RUTX08 USB Tools

The information in this page is updated in accordance with firmware version **RUTX_R_00.07.06.10**.

□

Contents

- 1 Summary
- 2 General
  - 2.1 USB Mount Settings
  - 2.2 Mounted File Systems
- 3 Printer Server
- 4 USB to Serial
  - 4.1 Console
  - 4.2 Over IP
  - 4.3 Modbus gateway
  - 4.4 NTRIP client
  - 4.5 IP Filter
- 5 Network Shares
  - 5.1 General Settings
  - 5.2 Shared Directories
  - 5.3 Users
- 6 DLNA
  - 6.1 General Settings
  - 6.2 Advanced Settings

Summary

The **USB Tools** page is used to manage services related to the device's USB connector.

This chapter of the user manual provides an overview of the USB Tools page in RUTX08 devices.

If you're having trouble finding this page or some of the parameters described here on your device's WebUI, you should **turn on "Advanced WebUI" mode**. You can do that by clicking the "Advanced" button, located at the top of the WebUI.

General

The **General** section is used to manage global USB settings. It consists of two subsections:

USB Mount Settings

The **USB Mount Settings** section is used to set the transfer type (synchronous or asynchronous) for the USB device.

Synchronous ensures that all changes to the according filesystem are immediately flushed to disk. **Enabling this will drastically lower the life expectancy of your USB device.**

Mounted File Systems

The **Mounted File Systems** list displays USB mass storage devices (MSD) currently attached to this device.

Field	Value	Description
Device	filepath; default: **/dev/sd***	The filesystem of the attached USB MSD.
Mount Point	filepath; default: **/mnt/sd***	The root directory of the mounted filesystem.
Available	string; default: **none**	Displays available storage information for a mounted USB MSD.
Used	string; default: **none**	Displays used storage information for a mounted USB MSD.
In Use	string; default: **none**	Indicates whether a USB MSD is currently in use by some device service or not.

The **Format** button can be used to format the attached device, and the **Unmount** button next to each entry in the list is used to unmount an attached USB device. Please make sure to unmount a USB device before physically detaching it from the RUTX08.

The **Refresh** is used to refresh the information in the Mounted File Systems list. If you attach a new USB device and cannot see it in the list, try clicking the 'Refresh' button.

Note: Usb with NTFS partition is supported in read-only mode.

Printer Server

The **Printer Server** feature provides the possibility to configure access to a printer that is connected to the USB port of the device. After the printer is connected to the device's USB port and configured, it can be utilized by users in the local network (LAN, WiFi) or remotely.

The 'Add' button lets you add and manage additional printers. To configure a printer instance, click the Edit button located next to it:

Field	Value	Description
Enable	off | on; default: **off**	Turns USB printer support on or off.
Device	filepath; default: **/dev/usb/lp0**	Printer's device file.
Port	integer [9100..9109]; default: **9100**	Printer's TCP port.
Bidirectional mode	off | on; default: **on**	Turns bidirectional mode on or off.

For step-by-step instructions on how to use a printer with RUTX08, **click here.**

USB to Serial

The device's USB connector can also be used as a **serial port**.

Field	Value	Description
Enabled	off | on; default: **off**	Turns the USB to Serial service on or off.
Baud rate	300 | 1200 | 2400 | 4800 | 9600 | 19200 | 38400 | 57600 | 115200; default: **115200**	Data rate for serial data transmission (in bits per second).
Data bits	5 | 6 | 7 | 8; default: **8**	Number of data bits for each character
Parity	None | Odd | Even; default: **None**	In serial transmission, parity is a method of detecting errors. An extra data bit is sent with each data character, arranged so that the number of 1 bits in each character, including the parity bit, is always odd or always even. If a byte is received with the wrong number of 1s, then it must have been corrupted. However, an even number of errors can pass the parity check. • **None** (**N**) - no parity method is used • **Odd** (**O**) - the parity bit is set so that the number of "logical ones (1s)" has to be odd. • **Even** (**E**) - the parity bit is set so that the number of "logical ones (1s)" has to be even.
Stop bits	1 | 2; default: **1**	Stop bits sent at the end of every character allow the receiving signal hardware to detect the end of a character and to resynchronise with the character stream. Electronic devices usually use one stop bit. Two stop bits are required if slow electromechanical devices are used.

Flow control	None | RTS/CTS | Xon/Xoff; default: **None**	In many circumstances a transmitter might be able to send data faster than the receiver is able to process it. To cope with this, serial lines often incorporate a "handshaking" method, usually distinguished between hardware and software handshaking. • **RTS/CTS** - hardware handshaking. RTS and CTS are turned OFF and ON from alternate ends to control data flow, for instance when a buffer is almost full. • **Xon/Xoff** - software handshaking. The Xon and Xoff characters are sent by the receiver to the sender to control when the sender will send data, i.e., these characters go in the opposite direction to the data being sent. The circuit starts in the "sending allowed" state. When the receiver's buffers approach capacity, the receiver sends the Xoff character to tell the sender to stop sending data. Later, after the receiver has emptied its buffers, it sends an Xon character to tell the sender to resume transmission.
Serial type	Console | Over IP | Modbus gateway | NTRIP client; default: **Console**	Specifies the serial connection type. **Look to the sections below for information on different USB to Serial type options.**

Console

Console mode requires no further configuration than the settings above and is used as a direct-access method to the device's shell interface. For this purpose you may want use such applications as PuTTY on Windows and microcom, minicom, picocom or similar applications on Linux.

Over IP

The **Over IP** serial type is used to manage serial connections over a TCP/IP network.

Field	Value	Description
Protocol	TCP | UDP; default: **TCP**	Protocol used in the communication process.
Mode	Server | Client | Bidirect; default: **Server**	The role of this device in the serial connection: • **Server** - the device waits for incoming connections • **Client** - the device initiates the connection. • **Bidirect** - acts as client by default but waits for incoming connections at the same time.
No leading zeros	off | on; default: **off**	When checked, indicates that the first hex zeros should be skipped.
Server settings: Port	integer [0..65535]; default: **none**	Internal port number used to listen for incoming connections.
Server settings: Timeout (s)	integer; default: **none**	Specifies an inactivity time limit (in second) after which an inactive clients is disconnected.

Server settings: Open port automatically	off | on; default: **on**	Automatically adds a traffic rule in the firewall configuration to open the required port for NTRIP communication.
Client settings: Server Address	ip | host; default: **none**	IP address or hostname of the server that this client will connect to.
Client settings: Port	integer [0..65535]; default: **none**	Server's listening port number.
Client settings: Reconnect interval (s)	integer; default: **none**	Time period (in seconds) between reconnection attempts in case a connection fails.
Echo	off | on; default: **off**	Turn USB to serial echo on or off.

Modbus gateway

The **Modbus gateway** serial type allows redirecting TCP data coming to a specified port to an RTU specified by the Slave ID. The Slave ID can be specified by the user or be obtained directly from the Modbus header.

Field	Value	Description
Listening IP	ip; default: **0.0.0.0**	IP address to listen for incoming connections. The default value (*0.0.0.0*) means that this device will listen for incoming connections on any interface or IP address.
Port	integer [0..65535]; default: **502**	Port number to listen for incoming connections.
Slave ID configuration type	User defined | Obtained from TCP; default: **User defined**	Specifies whether slave IDs are user defined or automatically obtained from TCP.
Slave ID | Permitted slave IDs	integer | range of integers; default: **1 or 1-247**	Specifies the slave ID of range of permitted slave IDs. The way this field is named and its function depends on the value of the *Slave ID configuration* field. A range of IDs can be specified by placing a hyphen (-) between two integer numbers. For example, if you permit slave IDs in the range of 10 to 20, you would specify it as: *10-20* You can also specify multiple values that are not connected in a range using commas (,). For example, to specify 6, 50 and 100 as permitted slave IDs, you would have to use: *6,50,100*
Open port automatically	off | on; default: **on**	Automatically adds a traffic rule in the firewall configuration to open the required port for serial communication. **Caution:** use with care if listening IP is left as the default value (*0.0.0.0*). Leaving it as such will leave the device open for remote connections on the specified port.
Echo	off | on; default: **off**	Turn USB to serial echo on or off.

NTRIP client

Networked Transport of RTCM via Internet Protocol (NTRIP) is a protocol for streaming differential GPS (DGPS) data over the Internet in accordance with specification published by RTCM.

Field	Value	Description
IP address	ip; default: **0.0.0.0**	IP address of an NTRIP server.
Port	integer [0..65535]; default: **none**	TCP/UDP port used for NTRIP communication.
Mount point	filepath | string; default: **none**	NTRIP mount point.
Data format	NTRIP V2.0 TCP/IP | NTRIP V2.0 RSTP/RTP | NTRIP V1.0 | Automatic detection | NTRIP V2.0 UDP; default: **NTRIP V1.0**	Version of NTRIP protocol.
Username	string; default: **none**	Username for authentication to an NTRIP server.
Password	string; default: **none**	Password for authentication to an NTRIP server.
Default NMEA string	string; default: **none**	Optional NMEA string that will be used as the default value when initiating the connection to the NTRIP server (this value is only sent to the server if there is no NMEA from router's GPS device).
Use device GPS	off | on; default: **off**	Allows to obtain default NMEA string from the router's GPS device. Only works if GPS service is enabled and location fix is obtained at the time of NTRIP service start.
Open port automatically	off | on; default: **on**	Automatically adds a traffic rule in the firewall configuration to open the required port for serial communication.
Echo	off | on; default: **off**	Turn USB to serial echo on or off.

IP Filter

The **IP Filter** section is used for configuring which network is allowed to communicate with the device. You may add a new instance by selecting the Interface and pressing Add.

Then enter the IP address and save.

Network Shares

The **Network Shares** section is used to manage Network-attached storage (NAS) such as USB drives and hard drives. The RUTX08 device supports the following file system architectures:

- FAT
- FAT32
- exFAT
- NTFS
- ext2
- ext3
- ext4

General Settings

The **General** section is used to set up **Samba** - a software solution for using the Server Message Block (SMB) networking protocol, which provides shared file access between nodes on a computer network. Refer to the figures and table below for more information about Samba configuration.

Field	Value	Description
Enable	off | on; default: **off**	Turns Samba on or off.
Hostname	string; default: **Router_share**	Name of the Samba server.
Description	string; default: **Router share**	Short description about the Same server.
Workgroup	string; default: **WORKGROUP**	Name of the server's workgroup.
Share home-directories	off | on; default: **on**	Allows system users to reach their home directories via network shares.
Interfaces	lan; default: **none**	Bind samba server to specified interfaces
Insert custom configuration to config	string; default: **none**	Insert custom line to configuration file.

Shared Directories

The **Shared Directories** section is used to configure access to the device's files and directories, including USB storage drives. The list of Shared Directories is empty by default; click the 'Add' button in order to create a new configuration:

The newly added Shared Directory configuration should look similar to this:

Field	Value	Description

Name	string; default: **none**	Name of a shared directory.
Path	filepath; default: **/mnt/**	Path to a shared directory. To share an entire drive, choose an automatically generated path from this drop-down box (for example, /mnt/sda1). To share a specific directory on the drive, specify the full path to that directory (for example, /mnt/sda1/shared/video).
Allowed users	samba user(s); default: **none**	Samba user(s) that are permitted to access a Shared Directory. Users can be created from the Users menu tab.
Read-only	off | on; default: **off**	Makes a Shared Directory read-only, which means the directory can only be accessed to view and read files (not write).
Browseable	off | on; default: **on**	Makes a Shared Directory browsable; i.e., visible in shared directory network discovery.
Allow guests	off | on; default: **off**	Turns guest access on or off. Guest access allows anonymous connections to a Shared Directory.
Delete	-(interactive button)	Deletes a Shared Directory configuration.

To connect to the router's SAMBA server from Windows, specify the address in this format:

```
\\smb_server_address\share_name
```

Replace *smb_server_address* with the IP address of this device or SAMBA share hostname; replace *share_name* with the name of the "share" (as specified in the 'Name' field). For example:

```
\\192.168.1.1\my_share
\\Router_share\johns_files
```

Users

The **Users** section is used to create Samba users that can be granted access to Shared Directories. To add a new user, enter a custom username, password and click the 'Add' button.

The newly added User should appear in the Users list. To change the password of a Samba User, click the 'Edit' button located next to it:

This will redirect you to the Settings page for that User which should look similar to this:

DLNA

The **Digital Living Network Alliance (DLNA)** standard provides the possibility to stream media files from local storage to DLNA-capable devices such as computers, Smart TVs, tablets, etc.

This page is used to configure the DLNA service on the device. When the DLNA service is enabled,

the device listens for incoming connections on the specified network interface and port number. Clients connecting to that port number on that interface will be granted access to the media files residing in the specified directories.

DLNA devices use Universal Plug and Play (UPnP) to discover and communicate with each other on a network. To access the DLNA server hosted on this device you will need a DLNA-capable application. If you're using your phone or other mobile device, simply search for DLNA in the device's application store. On Linux and Windows computers you can use applications capable of playing network media streams (such as VLC). Smart TVs and similar devices may have DLNA support built-in by default.

File format types supported by the DLNA service on this device:

- Audio - WMA, WAV, MP3, FLAC
- Image - JPEG
- Video - WMV, MTS, MP4, MKV, MOV

Note: DLNA is additional software that can be installed from the **System → Package Manager** page.

General Settings

Field	Value	Description
Enable	off | on; default: **off**	Turns the DLNA service on or off.
Port	integer [0..65535]; default: **8200**	DLNA service listening port.
Friendly name	string; default: **Teltonika DLNA Server**	The name of this server as it will be displayed to clients.
Root container	Standard container | Browse directory | Music | Video | Pictures; default: **Standard container**	Specifies which type of files will be made available in DLNA file sharing. • **Standard container** - allows browsing the shared media directory but also organizes files by type (music, video, pictures). • **Browse directory** - allows browsing the shared media directory . • **Music, Video, Pictures** - only shares the files of the specified type.

Media directories	filepath(s); default: **/mnt**	A list of directories that will be scanned by the DLNA service and made available to clients. You can also specify what types of files should be displayed from the directy. Use 'A' for audio, 'V' for video, 'P' for images followed by a comma and the path to the directory. For example: • *A,/mnt* - share only audio files from the */mnt/sda1* directory. • *V,/mnt* - share only video files from the */mnt/sda1* directory. • *P,/mnt* - share only image files from the */mnt/sda1* directory. Click the plus symbol to add more directories.
Album art names	filename(s); default: **Album.jpg**	Name of the file(s) to check when searching for album art images. Click the plus symbol to specify more file names.

Advanced Settings

Field	Value	Description
Interfaces	network interface; default: **br-lan**	Network interface(s) on which this DLNA server will be hosted.
Database directory	filepath; default: **/var/run/minidlna**	The directory where DLNA will store its database and album art cache.
Enable inotify	off | on; default: **on**	Turns inotify on or off. Inotify is a Linux kernel subsystem that notices changes to the filesystem and reports on these changes to applications, in this case, the DLNA service.
Enable TIVO	off | on; default: **off**	Turn support for streaming .jpg and .mp3 files to a TiVo supporting HMO on or off.
Strict to DLNA standard	off | on; default: **off**	Set this to strictly adhere to DLNA standards. Turning this on will allow server-side downscaling of very large JPEG images, which may hurt JPEG serving performance on (at least) Sony DLNA products.
Notify interval	integer; default: **900**	Notify interval in seconds.