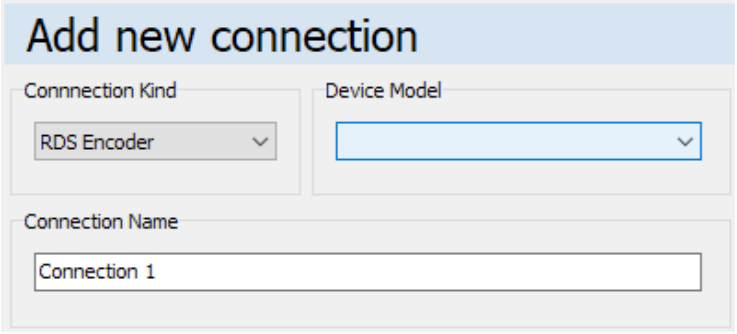
Supported RDS Encoders

and other devices and providers

To Add a New Device (e.g. RDS Encoder)

1. Double-click on **Add new connection** icon.

2. In the **Connection Kind** field, select **RDS Encoder** or another, appropriate kind**.**



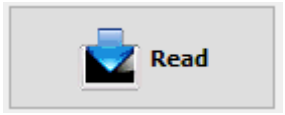
3. Select the **Device Model.**
4. Configure the connection parameters.
5. Confirm by **Add** button.

To Activate the Application

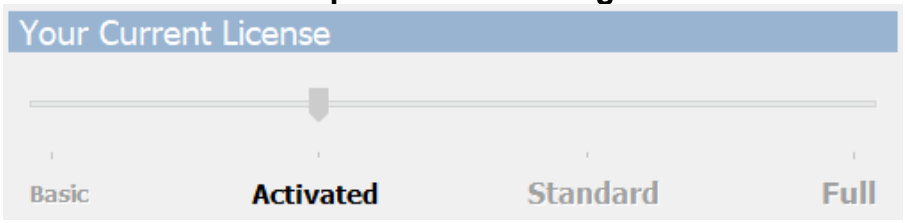
The Activation is required only for free use with FM broadcast RDS encoders from Pira.cz. The Activation is a simple and automated process of getting access to all functions and features associated with these RDS encoders.

The Activation is completely free of charge and it's supported by all RDS encoders and communication protocols with the exception of Demo encoder and those marked with the 🗝 (key) icon.

1. Make sure the connection is set up as **bidirectional**.
2. Perform any bidirectional operation with the RDS encoder, for example **RDS Content – Program – Read**:



3. Check the status in **Help – License Manager**:



4. The Activation is permanent in the context of the installation and it's valid for all Connections.

Note:

Users who purchased the Standard or Full license also got all benefits of the Activated license.

Connection Kind: RDS Encoder / Device Model: **P164**

Communication interface	Ethernet, USB
Minimum firmware version required	2.2h
Free use (supports Activation)	Yes ✓
Communication protocol support	ASCII, UECP, XCMD
Default communication protocol	ASCII, XCMD
RDS Spy support	Yes ✓
Data Sets	6
Direct virtual port support	Yes ✓
Product purchase	

RDS Encoder / **P132**

Communication interface	Ethernet, USB
Minimum firmware version required	2.2h
Free use (supports Activation)	Yes ✓
Communication protocol support	ASCII, UECP, XCMD
Default communication protocol	ASCII, XCMD
RDS Spy support	Yes ✓
Data Sets	6
Direct virtual port support	Yes ✓

RDS Encoder / **P232**

Communication interface	Device dependent
Minimum firmware version required	2.2h
Free use (supports Activation)	Yes ✓
Communication protocol support	ASCII, UECP, XCMD
Default communication protocol	ASCII, XCMD
RDS Spy support	Yes ✓
Data Sets	6
Direct virtual port support	Yes ✓
Product purchase	

The P232 encoder is widely used in various FM equipments. It can be identified by 44-pin ‘46K80’ integrated circuit and 16.000 MHz crystal on the device’s board.

RDS Encoder / **P232U**

Communication interface	Serial / USB
Minimum firmware version required	2.2h
Free use (supports Activation)	Yes ✓
Communication protocol support	ASCII, UECP, XCMD
Default communication protocol	ASCII, XCMD
RDS Spy support	Yes ✓
Data Sets	6
Direct virtual port support	Yes ✓
Product purchase	

RDS Encoder / **P332**

Communication interface	Ethernet, Serial
Minimum firmware version required	2.2h
Free use (supports Activation)	Yes ✓
Communication protocol support	ASCII, UECP, XCMD
Default communication protocol	ASCII, XCMD
RDS Spy support	Yes ✓
Data Sets	6
Direct virtual port support	Yes ✓

RDS Encoder / **Readbest**

Communication interface	Device dependent
Free use (supports Activation)	Yes ✓ (ASCII protocol)
Communication protocol support	ASCII, UECP 🗝
Default communication protocol	ASCII
RDS Spy support	Yes (from RDS fw version 1.5)
Data Sets	Device dependent
Direct virtual port support	Yes ✓

The Readbest encoder is a C-based software library which is implemented in several FM broadcasting devices. Choose this option if it is explicitly mentioned in the device's manual.

Please request the latest RDS firmware from your vendor.

Notes:

* For this device model, the option 'Encapsulate All Outgoing Data to UECP' is not applicable to ASCII commands which have no UECP equivalent.

RDS Encoder / **Demo Encoder**

Communication interface	TCP/IP (localhost only)
Free use	Yes ✓
Communication protocol support	ASCII, UECP
Default communication protocol	ASCII
RDS Spy support	Yes
Data Sets	4
Direct virtual port support	Yes ✓

The Demo Encoder is not a physical device. Rather than this, it establishes a connection to the software emulator of FM broadcast encoder. The emulation is based on simplified Readbest encoder.

Some RDS features may not be available to the emulator, however the emulator is still useful to discover how RDS works. It will also serve excellently during setup of the External Text Sources before going “on-air”. User may visualize output data by clicking on the RDS Spy button.

For further details about how to configure the remote Demo Encoder, please follow the document readbest.pdf (READBEST RDS Encoder), section **Annexes / Demo Encoder**.

RDS Encoder / **Remote Bridge**

Communication interface	Ethernet, Serial
Free use	Yes ✓
Transfer protocol	Internal Magic RDS 4 protocol (ASCII compatible)
Output communication protocol	Target device dependent
RDS Spy support	N/A
Data Sets	N/A
Direct virtual port support	Yes ✓

The Remote Bridge is not a physical device. Rather than this, it forwards data using unidirectional communication mode to another (remote) Magic RDS 4 application for the purpose of data distribution to remote RDS encoder(s). The Remote Bridge does not need a specific RDS encoder model, i.e. it can send data to a network of different models.

Typically, the Remote Bridge connects to a Virtual port of a Bridge established in the remote Magic RDS 4 application.

For further details about how to configure the remote Magic RDS 4 application, please follow the document m4vp.pdf (Bridges and Virtual Ports), section **Remote Bridge**.

To Buy a License for the Application 🗝

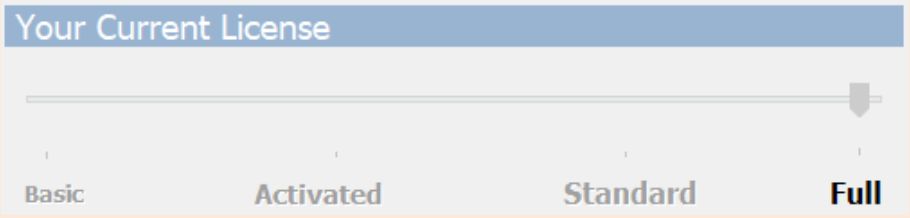
The Standard or Full licence brings further access to the functions and features of the application:

The Standard or Full license is necessary for RDS encoders or communication protocols whose are marked by the 🗝 (key) icon. It is also required for functions and features which are not associated with particular RDS encoder model (for example plain text export, hybrid radio or web publishing).

Until a license is purchased, the application remains fully functional in trial mode. The only limitation is that some text services may contain advertisements.

Purchasing the Standard or Full license is simple and typically takes no more than a few minutes:

1. In the Magic RDS 4 main menu, select **Help – License Manager.**
2. Click on the **Buy License** button to purchase the license online.
3. If you're a new user, continue using the option **Get new User ID**. Follow the instructions in the web browser to pay and generate your User ID and License key.
4. Finally, go again to the **Help – License Manager**. Fill your License key and click on **Apply** button:



5. The license is life-time and includes all future updates. Keep your User ID for future use.

RDS Encoder / **PIRA32** 🗝

Communication interface	Device dependent
Minimum firmware version required	1.6a
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol support	ASCII, UECP
Default communication protocol	ASCII
RDS Spy support	No
Data Sets	2
Direct virtual port support	Yes ✓

The PIRA32 encoder was introduced in 2004 and it is widely used in various FM equipments. It can be identified by 28-pin ‘18F25...’ integrated circuit and 4.332 MHz crystal on the device’s board.

To take advantage of all the benefits of the Radiotext Plus feature, it is recommended to enable the software RT+ encoder in Overview – Extensions.

RDS Encoder / **MRDS1322** 🗝

Communication interface	Device dependent
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	Binary
RDS Spy support	No
Data Sets	1
Direct virtual port support	No

These encoders are embedded in various FM equipments as a basic RDS encoder solution. The Magic RDS 4 now gives access to most of advanced RDS features also to users of the MicroRDS / TinyRDS / MRDS1322 encoders:

- Common or independent control of up to 128 encoders
- Direct support of connection over Ethernet
- Radiotext Plus (RT+) and real-time (CT) transmission
- External text sources with powerful text processing tools
- Task scheduler, Text conditions, SNMP, ASCII terminal emulator, encoder backup/restore
- Connection bridges with virtual ports and communication protocol translation (for example from UECP)

The MRDS1322 encoder can be identified by 20-pin ‘13K22’ integrated circuit and 4.332 MHz crystal on the device’s board.

By default, the software RT+ encoder is enabled for this model. You may disable it in Overview – Extensions.

Note: Legacy MRDS192 device requires I^2C compatible interface and is therefore not supported by the Magic RDS 4. In such case please consider purchasing any of the current RDS encoder models.

RDS Encoder / **Generic UECP** 🗝
DAB+ Encoder / **Generic UECP** 🗝

Communication interface	Device dependent
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	UECP
RDS Spy support	No
Data Sets	Device dependent
Direct virtual port support	No

This option applies to all RDS encoders which support a substantial part of the UECP (SPB 490) specification. This option also applies to all DAB+ encoders which support UECP MEC AA (DL) and MEC 48 (DL+). The Magic RDS 4 now gives access to advanced functions, including some RDS features which are not contained in the original UECP document:

- Common or independent control of up to 128 encoders
- Radiotext Plus (RT+) and Dynamic PS software extension
- External text sources with powerful text processing tools
- Task scheduler, Text conditions, SNMP, ASCII terminal emulator
- Connection bridges with virtual ports
- DAB+ Dynamic Label and DL+

To enable the DAB+ Dynamic Label, go to RDS Content – Radiotext – DAB+.

RDS Encoder / **Lite ASCII** 🗝

Communication interface	Device dependent
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	Basic set of ASCII commands
RDS Spy support	No
Data Sets	1
Direct virtual port support	No

These encoders are available either stand-alone or embedded in various FM equipments as a basic RDS encoder solution.
The Magic RDS 4 now gives access to some advanced RDS features also to users of the ‘Lite ASCII’ encoders:

- Common or independent control of up to 128 encoders
- Radiotext Plus (RT+), if supported by the encoder
- External text sources with powerful text processing tools
- Task scheduler, Text conditions, SNMP, ASCII terminal emulator
- Connection bridges with virtual ports and communication protocol translation

The ‘Lite ASCII’ encoder can be identified by its communication protocol (see the original device manual):

- The ‘Lite ASCII’ encoder uses specific command set including TEXT, DPS, DPSS, PARSE
- Any command entry is confirmed by ‘OK’ or ‘NO’ sequence

RDS Encoder / **User Defined 1** 🗝
General Text Output / **User Defined 1** 🗝

Communication interface	Device or application dependent
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	User defined ASCII commands
RDS Spy support	N/A
Data Sets	N/A
Direct virtual port support	No

This model represents a general purpose text data (“now playing”) output to any device or application which accepts ASCII text format or HTTP query (URL) format. The target may be a physical device as well as streaming server application like Shoutcast etc.

This model selection is intended especially for use with the External text tool (“now playing” etc.). Static content cannot be edited by this way. After selecting of this model, the user can define optional Prefix and Suffix for Radiotext and Dynamic PS. To define the Prefix and Suffix, go to Device Setup – Special. To find the correct Prefix and Suffix, please follow your device or target application documentation.

Further URL format options are available for HTTP query method. For this method, the Prefix and Suffix fields are usually blank.

Example: If following prefix is defined for Radiotext: **`RT=`**

resulting output string will be: **`RT=The text from External text source here↵`**

General Text Output / **XML Template** 🗝

Communication interface	Device or application dependent
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	User defined datagram structure
RDS Spy support	N/A
Data Sets	N/A
Direct virtual port support	No

This model represents a general purpose text data (“now playing”) output to any device or application which requires specific ASCII data format, for example the XML. It is a more general variant of previous User Defined 1 model. The target may be a physical device as well as another application or service.

This model selection is intended especially for use with the External text tool (“now playing” etc.). Static content cannot be edited by this way. After selecting of this model, the user can define the output data structure.

To define the data structure, go to Device Setup – Special. To find the correct format, please follow your device or target application documentation.

Example:

If following simple XML template is defined: **`<root><current>%TEXT%</current></root>`**
resulting output string will be:
`<root><current>The text from External text source here</current></root>`

DAB+ Encoder / **AVT PAD DirectFTP** 🗝

Communication interface	FTP upload or file save
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	AVT PAD DL+
RDS Spy support	N/A
Data Sets	N/A
Direct virtual port support	No

With the AVT DAB+ encoders the customer is enabled to provide PAD data directly to the encoder, instead to a DAB multiplexer. Dynamic PAD data can be loaded via FTP. The FTP account offers a folder “sls” for the dls file named “text.dls”.

Notes:

- The file encoding is UTF-8
- In the External Text Sources, enable the RT+

Example:

```
You are listening: <item.artist>Best</item.artist> - <item.title>Song</item.title>
```

This will display on conventional DAB+ receiver: “You are listening: Best - Song” and on enhanced DAB+ receivers set “Best” as the artist and “Song” as the song title.

Hybrid Radio / **Simple HTTP** 🗝

Communication interface	HTTP GET or HTTP POST
Free use (supports Activation)	No, requires Standard or Full version
Communication protocol	HTTP, HTTPS
RDS Spy support	N/A
Data Sets	N/A
Direct virtual port support	N/A

The Simple HTTP model allows Magic RDS 4 to send text or metadata to external systems using standard HTTP requests. This can be used to integrate with web services, automation systems, metadata platforms, hybrid radio services, custom scripts, and other applications capable of receiving HTTP GET or POST requests.

The target server is specified by its URL. Additional data can be transmitted as configurable parameters. Depending on the selected method, parameters are either appended to the URL (GET) or sent in the request body (POST).

Typical applications include updating "Now Playing" information, sending song titles and artist names to web services, triggering external actions, or exchanging metadata with third-party systems.

All parameter values are automatically URL-encoded before transmission to ensure correct handling of special characters and international text.

For exact setup, follow the data format specification of the destination service.