRP

Dish Rotator  
S-Band

Command dispatcher  
*Groundstation.exe*

COM x

COM 4

COM 1

EGSE router

EGSE port 68

simple test environment  
*RadioTestingClient.exe*

EGSE port 64

**I/F ground station**

Fig 2 Context and architecture diagram

# Appendix: Supported transceivers

|  |  |  |
| --- | --- | --- |
| Rig # | Manufacturer | Model |
| **1** | **Hamlib** | **Dummy** |
| 2 | Hamlib | NET rigctl |
| 101 | Yaesu | FT-847 |
| 103 | Yaesu | FT-1000D |
| 104 | Yaesu | MARK-V FT-1000MP |
| 105 | Yaesu | FT-747GX |
| 106 | Yaesu | FT-757GX |
| 107 | Yaesu | FT-757GXII |
| 109 | Yaesu | FT-767GX |
| 110 | Yaesu | FT-736R |
| 111 | Yaesu | FT-840 |
| 113 | Yaesu | FT-900 |
| 114 | Yaesu | FT-920 |
| 115 | Yaesu | FT-890 |
| 116 | Yaesu | FT-990 |
| 117 | Yaesu | FRG-100 |
| 118 | Yaesu | FRG-9600 |
| 119 | Yaesu | FRG-8800 |
| 120 | Yaesu | FT-817 |
| 121 | Yaesu | FT-100 |
| 122 | Yaesu | FT-857 |
| 123 | Yaesu | FT-897 |
| 124 | Yaesu | FT-1000MP |
| 125 | Yaesu | MARK-V Field FT-1000MP |
| 126 | Yaesu | VR-5000 |
| 127 | Yaesu | FT-450 |
| 128 | Yaesu | FT-950 |
| 129 | Yaesu | FT-2000 |
| 130 | Yaesu | FTDX-9000 |
| 131 | Yaesu | FT-980 |
| 132 | Yaesu | FT-DX5000 |
| 133 | Vertex Standard | VX-1700 |
| 134 | Yaesu | FT-1200 |
| 135 | Yaesu | FT-991 |
| 201 | Kenwood | TS-50S |
| 202 | Kenwood | TS-440 |
| 203 | Kenwood | TS-450S |
| 204 | Kenwood | TS-570D |
| 205 | Kenwood | TS-690S |
| 206 | Kenwood | TS-711 |
| 207 | Kenwood | TS-790 |
| 208 | Kenwood | TS-811 |
| 209 | Kenwood | TS-850 |
| 210 | Kenwood | TS-870S |
| 211 | Kenwood | TS-940S |
| 213 | Kenwood | TS-950SDX |
| **214** | **Kenwood** | **TS-2000** |
| 215 | Kenwood | R-5000 |
| 216 | Kenwood | TS-570S |
| 217 | Kenwood | TH-D7A |
| 219 | Kenwood | TH-F6A |
| 220 | Kenwood | TH-F7E |
| 221 | Elecraft | K2 |
| 222 | Kenwood | TS-930 |
| 223 | Kenwood | TH-G71 |
| 224 | Kenwood | TS-680S |
| 225 | Kenwood | TS-140S |
| 226 | Kenwood | TM-D700 |
| 227 | Kenwood | TM-V7 |
| 228 | Kenwood | TS-480 |
| 229 | Elecraft | K3/KX3 |
| 230 | Kenwood | TRC-80 |
| 231 | Kenwood | TS-590S |
| 232 | SigFox | Transfox |
| 233 | Kenwood | TH-D72A |
| 234 | Kenwood | TM-D710 |
| 236 | FlexRadio | 6xxx |
| 237 | Kenwood | TS-590SG |
| 238 | Elecraft | XG3 |
| 239 | Kenwood | TS-990s |
| 302 | Icom | IC-1275 |
| 303 | Icom | IC-271 |
| 304 | Icom | IC-275 |
| 306 | Icom | IC-471 |
| 307 | Icom | IC-475 |
| 309 | Icom | IC-706 |
| 310 | Icom | IC-706MkII |
| 311 | Icom | IC-706MkIIG |
| 312 | Icom | IC-707 |
| 313 | Icom | IC-718 |
| 314 | Icom | IC-725 |
| 315 | Icom | IC-726 |
| 316 | Icom | IC-728 |
| 319 | Icom | IC-735 |
| 320 | Icom | IC-736 |
| 321 | Icom | IC-737 |
| 322 | Icom | IC-738 |
| 323 | Icom | IC-746 |
| 324 | Icom | IC-751 |
| 326 | Icom | IC-756 |
| 327 | Icom | IC-756PRO |
| 328 | Icom | IC-761 |
| 329 | Icom | IC-765 |
| 330 | Icom | IC-775 |
| 331 | Icom | IC-781 |
| 332 | Icom | IC-820H |
| 334 | Icom | IC-821H |
| 335 | Icom | IC-970 |
| 336 | Icom | IC-R10 |
| 337 | Icom | IC-R71 |
| 338 | Icom | IC-R72 |
| 339 | Icom | IC-R75 |
| 340 | Icom | IC-R7000 |
| 341 | Icom | IC-R7100 |
| 342 | Icom | ICR-8500 |
| 343 | Icom | IC-R9000 |
| 344 | Icom | IC-910 |
| 345 | Icom | IC-78 |
| 346 | Icom | IC-746PRO |
| 347 | Icom | IC-756PROII |
| 351 | Ten-Tec | Omni VI Plus |
| 352 | Optoelectronics | OptoScan535 |
| 353 | Optoelectronics | OptoScan456 |
| 354 | Icom | IC ID-1 |
| 355 | Icom | IC-703 |
| 356 | Icom | IC-7800 |
| 357 | Icom | IC-756PROIII |
| 358 | Icom | IC-R20 |
| 360 | Icom | IC-7000 |
| 361 | Icom | IC-7200 |
| 362 | Icom | IC-7700 |
| 363 | Icom | IC-7600 |
| 364 | Ten-Tec | Delta II |
| 365 | Icom | IC-92D |
| 366 | Icom | IC-R9500 |
| 367 | Icom | IC-7410 |
| 368 | Icom | IC-9100 |
| 369 | Icom | IC-RX7 |
| 370 | Icom | IC-7100 |
| 371 | Icom | ID-5100 |
| 372 | Icom | IC-2730 |
| 373 | Icom | IC-7300 |
| 374 | Microtelecom | Perseus |
| 401 | Icom | IC-PCR1000 |
| 402 | Icom | IC-PCR100 |
| 403 | Icom | IC-PCR1500 |
| 404 | Icom | IC-PCR2500 |
| 501 | AOR | AR8200 |
| 502 | AOR | AR8000 |
| 503 | AOR | AR7030 |
| 504 | AOR | AR5000 |
| 505 | AOR | AR3030 |
| 506 | AOR | AR3000A |
| 508 | AOR | AR2700 |
| 513 | AOR | AR8600 |
| 514 | AOR | AR5000A |
| 515 | AOR | AR7030 Plus |
| 516 | AOR | SR2200 |
| 605 | JRC | NRD-525 |
| 606 | JRC | NRD-535D |
| 607 | JRC | NRD-545 DSP |
| 801 | Uniden | BC780xlt |
| 802 | Uniden | BC245xlt |
| 803 | Uniden | BC895xlt |
| 804 | Radio Shack | PRO-2052 |
| 806 | Uniden | BC250D |
| 810 | Uniden | BCD-396T |
| 811 | Uniden | BCD-996T |
| 812 | Uniden | BC898T |
| 902 | Drake | R-8A |
| 903 | Drake | R-8B |
| 1004 | Lowe | HF-235 |
| 1103 | Racal | RA6790/GM |
| 1105 | Racal | RA3702 |
| 1204 | Watkins-Johnson | WJ-8888 |
| 1402 | Skanti | TRP8000 |
| 1404 | Skanti | TRP 8255 S R |
| 1509 | Winradio | WR-G313 |
| 1601 | Ten-Tec | TT-550 |
| 1602 | Ten-Tec | TT-538 Jupiter |
| 1603 | Ten-Tec | RX-320 |
| 1604 | Ten-Tec | RX-340 |
| 1605 | Ten-Tec | RX-350 |
| 1607 | Ten-Tec | TT-516 Argonaut V |
| 1608 | Ten-Tec | TT-565 Orion |
| 1609 | Ten-Tec | TT-585 Paragon |
| 1611 | Ten-Tec | TT-588 Omni VII |
| 1612 | Ten-Tec | RX-331 |
| 1613 | Ten-Tec | TT-599 Eagle |
| 1701 | Alinco | DX-77 |
| 1801 | Kachina | 505DSP |
| 2201 | TAPR | DSP-10 |
| 2301 | Flex-radio | SDR-1000 |
| 2303 | DTTS Microwave Society | DttSP IPC |
| 2304 | DTTS Microwave Society | DttSP UDP |
| 2401 | RFT | EKD-500 |
| 2501 | Elektor | Elektor 3/04 |
| 2502 | SAT-Schneider | DRT1 |
| 2503 | Coding Technologies | Digital World Traveller |
| 2506 | AmQRP | DDS-60 |
| 2507 | Elektor | Elektor SDR-USB |
| 2508 | mRS | miniVNA |
| 2509 | SoftRock | Si570 AVR-USB |
| 2511 | KTH-SDR kit | Si570 PIC-USB |
| 2512 | FiFi | FiFi-SDR |
| 2513 | AMSAT-UK | FUNcube Dongle |
| 2514 | N2ADR | HiQSDR |
| 2515 | Funkamatuer | FA-SDR |
| 2516 | AE9RB | Si570 Peaberry V1 |
| 2517 | AE9RB | Si570 Peaberry V2 |
| **2518** | **AMSAT-UK** | **FUNcube Dongle Pro+** |
| 2701 | Rohde&Schwarz | ESMC |
| 2702 | Rohde&Schwarz | EB200 |
| 2801 | Philips/Simoco | PRM8060 |
| 2901 | ADAT www.adat.ch | ADT-200A |
| 3001 | Icom | IC-M700PRO |
| 3002 | Icom | IC-M802 |
| 3003 | Icom | IC-M710 |

Table 3 Supported transceivers

# Appendix: Supported rotators

|  |  |  |  |
| --- | --- | --- | --- |
| *Rig #* | Manufacturer | Model | Used at |
| 1 | Hamlib | Dummy |  |
| 2 | Hamlib | NET rotctl |  |
| 201 | Hamlib | EasycommI |  |
| 202 | Hamlib | EasycommII |  |
| 204 | Hamlib | EasycommIII |  |
| 301 | XQ2FOD | Fodtrack |  |
| 401 | Idiom Press | Rotor-EZ |  |
| 402 | Idiom Press | RotorCard |  |
| 403 | Hy-Gain | DCU-1/DCU-1X |  |
| 404 | DF9GR | ERC |  |
| 405 | Green Heron | RT-21 |  |
| 501 | SARtek | SARtek-1 |  |
| 601 | Yaesu | GS-232A |  |
| 602 | Yaesu/Kenpro | GS-232 |  |
| 603 | Yaesu | GS-232B |  |
| 604 | F1TE | GS232/F1TE Tracker |  |
| 701 | WA6UFQ | PcRotor |  |
| 801 | Heathkit | HD 1780 Intellirotor |  |
| **901** | **SPID** | **Rot2Prog** | **HB9HSLU** |
| 902 | SPID | Rot1Prog |  |
| 1001 | M2 | RC2800 |  |
| 1101 | EA4TX | ARS RCI AZ&EL |  |
| 1102 | EA4TX | ARS RCI AZ |  |
| 1201 | AMSAT | IF-100 |  |
| 1301 | LA7LKA | ts7400 |  |
| 1401 | Celestron | NexStar |  |
| 1501 | DG9OAA | Ether6 (via ethernet) |  |
| 1601 | CNCTRK | CNCTRK |  |
| 1701 | Prosistel | Prosistel D |  |

Table 3 Supported rotators

1. The [GNU Lesser General Public License](http://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) LGPL for the “front end” and “back end” library source code files, and the [GNU General Public License](http://en.wikipedia.org/wiki/GNU_General_Public_License) GPL for the supplied programs source code files apply. Extracts from the hamlib manual are covered by the [GNU Free Documentation License](http://en.wikipedia.org/wiki/GNU_Free_Documentation_License) GFDL. Our software makes use of the unaltered HAMLIB library. [↑](#footnote-ref-1)
2. The VHF/UHF transceiver can optionally use two distinct radio devices, one for the uplink and another one for the downlink. [↑](#footnote-ref-2)
3. The original HAMLIB documentation lists restrictions for some brands of equipment. There are no restrictions known for the Kenwood TS-2000 and many others brands. [↑](#footnote-ref-3)
4. All commands are case sensitive. [↑](#footnote-ref-4)
5. Commands and parameters: see next chapter [↑](#footnote-ref-5)
6. Depending on the reset state of the transceiver, additional setting may be necessary. [↑](#footnote-ref-6)
7. Probably unavailable for our rotator, but unchecked [↑](#footnote-ref-7)
8. Depending on the reset state of the transceiver, additional setting may be necessary. [↑](#footnote-ref-8)
9. It is desirable that the S-Band transceiver implements one of the common CAT protocols [↑](#footnote-ref-9)