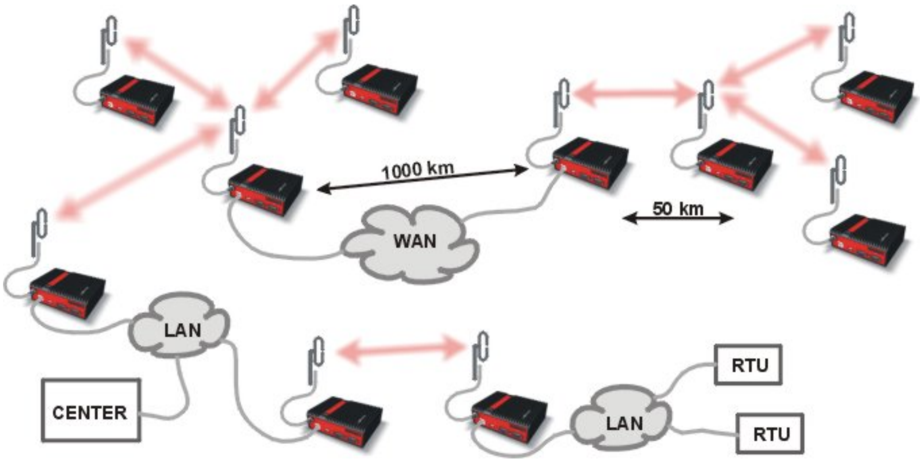
RADIO DATA NETWORKS



RipEX
Application notes

version 1.2
1/31/2012
fw 1.0.9.0

RACOM s.r.o. • Mirova 1283 • 592 31 Nove Mesto na Morave • Czech Republic
Tel.: +420 565 659 511 • Fax: +420 565 659 512 • E-mail: racom@racom.eu

www.racom.eu

Table of Contents

1. Address planing

In Router mode standard IP routing is used between individual RipEX radio modems and their interfaces. The only non-standard feature is that even if you assign all RipEX's radio interface IP addresses to a single network, the RipEX's may not "hear" each other over the radio channel; therefore, routing tables should include even routes within the same network (over repeaters).

This Application Note draws attention to certain situations in which routing tables can be simplified significantly.

1.1. End devices connected via serial interface

Every RipEX radio modem has two network interfaces, and hence two IP addresses. First is the Ethernet interface, second the radio interface. Serial interfaces are defined by their UDP port and are shared for the entire RipEX modem; as a result both RipEX IP addresses can be used to access them (both IP addresses work equally well).

The destination IP address of a packet received via the serial interface is determined inside the radio modem from the "SCADA address" depending on the protocol used, either using a mask or table (see RipEX manual, Adv. config., Protocols[1]). The source IP is generated similarly.

If all devices are connected to RipEX's via serial interface, it is helpful to only use the radio IP addresses for translation and routing of data. Ethernet IP addresses may be assigned randomly (you could keep their defaults, however we recommend setting Ethernet addresses similar to radio IP addresses to keep things organised). Remote service access over the radio channel is also possible via the IP addresses of the radio interface.

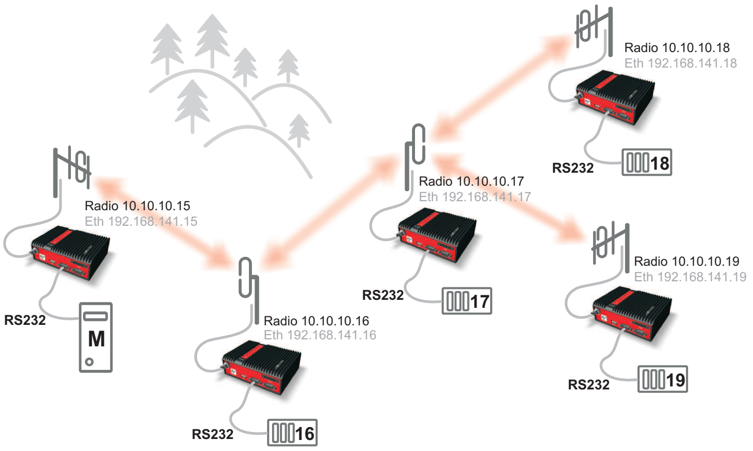


Fig. 1.1: Network 1

The following paragraph shows routing tables for individual radio modems which enable mutual communication between all devices. All destinations share the mask 255.255.255.255, i.e. 10.10.10.xx/32, interface Auto or Radio:

- For 10.10.10.15

```
Destination via Gateway
10.10.10.17 via 10.10.10.16
```

[1] http://www.racom.eu/eng/products/m/ripex/h-menu.html#protocols

```
10.10.10.18 via 10.10.10.16
10.10.10.19 via 10.10.10.16
```

- For 10.10.10.16

```
10.10.10.18 via 10.10.10.17
10.10.10.19 via 10.10.10.17
```

- For 10.10.10.17

```
10.10.10.15 via 10.10.10.16
```

- For 10.10.10.18

```
10.10.10.15 via 10.10.10.17
10.10.10.16 via 10.10.10.17
10.10.10.19 via 10.10.10.17 (this record is only necessary if you require
                             communication between end devices 19 and 18)
```

- For 10.10.10.19

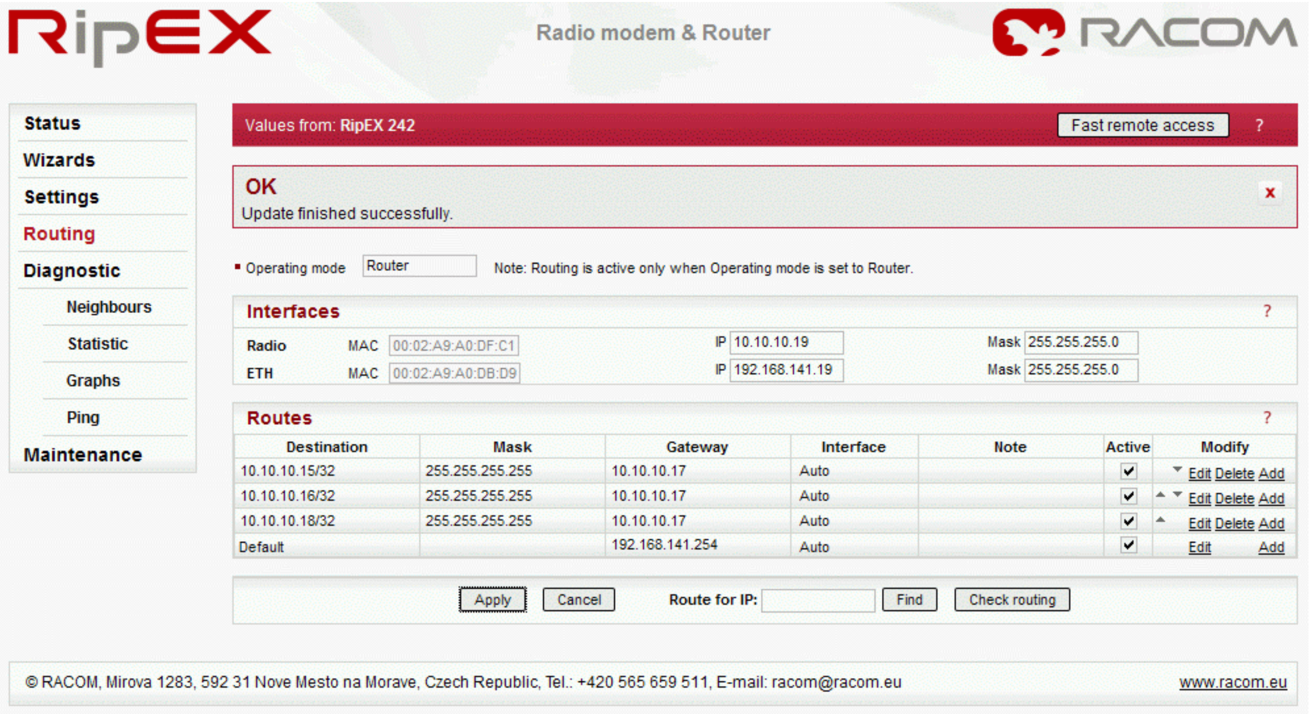
```
10.10.10.15 via 10.10.10.17
10.10.10.16 via 10.10.10.17
10.10.10.18 via 10.10.10.17 (this record is only necessary if you require
                             communication between end devices 19 and 18)
```

To display the full routing table type "ip route show table normal" in CLI interface

- For 10.10.10.19

```
10.10.10.15 via 10.10.10.17 dev radio  proto static
broadcast 10.10.10.0 dev radio  proto static  scope link  src 10.10.10.19
broadcast 10.10.10.255 dev radio  proto static  scope link  src 10.10.10.19
10.10.10.16 via 10.10.10.17 dev radio  proto static
10.10.10.18 via 10.10.10.17 dev radio  proto static
10.10.10.0/24 dev radio  proto static  scope link
192.168.141.0/24 dev eth0  proto static  scope link
default via 192.168.141.254 dev eth0  proto static
```

An example of a routing table on page Routing for 10.10.10.19



RipEX Radio modem & Router RACOM

Status
Wizards
Settings
Routing
Diagnostic
Neighbours
Statistic
Graphs
Ping
Maintenance

Values from: RipEX 242 | Fast remote access | ?

OK
Update finished successfully.

Operating mode: Router
Note: Routing is active only when Operating mode is set to Router.

Interfaces

Radio	MAC 00:02:A9:A0:DF:C1	IP 10.10.10.19	Mask 255.255.255.0
ETH	MAC 00:02:A9:A0:DB:D9	IP 192.168.141.19	Mask 255.255.255.0

Routes

Destination	Mask	Gateway	Interface	Note	Active	Modify
10.10.10.15/32	255.255.255.255	10.10.10.17	Auto		☑	Edit Delete Add
10.10.10.16/32	255.255.255.255	10.10.10.17	Auto		☑	Edit Delete Add
10.10.10.18/32	255.255.255.255	10.10.10.17	Auto		☑	Edit Delete Add
Default		192.168.141.254	Auto		☑	Edit Add

Apply | Cancel | Route for IP: | Find | Check routing

© RACOM, Mirova 1283, 592 31 Nove Mesto na Morave, Czech Republic, Tel.: +420 565 659 511, E-mail: racom@racom.eu www.racom.eu

If SCADA device addresses can be chosen arbitrarily, routing can be significantly simplified when radio IP addresses can be grouped to subnets according to radio network layout.

One example of simplification is shown with repeaters connecting to separate subnets. The routing table can then contain a single record for all devices on the subnet.

In this example the first repeater connects to subnet 10.10.10.0/29, i.e. devices may have addresses from 10.10.10.1 to 10.10.10.6 (10.10.10.0 is reserved for the subnet, address 10.10.10.7 for broadcasting).

See e.g. http://www.subnet-calculator.com/subnet.php?net_class=A

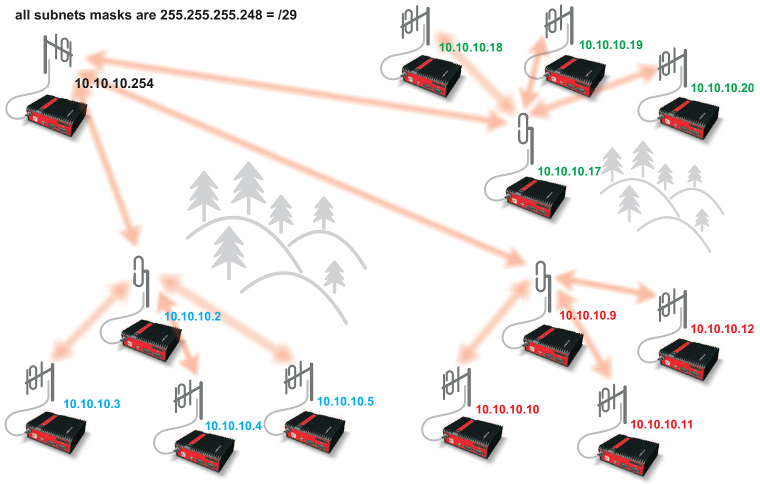


Fig. 1.2: Network with subnets

- For 10.10.10.254

```
Destination via Gateway
10.10.10.1/29  via 10.10.10.2
10.10.10.8/29  via 10.10.10.9
10.10.10.16/29 via 10.10.10.17
```

- For 10.10.10.2

```
10.10.10.8/29  via 10.10.10.9
10.10.10.16/29 via 10.10.10.17
```

- For 10.10.10.3 and 10.10.10.4 and 10.10.10.5

```
10.10.10.248/29 via 10.10.10.2
10.10.10.8/29   via 10.10.10.2
10.10.10.16/29  via 10.10.10.2
```

- For 10.10.10.9

```
10.10.10.1/29 via 10.10.10.2
10.10.10.8/29 via 10.10.10.9
```

- For 10.10.10.10 and 10.10.10.11 and 10.10.10.12

```
10.10.10.248/29 via 10.10.10.9
10.10.10.1/29   via 10.10.10.9
10.10.10.16/29  via 10.10.10.9
```

- For 10.10.10.17

```
10.10.10.1/29  via 10.10.10.2
10.10.10.16/29 via 10.10.10.17
```

- For 10.10.10.18 and 10.10.10.19 and 10.10.10.20

```
10.10.10.248/29 via 10.10.10.17
10.10.10.1/29   via 10.10.10.17
10.10.10.8/29   via 10.10.10.17
```

1.2. End devices connected over Ethernet

Both radio modem's network interfaces must be used for routing. Radio modem routing works the same as standard IP routing – for more information refer to http://www.comptechdoc.org/independent/networking/guide/netguide.pdf chapter Network Routing

Limitations:

A. **If you can set the IP address, network mask, gateway and routing table in the IP device connected to RipEX**

There are no limitations to setting up routing in this case. The only rule is that the range of radio and Ethernet IP addresses must not overlap.

B. **If you can only set the IP address, network mask and gateway, not the routing table in the IP device connected to RipEX**

In this case destination addresses must not be on the same network (i.e. the destination address must always be outside of the network mask). A destination address is the IP address of one of the devices connected to RipEX's which mutually communicate over the radio channel.

C. **If the connected device allows neither network mask, nor gateway to be set up**

Router mode cannot be used at all; use Bridge mode instead.

1.3. Ethernet addressing

If you can set up IP addresses of the end devices connected over Ethernet, you can simplify routing by hierarchic division into subnets, either complete or for routing purposes only. An example of such network layout follows.

The centre and main repeater form distinct networks with mask 255.255.255.0 (/24), the sub-networks narrow down towards the end devices 255.255.255.192 (/26) and then 255.255.255.248 (/29). Routing tables are only given for a single branch of the network for clarity. They will be similar for other RipEX's. Only Master – Slave type applications are presumed – without any direct communication between Slave devices.

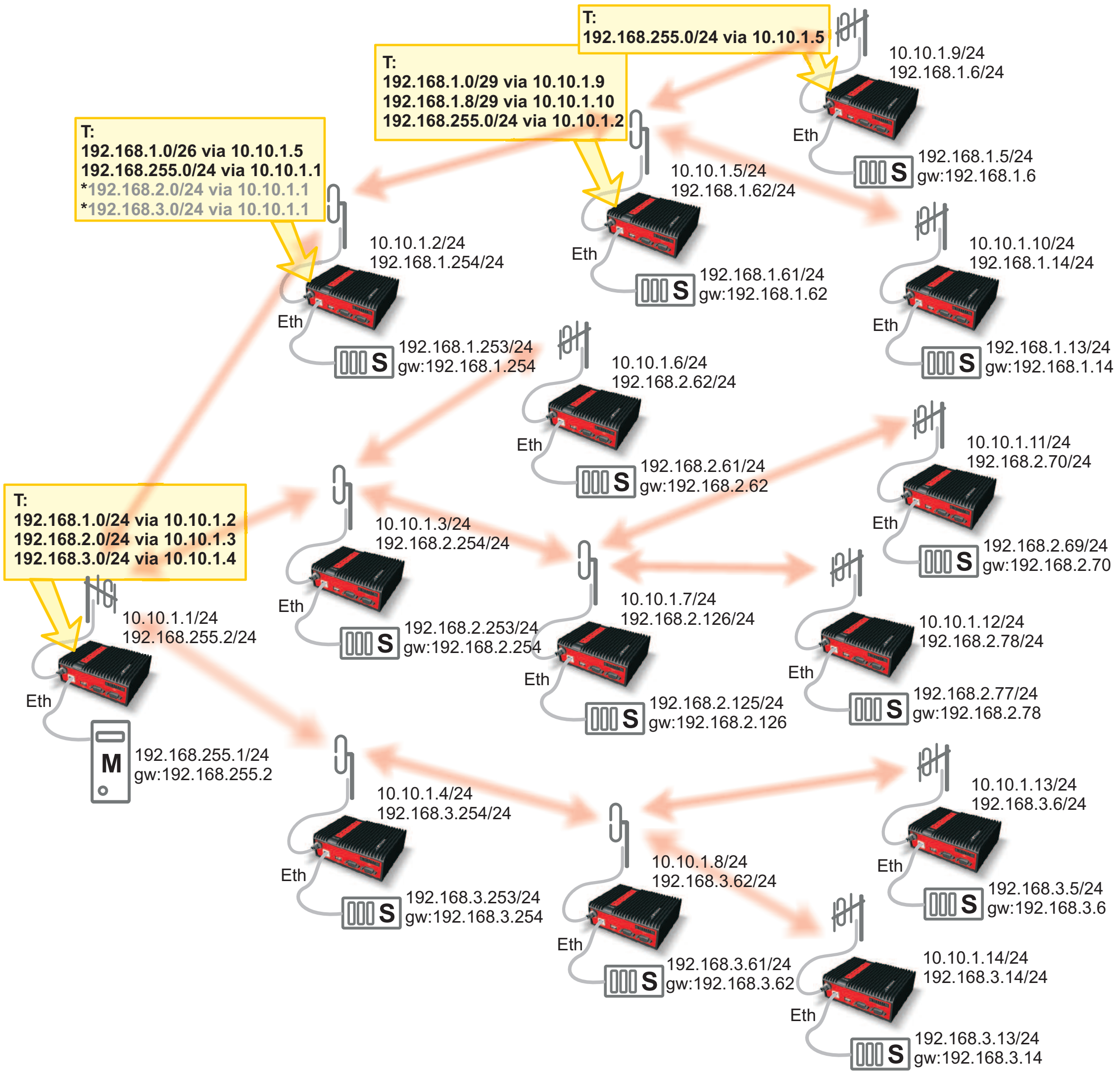


Fig. 1.3: Network with standard masks

Virtual network narrowing may also be used, while in reality narrower masks will be only used for routing purposes. This would allow you to use even the addresses reserved for network and broadcasting, though we do not recommend doing so.

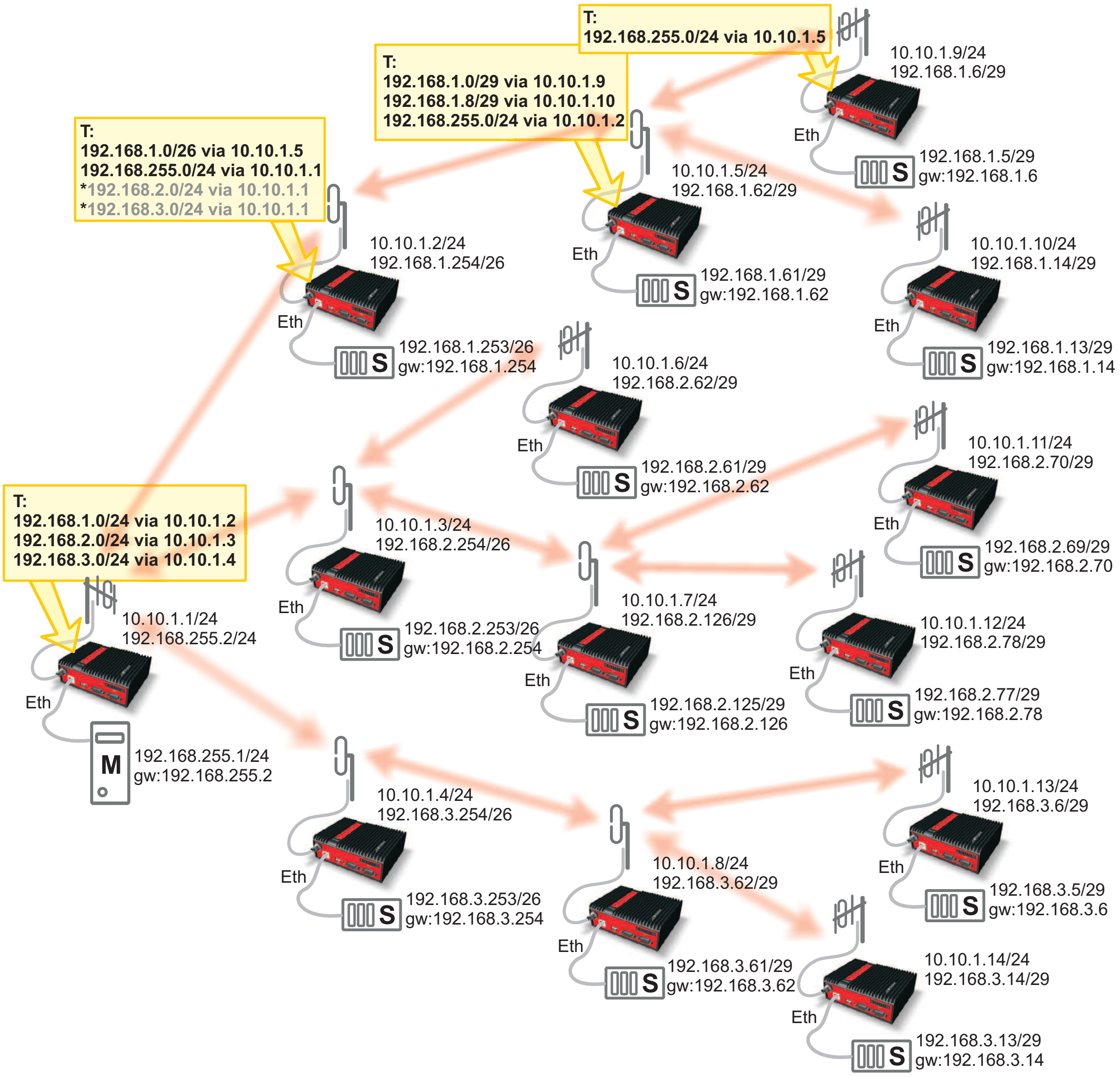


Fig. 1.4: Network with narrowed masks

2. SNMP for RACOM RipEX

2.1. Simple Network Management Protocol

SNMP is a simple, widespread and useful standardised protocol used to read values from devices or to set them. Values can be obtained at regular intervals, saved to a database and then displayed as a graph. It allows you to, for instance, monitor activity on the radio channel, temperature history or data flow through ports.

SNMP also enables you to generate and receive alarms (SNMP traps).

2.1.1. How does SNMP work?

SNMP protocol requires two parties for communication.

One of them is **manager** – you could use commercial as well as free software such as Zeenos, Nagios, Cacti, etc.

The other is **SNMP agent** which is part of RipEX firmware. It does two things:

- Agent responds to manager's queries. Several managers may read values at once and they can send their queries at any time.
- Agent sends its manager notifications (traps) in defined situations (when a value is exceeded or at regular intervals). These traps are only sent to one manager.

For more information on SNMP and its installation and usage go to Simple Network Management Protocol[1] on MSDN website.

2.1.2. SNMP communication

In SNMP each value is uniquely identified using **OID (Object IDentifier)** code.

Normal communication, in which first query and then response are sent, works as follows:

- A query is sent – manager sets message type to **GET**, includes OID for the required value and sets the value to NULL
- Response is returned – agent sets message type to **RESPONSE** and sends the requested value along with its OID

Basic query types are:

SNMP GET – returns a single value
GET-NEXT – returns next value (using the next OID)
GET-BULK – as of SNMPv2. Returns several values in a single packet (for example, the temperature, voltage, number of transmitted messages or bytes per second, etc.).

SNMP allows you to not only receive but also set values using **SET** type message (unsupported by RipEX).

Another message type is **TRAP** which sends spontaneous messages (values) from agent to manager.

[1] http://go.microsoft.com/fwlink/?LinkId=45736

2.1.3. MIB database – Management Information Base

OID, which uniquely identifies every value in SNMP, is formed by a sequence of numbers divided by points. This number is derived from superordinate element's OID, divided by a full stop from the current number. The entire tree structure is saved to MIB database. In addition, MIB database contains the names and descriptions of individual values (OID). Other values can be added to MIB database using part of the structure saved in the MIB file.

An example of OID is 1.3.6.1.4.1.33555.2.1.1.3 in RipEX MIB database which corresponds to serialNumber – (Unsigned32) – Manufacturing serial number of the product.

SNMP does not require an MIB database to work as SNMP packets only include OID; if you don't know the right OID, however, database can help you retrieve it.

2.2. SNMP in RipEX

In RipEX SNMP protocol can be used to:

- Read configuration parameters from MIB,
- Read operation statistics on the radio channel, and
- Sends traps when set thresholds for monitored values are exceeded (TxLost [%], Ucc, Temp, PWR, VSWR, ETH [Rx/Tx], COM1 [Rx/Tx], COM2[Rx/Tx])

For detailed description of individual values refer to section RipEX MIB bellow.

RipEX utilises SNMP versions **SNMPv1** and **SNMPv2c** – it uses **community string** for authentication, which is fixed to "**public**" and cannot be changed. SNMP uses UDP protocol for communication; delivery checks are implemented from version 2.

By default RipEX uses UDP port 161 (SNMP) for queries. Manager, which sends the query, dynamically chooses a port from which it sends its query to RipEX port 161. RipEX replies from port 161 to the dynamically selected port of the manager.

RipEX launches SNMP agent automatically on start-up and it cannot be switched off by user. Only traps' behaviour can be influenced (see Alarm management settings, RipEX manual, Adv. config.[2]).

RipEX sends alarm states (traps) to manager from port 162 (SNMPTRAP). Users cannot change this port number in RipEX.

When using SNMP over radio channel we recommend setting RipEX to router mode. From the point of radio network, SNMP is typically a standalone application sharing the radio channel with others. Thus it causes collisions, which are automatically resolved by the radio channel protocol in router mode. The radio channel uses no protocol in bridge mode, meaning two competing applications can only be run at a great risk of collisions and with the knowledge that packets from both applications may be irretrievably lost.

[2] http://www.racom.eu/eng/products/m/ripex/h-menu.html

2.2.1. RipEX MIB

Name	OID	
ripex	1.3.6.1.4.1.33555.2	
station	1.3.6.1.4.1.33555.2.1	
device	1.3.6.1.4.1.33555.2.1.1	
stationName	1.3.6.1.4.1.33555.2.1.1.1	Name of the station.
type	1.3.6.1.4.1.33555.2.1.1.2	Device type.
serialNumber	1.3.6.1.4.1.33555.2.1.1.3	Manufactoring serial number of the product.
deviceMode	1.3.6.1.4.1.33555.2.1.1.4	Station working mode.
hwVersions	1.3.6.1.4.1.33555.2.1.1.5	
hwVerModem	1.3.6.1.4.1.33555.2.1.1.5.1	HW version of the modem.
hwVerRadio	1.3.6.1.4.1.33555.2.1.1.5.2	HW version of the radio.
swVersions	1.3.6.1.4.1.33555.2.1.1.7	
firmware	1.3.6.1.4.1.33555.2.1.1.7.1	Firmware version.
bootloader	1.3.6.1.4.1.33555.2.1.1.7.2	Bootloader version.
system	1.3.6.1.4.1.33555.2.1.2	
useCpu1	1.3.6.1.4.1.33555.2.1.2.1	Average number of processes during last 1 minute.
useCpu5	1.3.6.1.4.1.33555.2.1.2.2	Average number of processes during last 5 minutes.
useCpu15	1.3.6.1.4.1.33555.2.1.2.3	SAverage number of processes during last 15 minutes.
useMemory	1.3.6.1.4.1.33555.2.1.2.4	System use memory in %.
useLogStorage	1.3.6.1.4.1.33555.2.1.2.5	Use storage for log in %.
interface	1.3.6.1.4.1.33555.2.2	
ifRadio	1.3.6.1.4.1.33555.2.2.1	
rRxFrequency	1.3.6.1.4.1.33555.2.2.1.1	Radio interface RX frequency in Hz.
rTxFrequency	1.3.6.1.4.1.33555.2.2.1.2	Radio interface TX frequency in Hz.
rRfPwr	1.3.6.1.4.1.33555.2.2.1.3	Radio interface RF Power in W.
rEncryption	1.3.6.1.4.1.33555.2.2.1.4	Radio interface encryption method.
rFEC	1.3.6.1.4.1.33555.2.2.1.5	Radio interface FEC.
ifEth	1.3.6.1.4.1.33555.2.2.2	
eGateway	1.3.6.1.4.1.33555.2.2.2.1	Ethernet interface gateway address.
eDhcp	1.3.6.1.4.1.33555.2.2.2.2	Ethernet interface DHCP mode.
eShaping	1.3.6.1.4.1.33555.2.2.2.3	Ethernet interface shaping status.
eBCastMCast	1.3.6.1.4.1.33555.2.2.2.4	Ethernet interface broadcast and multicast status.
ifCom	1.3.6.1.4.1.33555.2.2.3	
ifComNumber	1.3.6.1.4.1.33555.2.2.3.1	The number of COM interfaces.
ifComTable	1.3.6.1.4.1.33555.2.2.3.2	A list of COM interface entries.
ifComEntry	1.3.6.1.4.1.33555.2.2.3.2.1	A COM interface entry.
comIndex	1.3.6.1.4.1.33555.2.2.3.2.1.1	A unique index for each interface.
comUdpPort	1.3.6.1.4.1.33555.2.2.3.2.1.2	COM interface UDP port.
comIdle	1.3.6.1.4.1.33555.2.2.3.2.1.3	COM interface idle in bytes.
comMtu	1.3.6.1.4.1.33555.2.2.3.2.1.4	COM interface MTU in bytes.
comProtocol	1.3.6.1.4.1.33555.2.2.3.2.1.5	COM interface protocol.
statistics	1.3.6.1.4.1.33555.2.3	
stRadio	1.3.6.1.4.1.33555.2.3.1	
stRadioTotal	1.3.6.1.4.1.33555.2.3.1.1	
stRadioTotDuplicates	1.3.6.1.4.1.33555.2.3.1.1.1	Total radio duplicate packets counter.
stRadioTotRepeats	1.3.6.1.4.1.33555.2.3.1.1.2	Total radio repeated packets counter.
stRadioTotLost	1.3.6.1.4.1.33555.2.3.1.1.3	Total radio lost packets counter.
stRadioTotCtlPacketsRX	1.3.6.1.4.1.33555.2.3.1.1.4	Total RX radio control packets counter.
stRadioTotCtlPacketsTX	1.3.6.1.4.1.33555.2.3.1.1.5	Total TX radio control packets counter.
stRadioTotDataErr	1.3.6.1.4.1.33555.2.3.1.1.6	Total radio data error packets counter.
stRadioTotRejected	1.3.6.1.4.1.33555.2.3.1.1.7	Total radio rejected packets counter.
stRadioTotPacketsRX	1.3.6.1.4.1.33555.2.3.1.1.8	Remote station total RX packets counter.
stRadioTotPacketsTX	1.3.6.1.4.1.33555.2.3.1.1.9	Remote station total TX packets counter.
stRadioTotBytesRX	1.3.6.1.4.1.33555.2.3.1.1.10	Remote station total RX bytes counter.
stRadioTotBytesTX	1.3.6.1.4.1.33555.2.3.1.1.11	Remote station total TX bytes counter.
stRadioTotIpErr	1.3.6.1.4.1.33555.2.3.1.1.12	Total radio IP error packets counter.
stRadioTotHeadErr	1.3.6.1.4.1.33555.2.3.1.1.13	Total radio header error packets counter.
stRadioTotFalseSync	1.3.6.1.4.1.33555.2.3.1.1.14	Total radio false sync counter.
stRadioRemNumber	1.3.6.1.4.1.33555.2.3.1.2	The number of remote stations.
stRadioRemTable	1.3.6.1.4.1.33555.2.3.1.3	A list of remote station entries.
stRadioRemEntry	1.3.6.1.4.1.33555.2.3.1.3.1	A radio remote station entry.
stRemIndex	1.3.6.1.4.1.33555.2.3.1.3.1.1	Remote station index.
stRemIpAddr	1.3.6.1.4.1.33555.2.3.1.3.1.2	Remote station IP address.
stRemPacketsRX	1.3.6.1.4.1.33555.2.3.1.3.1.3	Remote station RX packets counter.
stRemPacketsTX	1.3.6.1.4.1.33555.2.3.1.3.1.4	Remote station TX packets counter.
stRemBytesRX	1.3.6.1.4.1.33555.2.3.1.3.1.5	Remote station RX bytes counter.

stRemBytesTX	1.3.6.1.4.1.33555.2.3.1.3.1.6	Remote station TX bytes counter.
stRemDuplicates	1.3.6.1.4.1.33555.2.3.1.3.1.7	Remote station duplicate packets counter.
stRemRepeats	1.3.6.1.4.1.33555.2.3.1.3.1.8	Remote station repeated packets counter.
stRemLost	1.3.6.1.4.1.33555.2.3.1.3.1.9	Remote station lost packets counter.
stRemCtlPacketsRX	1.3.6.1.4.1.33555.2.3.1.3.1.10	Remote station RX radio control packets counter.
stRemCtlPacketsTX	1.3.6.1.4.1.33555.2.3.1.3.1.11	Remote staion TX radio control packets counter.
stRemDataErr	1.3.6.1.4.1.33555.2.3.1.3.1.12	Remote station data error packets counter.
stRemRejected	1.3.6.1.4.1.33555.2.3.1.3.1.13	Remote station rejected packets counter.
stRemTotalPacketsRX	1.3.6.1.4.1.33555.2.3.1.3.1.14	Remote station total RX packets counter.
stRemTotalPacketsTX	1.3.6.1.4.1.33555.2.3.1.3.1.15	Remote station total TX packets counter.
stRemTotalBytesRX	1.3.6.1.4.1.33555.2.3.1.3.1.16	Remote station total RX bytes counter.
stRemTotalBytesTX	1.3.6.1.4.1.33555.2.3.1.3.1.17	Remote station total TX bytes counter.
stCom	1.3.6.1.4.1.33555.2.3.2	
stComNumber	1.3.6.1.4.1.33555.2.3.2.1	The number of COM ports.
stComTable	1.3.6.1.4.1.33555.2.3.2.2	A list of COM port entries.
stComEntry	1.3.6.1.4.1.33555.2.3.2.2.1	
stComIndex	1.3.6.1.4.1.33555.2.3.2.2.1.1	The COM port index.
stComPacketsRX	1.3.6.1.4.1.33555.2.3.2.2.1.2	COM RX packets counter.
stComPacketsTX	1.3.6.1.4.1.33555.2.3.2.2.1.3	COM TX packets counter.
stComBytesRX	1.3.6.1.4.1.33555.2.3.2.2.1.4	COM RX bytes counter.
stComBytesTX	1.3.6.1.4.1.33555.2.3.2.2.1.5	COM TX bytes counter.
watchedValues	1.3.6.1.4.1.33555.2.4	
wvLocal	1.3.6.1.4.1.33555.2.4.1	
wvNoiseLast	1.3.6.1.4.1.33555.2.4.1.1	Local station - last noise value in dBm.
wvNoiseAvg	1.3.6.1.4.1.33555.2.4.1.2	Local station - average noise value in hundredths of dBm.
wvLoadLast	1.3.6.1.4.1.33555.2.4.1.3	Local station - last load value in %.
wvLoadAvg	1.3.6.1.4.1.33555.2.4.1.4	Local station - average load value in hundredths of %.
wvTxlostLast	1.3.6.1.4.1.33555.2.4.1.5	Local station - last Tx Lost value in %.
wvTxlostAvg	1.3.6.1.4.1.33555.2.4.1.6	Local station - average Tx Lost value in hundredths of %.
wvUccLast	1.3.6.1.4.1.33555.2.4.1.7	Local station - last Ucc value in tenths of Volt.
wvUccAvg	1.3.6.1.4.1.33555.2.4.1.8	Local station - average Ucc value in thousandths of Volt.
wvTempLast	1.3.6.1.4.1.33555.2.4.1.9	Local station - last modem temperature value in tenths of °C.
wvTempAvg	1.3.6.1.4.1.33555.2.4.1.10	Local station - average modem temp value in thousandths of °C.
wvRfpwrLast	1.3.6.1.4.1.33555.2.4.1.11	Local station - last RF power value in tenths of W.
wvRfpwrAvg	1.3.6.1.4.1.33555.2.4.1.12	Local station - average RF power value in thousandths of W.
wvVswrLast	1.3.6.1.4.1.33555.2.4.1.13	Local station - last VSWR value from interval <3;25> in tenths
wvVswrAvg	1.3.6.1.4.1.33555.2.4.1.14	Local station - average VSWR value from interval <3;25> in thousandths
wvRemoteNumber	1.3.6.1.4.1.33555.2.4.2	The number of remote stations.
wvRemoteTable	1.3.6.1.4.1.33555.2.4.3	A list of remote stations.
wvRemoteEntry	1.3.6.1.4.1.33555.2.4.3.1	A remote station watched values entry.
wvRemIndex	1.3.6.1.4.1.33555.2.4.3.1.1	A unique index for each remote station.
wvRemIpAddr	1.3.6.1.4.1.33555.2.4.3.1.2	IP address of remote station.
wvRemHearings	1.3.6.1.4.1.33555.2.4.3.1.3	Total heared packets from remote station.
wvRemRssLast	1.3.6.1.4.1.33555.2.4.3.1.4	Remote station - last rss value in dBm.
wvRemRssAvg	1.3.6.1.4.1.33555.2.4.3.1.5	Remote station - average rss value in hundredths of dBm.
wvRemDqLast	1.3.6.1.4.1.33555.2.4.3.1.6	Remote station - last dq value.
wvRemDqAvg	1.3.6.1.4.1.33555.2.4.3.1.7	Remote station - average dq value hundredths.
wvRemNoiseLast	1.3.6.1.4.1.33555.2.4.3.1.8	Remote station - last noise value in dBm.
wvRemNoiseAvg	1.3.6.1.4.1.33555.2.4.3.1.9	Remote station - average noise value in hundredths of dBm.
wvRemLoadLast	1.3.6.1.4.1.33555.2.4.3.1.10	Remote station - last load value in %.
wvRemLoadAvg	1.3.6.1.4.1.33555.2.4.3.1.11	Remote station - average load value in hundredths of %.
wvRemTxlostLast	1.3.6.1.4.1.33555.2.4.3.1.12	Remote station - last Tx Lost value in %.
wvRemTxlostAvg	1.3.6.1.4.1.33555.2.4.3.1.13	Remote station - average Tx Lost value in hundredths of %.
wvRemUccLast	1.3.6.1.4.1.33555.2.4.3.1.14	Remote station - last Ucc value in tenths of Volt.
wvRemUccAvg	1.3.6.1.4.1.33555.2.4.3.1.15	Remote station - average Ucc value in thousandths of Volt.
wvRemTempLast	1.3.6.1.4.1.33555.2.4.3.1.16	Remote station - last modem temperature value in tenths of °C.
wvRemTempAvg	1.3.6.1.4.1.33555.2.4.3.1.17	Remote station – avg modem temp value in thousandths of °C.
wvRemRfpwrLast	1.3.6.1.4.1.33555.2.4.3.1.18	Remote station - last RF power value in tenths of W.
wvRemRfpwrAvg	1.3.6.1.4.1.33555.2.4.3.1.19	Remote station - average RF power value in thousandths of W.
wvRemVswrLast	1.3.6.1.4.1.33555.2.4.3.1.20	Remote station - last VSWR value from interval <3;25> in tenths
wvRemVswrAvg	1.3.6.1.4.1.33555.2.4.3.1.21	Remote station - average VSWR value from interval <3;25> in thousandths

RipEX SNMP Traps

SNMP Traps		
ripextraps	1.3.6.1.4.1.33555.2.10	
trpRss	1.3.6.1.4.1.33555.2.10.1	RSS of remote station is out of range.
trpDq	1.3.6.1.4.1.33555.2.10.2	DQ of remote station is out of range.
trpNoise	1.3.6.1.4.1.33555.2.10.3	Noise value out of range.
trpLoad	1.3.6.1.4.1.33555.2.10.4	Load value out of range.
trpTxlost	1.3.6.1.4.1.33555.2.10.5	Tx Lost value out of range.
trpUcc	1.3.6.1.4.1.33555.2.10.6	Ucc value out of range.
trpTemp	1.3.6.1.4.1.33555.2.10.7	Modem temperature value out of range.
trpRfpwr	1.3.6.1.4.1.33555.2.10.8	RF power value out of range.
trpVswr	1.3.6.1.4.1.33555.2.10.9	VSWR value out of range.
trpLanPr	1.3.6.1.4.1.33555.2.10.10	Ethernet Rx/Tx packet ratio out of range.
trpCom1Pr	1.3.6.1.4.1.33555.2.10.11	COM1 Rx/Tx packet ratio out of range.
trpCom2Pr	1.3.6.1.4.1.33555.2.10.12	COM2 Rx/Tx packet ratio out of range.
trpHwIn	1.3.6.1.4.1.33555.2.10.13	HW input in alarm state.

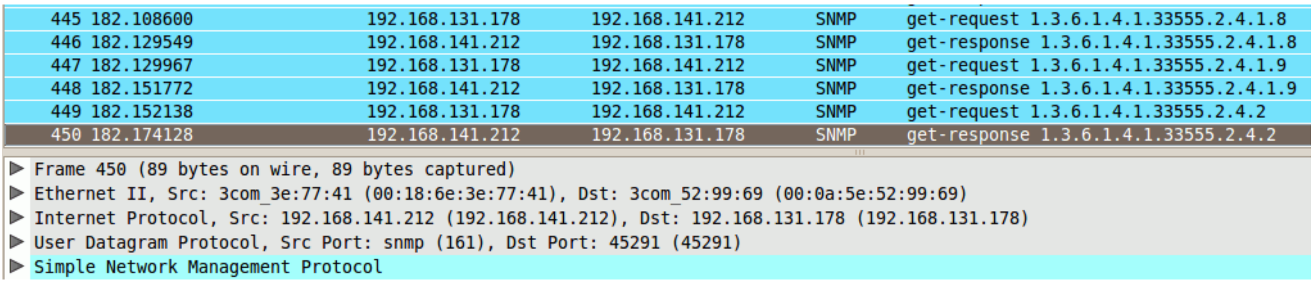
MIB description for RipEX devices – RipEX_MIB.pdf can be downloaded from: http://www.racom.eu/download/hw/ripex/free/eng/Racom-RipEX.mib

2.2.2. Setting up SNMP in RipEX

As we have mentioned, SNMP agent is always running to answer the queries from SNMP manager. Generating SNMP traps for alarms can be turned on and set up in Settings menu, section Device, item Alarm management. For more information see on-line help or chapter Advanced configuration[3] of the manual.

2.2.3. Setting up SNMP in manager

It is important to realise that the average size of a single query and response to a specific OID is approximately 180 Bytes. The following images shows monitoring of part of a communication, with time stamps, between manager 192.168.131.178 and agent (RipEX) 192.168.141.212. **Get-request** alternates with specific OID and **get-response**. The lower part shows message size and **src** and **dst** ports.



```
    445 182.108600        192.168.131.178    192.168.141.212    SNMP    get-request 1.3.6.1.4.1.33555.2.4.1.8
    446 182.129549        192.168.141.212    192.168.131.178    SNMP    get-response 1.3.6.1.4.1.33555.2.4.1.8
    447 182.129967        192.168.131.178    192.168.141.212    SNMP    get-request 1.3.6.1.4.1.33555.2.4.1.9
    448 182.151772        192.168.141.212    192.168.131.178    SNMP    get-response 1.3.6.1.4.1.33555.2.4.1.9
    449 182.152138        192.168.131.178    192.168.141.212    SNMP    get-request 1.3.6.1.4.1.33555.2.4.2
    450 182.174128        192.168.141.212    192.168.131.178    SNMP    get-response 1.3.6.1.4.1.33555.2.4.2
▶ Frame 450 (89 bytes on wire, 89 bytes captured)
▶ Ethernet II, Src: 3com_3e:77:41 (00:18:6e:3e:77:41), Dst: 3com_52:99:69 (00:0a:5e:52:99:69)
▶ Internet Protocol, Src: 192.168.141.212 (192.168.141.212), Dst: 192.168.131.178 (192.168.131.178)
▶ User Datagram Protocol, Src Port: snmp (161), Dst Port: 45291 (45291)
▶ Simple Network Management Protocol
```

Fig. 2.1: Record of communication between manager and agent

The entire MIB for a single RipEX is 10 kilobytes. If SNMP data is downloaded over the radio channel, SNMP generates significant traffic which can influence network performance (collisions, response times, etc.). That's why we recommend using the radio channel to query only selected OIDs and not all data. Set longer SNMP query interval in your manager device. The shortest recommended interval is minutes to tens of minutes.

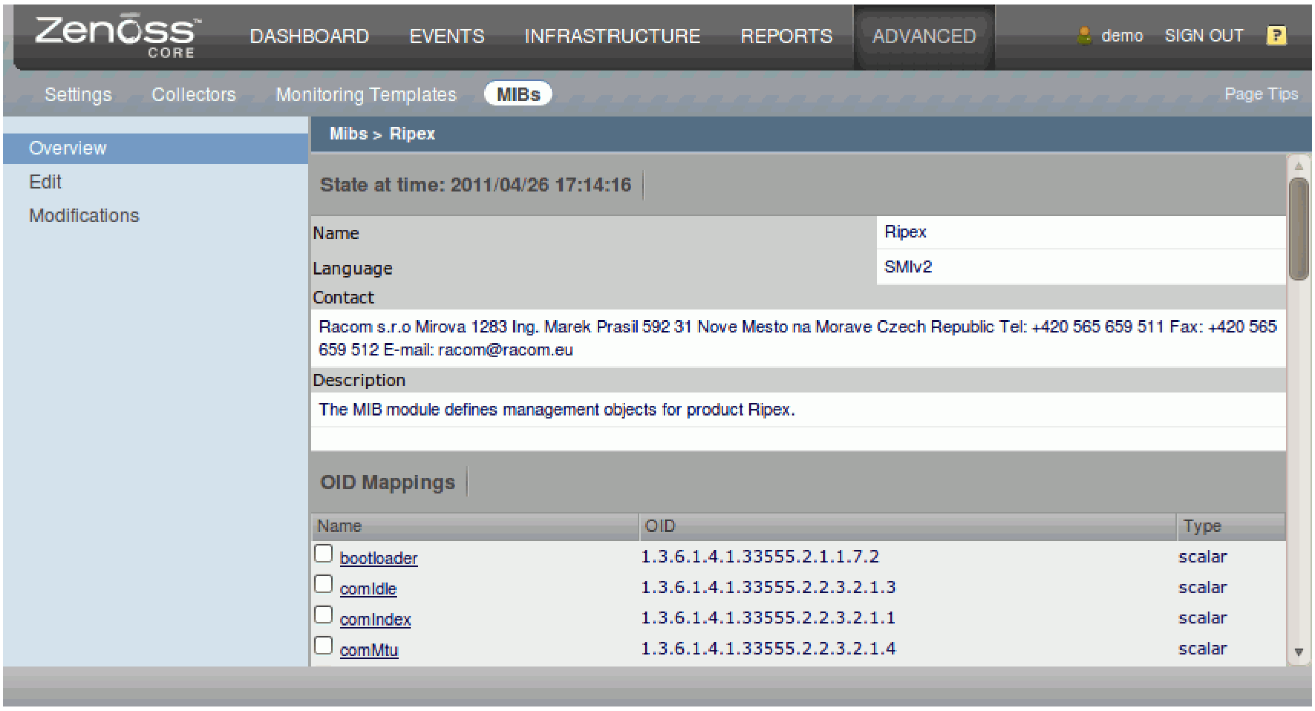
If possible, use RipEX Ethernet interface for SNMP communication to relieve the radio channel.

[3] http://www.racom.eu/eng/products/m/ripex/h-menu.html

2.2.4. Example of Zenoss settings for RipEX

These examples merely illustrate certain Zenoss settings. Refer to Zenoss manual for more information.

If you want to view MIB in Zenoss, go to **Advanced – MIBs**.



Mibs > Ripex

State at time: 2011/04/26 17:14:16

Name	Ripex
Language	SMIv2

Contact

Racom s.r.o Mirova 1283 Ing. Marek Prasil 592 31 Nove Mesto na Morave Czech Republic Tel: +420 565 659 511 Fax: +420 565 659 512 E-mail: racom@racom.eu

Description

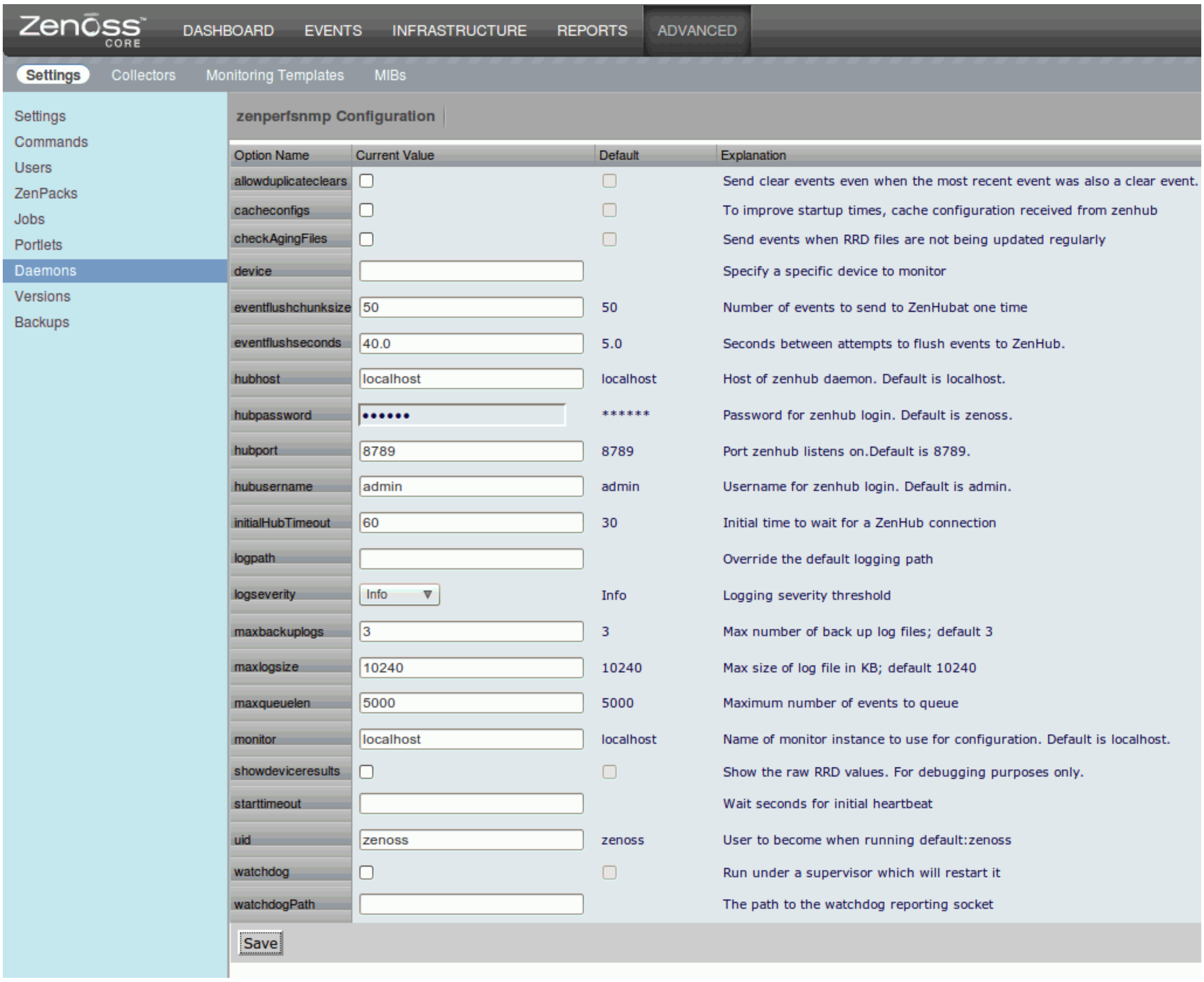
The MIB module defines management objects for product Ripex.

OID Mappings

Name	OID	Type
□ bootloader	1.3.6.1.4.1.33555.2.1.1.7.2	scalar
□ comIdle	1.3.6.1.4.1.33555.2.2.3.2.1.3	scalar
□ comIndex	1.3.6.1.4.1.33555.2.2.3.2.1.1	scalar
□ comMtu	1.3.6.1.4.1.33555.2.2.3.2.1.4	scalar

Fig. 2.2: RipEX MIB in Zenoss

To set global parameters of SNMP Managers in Zenoss select the **Edit config** command under **Advanced – Settings – Daemons – zenperfsnmp**.

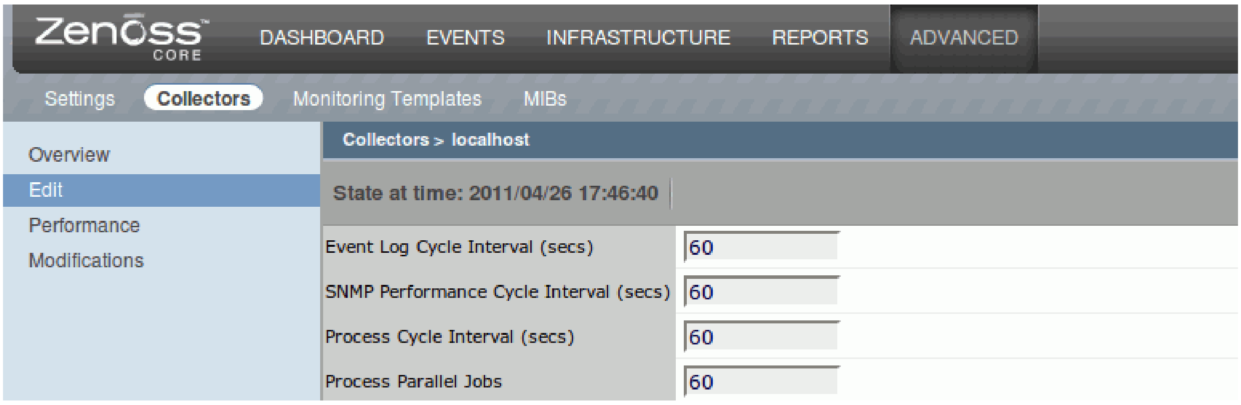


zenperfsnmp Configuration

Option Name	Current Value	Default	Explanation
allowduplicateclears	□	□	Send clear events even when the most recent event was also a clear event.
cacheconfigs	□	□	To improve startup times, cache configuration received from zenhub
checkAgingFiles	□	□	Send events when RRD files are not being updated regularly
device			Specify a specific device to monitor
eventflushchunksize	50	50	Number of events to send to ZenHubat one time
eventflushseconds	40.0	5.0	Seconds between attempts to flush events to ZenHub.
hubhost	localhost	localhost	Host of zenhub daemon. Default is localhost.
hubpassword	••••••	******	Password for zenhub login. Default is zenoss.
hubport	8789	8789	Port zenhub listens on.Default is 8789.
hubusername	admin	admin	Username for zenhub login. Default is admin.
initialHubTimeout	60	30	Initial time to wait for a ZenHub connection
logpath			Override the default logging path
logseverity	Info	Info	Logging severity threshold
maxbackuplogs	3	3	Max number of back up log files; default 3
maxlogsize	10240	10240	Max size of log file in KB; default 10240
maxqueuelen	5000	5000	Maximum number of events to queue
monitor	localhost	localhost	Name of monitor instance to use for configuration. Default is localhost.
showdeviceresults	□	□	Show the raw RRD values. For debugging purposes only.
starttimeout			Wait seconds for initial heartbeat
uid	zenoss	zenoss	User to become when running default:zenoss
watchdog	□	□	Run under a supervisor which will restart it
watchdogPath			The path to the watchdog reporting socket

Fig. 2.3: Setting SNMP Manager parameters in Zenoss

Other parameters, such as SNMP Performance Cycle Interval (secs) use the **Edit** command in **Advanced – Collectors – localhost**.



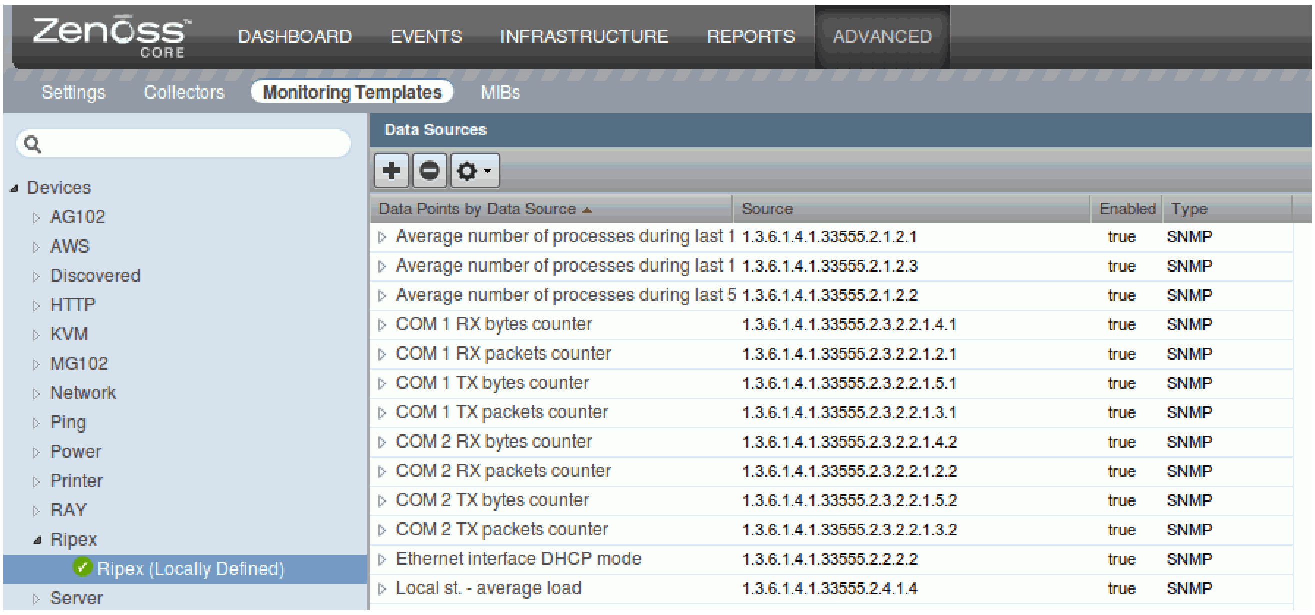
Collectors > localhost

State at time: 2011/04/26 17:46:40

Event Log Cycle Interval (secs)	60
SNMP Performance Cycle Interval (secs)	60
Process Cycle Interval (secs)	60
Process Parallel Jobs	60

Fig. 2.4: Setting SNMP Cycle Interval in Zenoss

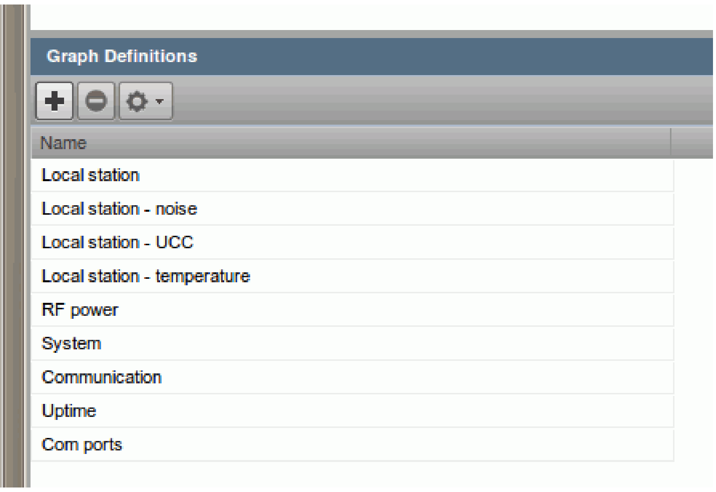
In this example, a RipEX template was created in Zenoss using Advanced – Monitoring Templates with individual OIDs from RipEX MIB.



Data Points by Data Source ▲	Source	Enabled	Type
Average number of processes during last 1	1.3.6.1.4.1.33555.2.1.2.1	true	SNMP
Average number of processes during last 1	1.3.6.1.4.1.33555.2.1.2.3	true	SNMP
Average number of processes during last 5	1.3.6.1.4.1.33555.2.1.2.2	true	SNMP
COM 1 RX bytes counter	1.3.6.1.4.1.33555.2.3.2.2.1.4.1	true	SNMP
COM 1 RX packets counter	1.3.6.1.4.1.33555.2.3.2.2.1.2.1	true	SNMP
COM 1 TX bytes counter	1.3.6.1.4.1.33555.2.3.2.2.1.5.1	true	SNMP
COM 1 TX packets counter	1.3.6.1.4.1.33555.2.3.2.2.1.3.1	true	SNMP
COM 2 RX bytes counter	1.3.6.1.4.1.33555.2.3.2.2.1.4.2	true	SNMP
COM 2 RX packets counter	1.3.6.1.4.1.33555.2.3.2.2.1.2.2	true	SNMP
COM 2 TX bytes counter	1.3.6.1.4.1.33555.2.3.2.2.1.5.2	true	SNMP
COM 2 TX packets counter	1.3.6.1.4.1.33555.2.3.2.2.1.3.2	true	SNMP
Ethernet interface DHCP mode	1.3.6.1.4.1.33555.2.2.2.2	true	SNMP
Local st. - average load	1.3.6.1.4.1.33555.2.4.1.4	true	SNMP

Fig. 2.5: Template for RipEX in Zenoss

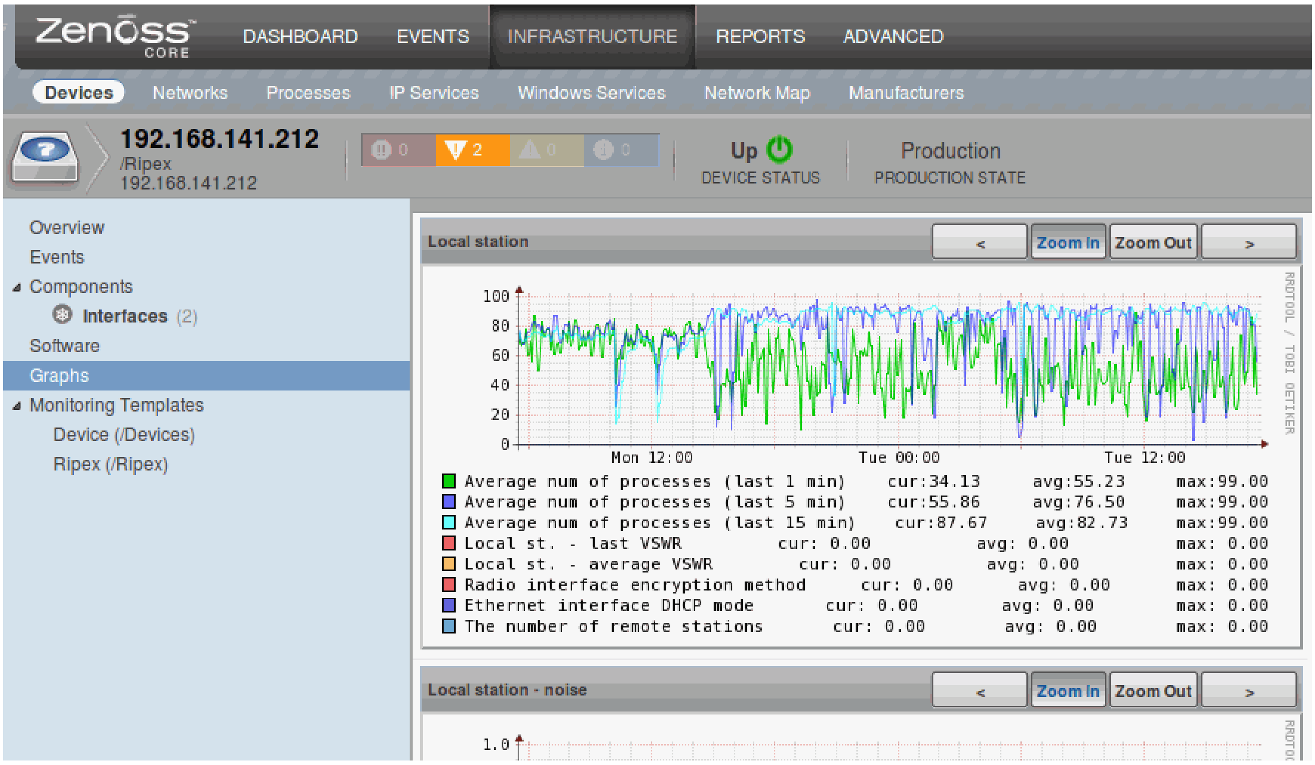
The individual OIDs can be grouped in graphs which depict the changes of monitored values over time. Thresholds can be defined for these values along with type (info, warning, error, etc.) as well as Event Class, etc.



Name
Local station
Local station - noise
Local station - UCC
Local station - temperature
RF power
System
Communication
Uptime
Com ports

Fig. 2.6: Graph settings in Zenoss

Afterwards you can assign the template created in this way to a specific device using **Infrastructure – Device**; it is then displayed in **Monitoring Templates** list and can be viewed using **Graphs** command.



```
🟩 Average num of processes (last 1 min)    cur:34.13     avg:55.23     max:99.00
🟦 Average num of processes (last 5 min)    cur:55.86     avg:76.50     max:99.00
🟦 Average num of processes (last 15 min)   cur:87.67     avg:82.73     max:99.00
🟥 Local st. - last VSWR          cur: 0.00         avg: 0.00         max: 0.00
🟧 Local st. - average VSWR          cur: 0.00         avg: 0.00         max: 0.00
🟥 Radio interface encryption method      cur: 0.00        avg: 0.00     max: 0.00
🟦 Ethernet interface DHCP mode         cur: 0.00        avg: 0.00       max: 0.00
🟦 The number of remote stations         cur: 0.00        avg: 0.00       max: 0.00
```

Fig. 2.7: Displaying graphs for a specific device in Zenoss

Zenoss also allows you to display current, average and maximum values in graphs.

3. Data speed and Modulations

On efficient use of narrowband radio channel

Introduction

The industrial narrowband *land mobile radio* (LMR) devices, as considered in this paper, have been the subject to European standard ETSI EN 300 113 [1]. The system operates on frequencies between 30 MHz and 1 GHz, with channel separations of up to 25 kHz, and is intended for private, fixed, or mobile, radio packet switching networks. Data telemetry, SCADA, maritime and police radio services; traffic monitoring; gas, water, and electricity producing factories are the typical system applications. Long distance coverage, high power efficiency, and efficient channel access techniques in half duplex operation are the primary advantages the system relays on. Very low level of adjacent channel power emissions and robust radio receiver architectures, with high dynamic range, enable for a system's co-existence with various communication standards without the additional guard band frequency intervals.

On the other hand, the strict limitations of the referenced standard as well as the state of the technology, has hindered the increase in spectrum efficiency, with which the system has used its occupied bandwidth. With its modification as well as with the new emerging specifications (ETSI EN 302 561 [2], ETSI EN 301 166 [3]) it is now possible for the up-to-date architectures of narrowband LMR devices to make the utilization of more efficient modes of system operation practically applicable.

The main objective of this paper is to describe the favorable properties of operational modes based on advanced nonlinear and linear digital modulation techniques in order to easy the decision on their usage and thus to help system integrators to increase the efficiency of the narrowband radio channel utilization allocated to the new generation of industrial LMR devices.

3.1. Narrowband radio transmitter

From the very advent of the radio transmission, it was evident that a radio device should not only use its occupied channel bandwidth effectively, but, in addition, should also avoid any unnecessary interference with other systems. Since then the frequency spectrum had been proving its importance and has become a scarce resource nowadays.

The narrowband radio devices under consideration are specified mostly by the European standard ETSI EN 300 113 [1]. Such radio equipments have to face challenging environmental and radio conditions all over the world. The dynamic range in the vicinity of 100 dB, very strict adjacent channel transmitted power attenuation requirements, high data sensitivity, adjacent channel selectivity, high level of radio blocking or desensitization and high co-channel rejection [1], are its most important radio characteristics to mention. It is no wonder that for such high dynamic range demands, super heterodyne transceiver architectures with a majority of analog components are still widely used. But yet the radio transceiver has to be small in dimensions, consumes low power and maintains all its parameters over the wide industrial temperature range and over extensive period of time for reasonable price. At the same time, it should provide enough flexibility to accommodate different channel bandwidths, digital modulation formats, data rates, and techniques, to combat negative effects of radio channel. From this point of view, the *software defined radio* (SDR) concept is, indisputably, a prospective alternative and has not been widely used by these systems. The rapid expansion of the digital signal processing, together with the advancements in signal analog-to-digital converters technology have, in recent years, made such projects economically feasible.

Today's LMR systems, being subject to [1], use mostly exponential constant envelope modulations GMSK, 2-CPFSK and 4-CPFSK. The application of the continuous phase modulations is mainly due to the extreme *adjacent channel transmitted power* (ACP) attenuation requirements, and inherent robustness against channel nonlinearities. Relatively simple implementation of non-coherent demodulators and synchronization algorithms also significantly contributes to the efficient channel usage, especially in packet-based switching networks. The systems thus maintain good power efficiency while the spectral efficiency reaches compromising values not exceeding 1 bit/s/Hz.

3.1.1. Digital modulation for narrowband channel

The prime classification of the digital modulation techniques into a *nonlinear* (or *exponential*) and *linear* modulation class is based on the way how the modulated signal has been generated. The complex modulation envelope of the linearly modulated signal such as M-PSK, M-QAM etc. can be described by a linear superposition of the properly filtered modulation impulses weighted by the information symbols. In case of the nonlinear modulation techniques, this general rule is valid only for the modulation signal which modulates the phase of the fundamental carrier signal. Thus the modulation process itself is nonlinear, exponential. The M-CPFSK in this case is recognized as a general class of nonlinear or exponential digital modulation with a continuous phase change.

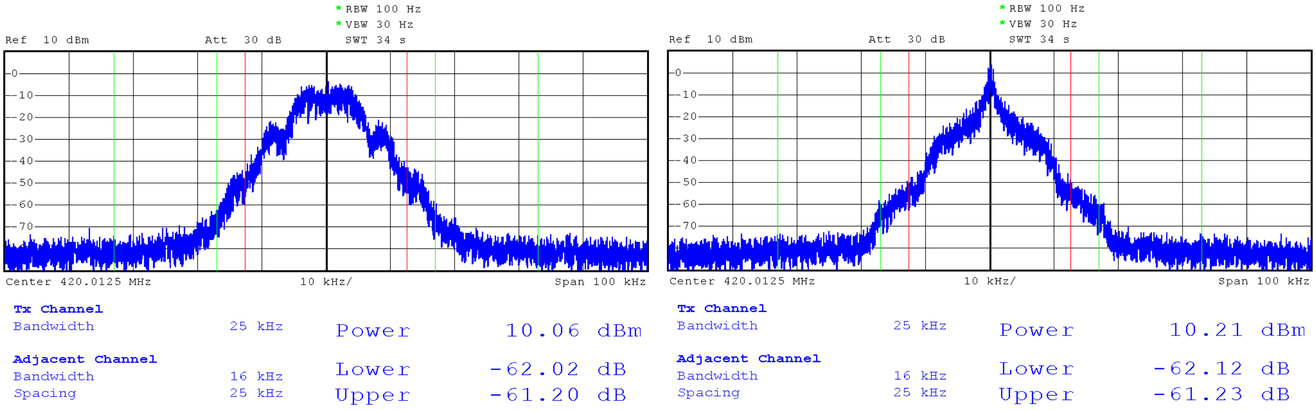
3.1.2. Adjacent channel power and spectrum efficiency

The adjacent channel power or *adjacent channel interference* (ACI) is that part of the total output power of a transmitter under defined conditions of modulation, which falls within a specified pass-band centred on the nominal frequency of either of the adjacent channels. This power is the sum of the mean power produced by the modulation, hum and noise of the transmitter. Adjacent channel power is usually referenced to the unmodulated carrier power [1]:

> *For a channel separation of 25 kHz, the adjacent channel power shall not exceed a value of **60 dB** below the transmitter power without the need to be below -37 dBm.*

It is interesting to note that, until 07/2007, the standard strictly demanded the adjacent channel power ratio of -70 dB.

The ACP parameter is particularly important in LMR systems, since it influences the density of the radio channels that can be used in a given area. Its value originated in the use of the traditional analog *frequency modulated* (FM) radio systems. Ironically, it was one of the main limitations for why those systems were – for many years – not able to utilize spectrally more efficient modulation schemes. The problem in this case is that all the advanced multi-level modulation techniques such as M-PSK, M-QAM, OFDM, CDMA or FBMCM have one negative property and that is a non-constant modulation envelope.



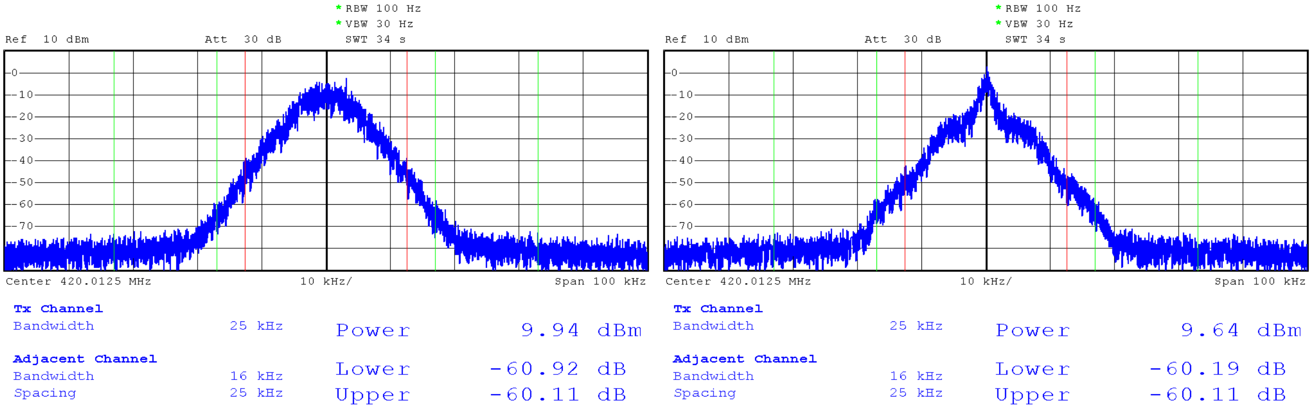
```
Tx Channel
Bandwidth          25 kHz    Power     10.06 dBm

Adjacent Channel
Bandwidth          16 kHz    Lower    -62.02 dB
Spacing            25 kHz    Upper    -61.20 dB
```

```
Tx Channel
Bandwidth          25 kHz    Power     10.21 dBm

Adjacent Channel
Bandwidth          16 kHz    Lower    -62.12 dB
Spacing            25 kHz    Upper    -61.23 dB
```

*Fig. 3.1: Modulated signal spectrums. (**left**) 2CPFSK with R=10.4 kBaud, modulation index h~0.6. (**right**) 2CPFSK with R=17.3 kBaud, modulation index h~0.2. 30 dB attenuator used in series.*

In the systems, where the transmitter power efficiency is of high importance, the *transmitter nonlinearity* also creates an important issue. Generally speaking, the higher the transmitter nonlinearity, the higher the transmitter efficiency can be reached. Unfortunately, the device with a nonlinear transfer function also tends to distort the spectrum of the transmitted signal, especially if the modulated signal exhibits the non-constant modulation envelope. In contrast, it is also true that only the non-constant envelope modulation can withstand a strict band limitation by means of modulation filtering – characterized by the roll-off parameter α in the following text. In other words, if the signal has a constant modulation envelope, it has an unlimited spectrum, and, if it has a band limited spectrum, it experiences the amplitude variations, which after passing through the nonlinear power amplifier, would be suppressed, but would also regenerate the side-lobes of the modulated signal spectrum. The phenomenon is known as the spectral *re-growth*, and it depends mainly on the three transmitter characteristics. Those are *peak to average power ratio* (PAPR) of the digital modulation scheme in use, *transmitter nonlinearity* and *the efficiency of the power amplifier linearization* or *pre-distortion technique* and all have to be considered when selecting the digital modulation technique for the system, where both power and spectrum are the key issues.

In light of these facts one can arrive at the conclusion that setting up the limit at −60 dB[1] rather than −70 dB was a reasonable step, while the initial limit has been left to be beyond the state of the present linearization technology for equipments production which in turn hindered the use of spectrally more efficient modulation techniques.



```
Tx Channel
Bandwidth              25 kHz     Power        9.94 dBm

Adjacent Channel
Bandwidth              16 kHz     Lower      -60.92 dB
Spacing                25 kHz     Upper      -60.11 dB
```

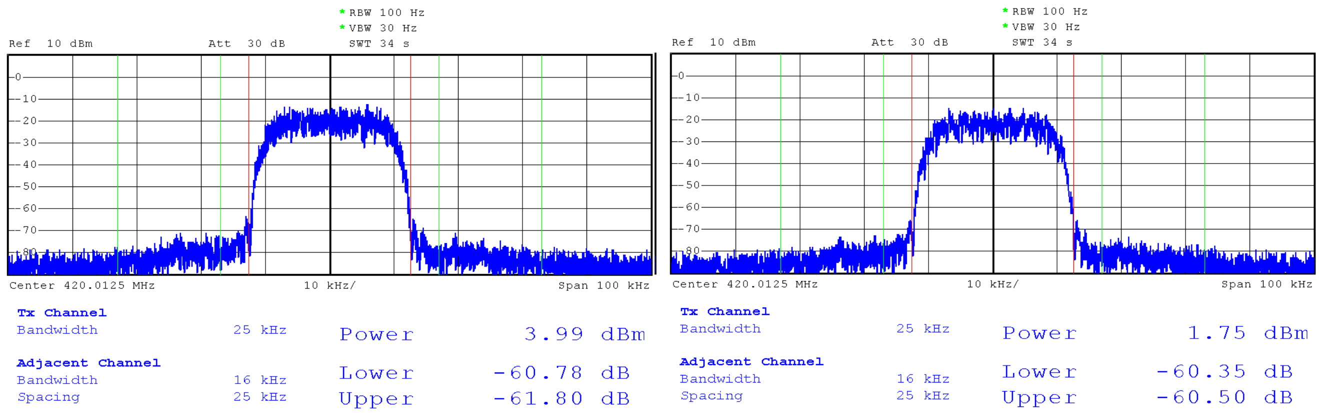
```
Tx Channel
Bandwidth              25 kHz     Power        9.64 dBm

Adjacent Channel
Bandwidth              16 kHz     Lower      -60.19 dB
Spacing                25 kHz     Upper      -60.11 dB
```

*Fig. 3.2: Modulated signal spectrums. (**left**) 4CPFSK with R=10.4 kBaud, modulation index h~0.3. (**right**) 4CPFSK with R=17.3 kBaud, modulation index h~0.1.*

3.1.3. Transmitter power efficiency

In this section, the measurement results concerning the overall narrowband transmitter power efficiency are presented. It is no ambition however, to provide exact power efficiency analysis of the particular high power amplifie~~r~~ with the selected linearization circuit proceeded. It is rather to give the example of the practically achievable overall transmitter power efficiencies and to show the differences related to selected digital modulation formats of each of the linear/nonlinear class.

[1] The standard [2] specifying the conformity testing for TETRA-like devices allows -55 dBc in normal or -50 dBc in extreme temperature conditions, assuming channel separation of 25 kHz.



```
Tx Channel
Bandwidth            25 kHz     Power      3.99 dBm

Adjacent Channel
Bandwidth            16 kHz     Lower    -60.78 dB
Spacing              25 kHz     Upper    -61.80 dB
```

```
Tx Channel
Bandwidth            25 kHz     Power      1.75 dBm

Adjacent Channel
Bandwidth            16 kHz     Lower    -60.35 dB
Spacing              25 kHz     Upper    -60.50 dB
```

*Fig. 3.3: Modulated signal spectrums. (**left**) π/4-DQPSK with R=17.3 kBaud, (**right**) 16-DEQAM with R=17.3 kBaud.*

As for the linear modulation techniques, the differentially encoded formats π/4-DQPSK, D8PSK and 16-DEQAM have been selected and tested mainly due to their low modulation envelope variations and inherent robustness against negative effects of signal propagation through the narrowband radio channel.

The 2CPFSK and 4CPFSK have been selected from the nonlinear modulation class. There is one particular parameter of high importance essentially influencing the characteristics of these modulation formats and that is a *modulation index*. It expresses the relation between the modulation rate and the maximum frequency deviation according to simple rule (1.1)

$$h = \frac{2\Delta f}{R(M-1)}, \qquad (1.1)$$

where *R* is the modulation rate, *M* is the number of modulation states and *Δf* is the maximum frequency deviation representing the outermost symbol frequency position. The selection of the modulation index in most practical applications of narrowband LMR has been driven by compromising requirements between the modulation rate, receiver sensitivity and adjacent channel power level. Its value usually converges to *1/M* with a well known example of MSK, particularly GMSK where M=2, thus h=0.5 as the lowest value needed to maintain an orthogonal signaling. In order to compare the modulation formats at the same spectrum efficiency we also measured the properties of 2CPFSK and 4CPFSK modulations with very low modulation index resulting in use of high symbol rate of 17.3 kBaud.

The examples of transmitted signal spectrums can be seen in Figure. 3.1 to Figure. 3.3. It is interesting to note the degradation of the signal spectrum with increased symbol rate in case of 2CPFSK and 4CPFSK that implicitly points out that the assigned bandwidth is not used effectively. It can be seen that the significant amount of the signal power is concentrated within the close vicinity of the carrier frequency and thus it results in poor ratio between the occupied signal bandwidth and the noise bandwidth of radio receiver (Table 3.1).

Tab. 3.1: Measurement results of the transmitter parameters for selected modes of operation.

Modulation Format	Symbol Rate	Modul. Parameter	P_{out}	ACI Lower	 Upper	Occupied Bandwidth @ 99.9%	P_{IN}	η_{TX}	Spectrum plot
[-]	[kBaud]	[-]	[dBm]	[dBc]	[dBc]	[kHz]	[W]	[%]	[-]
2CPFSK	10.4	h=0.6, α=0.28	40	-62	-61	19.8	35	29	Fig. 3.1
	17.3	h=0.2, α=0.28	40	-62	-61	16.6	35	29	Fig. 3.1
4CPFSK	10.4	h=0.3, α=0.28	40	-61	-60	19.6	35	29	Fig. 3.2
	17.3	h=0.1, α=0.28	40	-61	-60	17.2	35	29	Fig. 3.2
π/4-DQPSK	17.3	α=0.4	35	-61	-62	22.0	22.8	14	Fig. 3.3
D8PSK	17.3	α=0.4	35	-61	-61	22.0	22.8	14	-
16-DEQAM	17.3	α=0.4	35	-60.5	-60.5	22.0	20.4	10	Fig. 3.3
Measurement uncertainty ±2 dB									

The measurement values of achievable output power P_{out}, amount of adjacent channel interference *ACI* and overall transmitter power efficiency η_{TX} are collectively given for all the modulation formats in Table 3.1. It can be seen that the ACI limit (-60 dBc) is maintained for all of these settings; however, there are two penalties in case of linear modulation schemes that typically have to be paid for higher spectrum efficiency. Firstly, it is the lower output power level achievable. For this specific transmitter architecture it is in particular 35 dBm @ π/4-DQPSK, D8PSK and 33 dBm @ 16-DEQAM. Secondly, it is the lower value of the overall transmitter power efficiency reached. Comparing to exponential modes of system operation the efficiency of linear operational modes has decreased to 14% and 10%. Despite this negative trend, the achieved values of output power exceeding 3 W, and 2 W respectively, are considered practically applicable for next generation of narrowband LMR devices and as it will be shown in the next section they enable the system to use its occupied bandwidth with even higher communication efficiency.

3.2. Narrowband radio receiver

The most important parameters which describe the quality of narrowband radio receiver are *maximum usable (data) sensitivity, co-channel rejection, adjacent channel selectivity, desensitization* and *intermodulation response rejection*. Besides the maximum usable sensitivity, all other receiver parameters can be classified as the measures of the *receiver degradation parameters* used to analyze the degradation of its performance due to the presence of unwanted (interfering) signals. Although there is a strong relation between all of these parameters, in this paper the attention is given only to the first of them, to the maximum usable sensitivity in particular.

According to [1], the maximum usable data sensitivity is the minimum level of the signal (emf) at the receiver input, produced by a carrier at the nominal frequency of the receiver, modulated with a normal test signal, which will, *without interference*, produce, after demodulation, a data signal with a specified *bit-error-ratio* (BER) of 10^{-2} or a specified *successful message ratio* (SMR) of 80%.

> *The maximum usable sensitivity shall not exceed an electromotive force of **3.0 dBμV** under normal test conditions.*

Assigning this value as *S*, one can also express what *signal-to-noise ratio* (SNR) can be expected in relation to *noise figure* (NF) and transformed to the receiver input

$$\mathrm{SNR} = S - (10.\log(kT) + 10.\log(B_N) + \mathrm{NF}) \quad [\mathrm{dB}]. \qquad (2.1)$$

In (2.1), *k* is the Boltzmann's constant, *T* is the absolute temperature in Kelvin and B_N is the receiver noise bandwidth of e.g. 25 kHz.

3.2.1. Maximum usable data sensitivity

In this section, the results of maximum usable data sensitivity measurement (Figure 3.4) for the complete narrowband radio transceiver are presented. All the results are given for 25 kHz channel separation.

Firstly, let us focus on operational modes with exponential modulations, Figure 3.4. It can be seen that the *emf* sensitivity limit of +3 dBμV (-110 dBm @ 50 Ω) is fulfilled with margin for both modulations (2CPFSK, 4CPFSK) when running at the symbol rate of 10.4 kBaud. When higher symbol rates are selected, these modulations loss their power efficiency rapidly and for the selected symbol rate of 17.3 kBaud, the sensitivities lower down to the values of −107 dBm @ BER=10^{-2} and −102 dBm @ BER=10^{-2} for 2CPFSK and 4CPFSK respectively. This discrepancy is caused mainly due to the fact that there is a significantly lower frequency deviation used at the higher symbol rates. The decrease in power efficiency with increasing spectrum efficiency is not linear as for the typical linear modulations. Although possible, this example documents that the increase in spectrum efficiency of exponential modulation techniques cannot be considered for efficient use of assigned bandwidth.

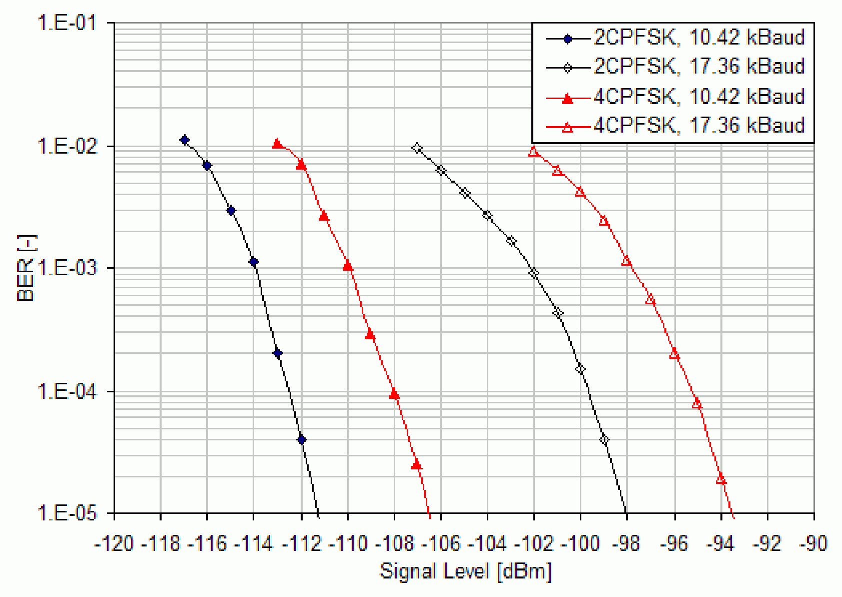


Fig. 3.4: Maximum usable sensitivity measurement results for different settings of exponential modulations.

The second set of measurement results, presented in Figure 3.5, documents the power efficiency analysis of operational modes based on the linear modulation techniques. It can be seen that when using the linear π/4-DQPSK, the radio receiver can still reach the data sensitivity limit even for 17.3 kBaud with a 2 dB margin. Even from this comparison it is evident that the π/4-DQPSK mode of operation outperforms the 4-CPFSK at higher spectrum efficiencies. Further increase in spectrum efficiency can be reached by higher order constellations such as D8PSK and 16DEQAM and the radio receiver can still maintain practically applicable sensitivities of −107 dBm @ BER=10^{-2} and −105 dBm @ BER=10^{-2} respectively.

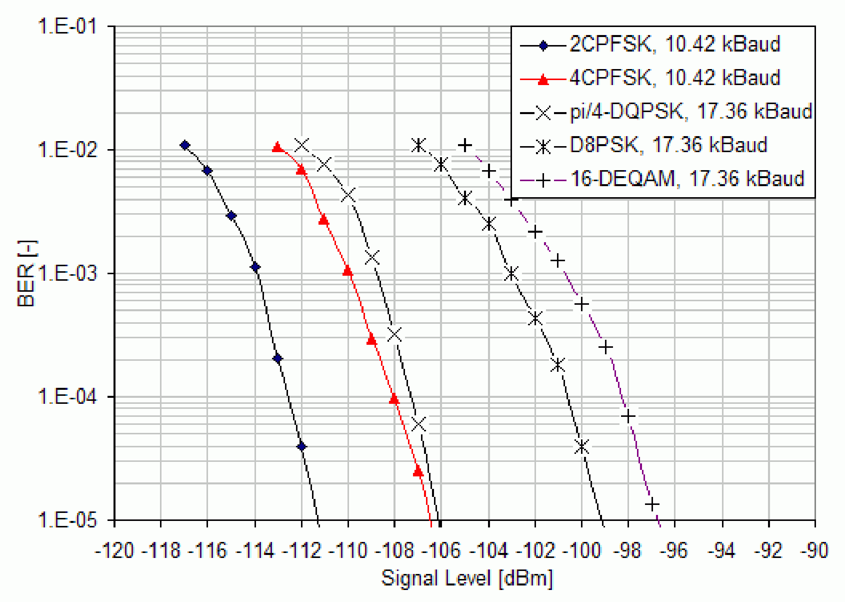


Fig. 3.5: Maximum usable sensitivity measurement results. Channel separation 25 kHz.

3.2.2. Efficient use of narrowband radio channel

As it has been written in the Section 1, the radio transceiver in exponential modulation mode can make use of higher transmitter power. In order to take this fact into account the *system gain (SG)* or the *maximum allowed path loss* (2.2)

$$SG\ [dB] = P_{out} - S\ , \qquad (2.2)$$

is usually calculated for the wireless communication systems. Here the P_{out} is the available transmitter power expressed in dBm and *S* is the measured value of radio receiver sensitivity, also in dBm. It expresses the referential value of the link budget, assuming 0 dBi of antennas gain and together with the *spectrum efficiency* given by (2.3) it expresses how effectively the radio device uses its assigned bandwidth

$$\eta\ [bit/s/Hz] = \frac{R_b}{B}\ . \qquad (2.3)$$

In (2.3), the R_b is the *raw bit rate* given in [bits/s] and *B* is the frequency bandwidth assigned to the radio system, 25 kHz in particular.

All these performance characteristics are collectively given in Table 3.2. It can be seen that even with the lower available transmitter power, the radio transceiver can reach wider system gain at higher spectrum efficiencies while running in linear as oppose to the exponential modulation mode. On the other hand, if the long distance coverage is of the primary application concern, even the 2CPFSK modulation having spectrum efficiency of 0.4 bit/s/Hz, but the system gain of impressive, 157 dB, can be a reasonable option.

Tab. 3.2: Overall performance characteristics of the narrowband radio transceiver for selected modes of operation.

Modulation Format	Modul. Param.	Symbol Rate	Raw Bit Rate	Spectrum Efficiency	Data Sensitivity @ BER 10^{-2}	Available Output Power	System Gain
[-]	[-]	[kBaud]	[kbits/s]	[bit/s/Hz]	[dBm]	[dBm]	[dB]
2CPFSK	h=0.6, α=0.28	10.42	10.42	0.42	-117	40	157
	h=0.2, α=0.28	17.36	17.36	0.69	-107	40	147
4CPFSK	h=0.3, α=0.28	10.42	20.83	0.83	-113	40	153
	h=0.1, α=0.28	17.36	34.72	1.39	-102	40	142
π/4-DQPSK	α=0.4	17.36	34.72	1.39	-112	35	147
D8PSK	α=0.4	17.36	52.08	2.08	-107	35	142
16-DEQAM	α=0.4	17.36	69.44	2.78	-105	33	138
Measurement uncertainty ±2 dB							

3.3. Conclusion

As it was shown in this paper, the strict limits of the referenced standard as well as the state of the technology hindered increasing the communication efficiency with which the narrowband systems have been using the occupied frequency bandwidth. The key limiting factor that has been identified was the limit of adjacent channel power attenuation. Lessening the requirement from -70 dBc to -60 dBc in 2007 has opened up the closed door for implementation of linear digital modulation techniques. However, as it has been shown in later sections, a reasonable use of the exponential modulation can be still beneficial for these systems. Based on the results presented, the most important concluding notes can be seen in the following:

- When the long distance coverage as well as the overall power efficiency are of the primary application concern, the use of exponential modulation techniques 2CPFSK and 4CPFSK at relatively low symbol rates e.g 10.4 kBaud can be the recommended option. In this case, the nonlinear modulation techniques can make use of higher frequency deviation and increase the system gain by outstanding values of receiver sensitivities. At the 10 W of output power the system gain of 157 dB and 153 dB for 2CPFSK and 4CPFSK modulation techniques respectively can be expected.

- When higher symbol rates are selected, the exponential modulation techniques lose their power efficiency (and their main advantage) significantly. Further increase of the exponential modulation spectrum efficiency from the values currently being used by the narrowband systems (up to 1 bit/s/Hz) can be therefore considered inefficient.

- From all the modulation formats studied, the π/4-DQPSK can provide the narrowband LMR system with communication efficiency closest to the optimal communication systems. The proposed solution based on this modulation technique can reach the spectrum efficiency of up to 1.5 bit/s/Hz. The data sensitivity limit required by [1] can also by fulfilled with margin of 2-3 dB, resulting in the system gain of 147 dB.

- For applications where higher data throughputs are needed the additional increase in spectrum efficiency can be gained by D8PSK and 16-DEQAM modulation formats. However, compared to π/4-DQPSK, an increase in overall communication efficiency cannot be expected, while there is the inevitable penalty in power efficiency characteristic.

References

[1] ETSI EN 300 113-1 V1.6.2 (2009-11), Electromagnetic compatibility and Radio spectrum Matters (ERM), Part 1: Technical characteristics and methods of measurement. European Standard. ETSI, 11/2009.

[2] ETSI EN 302 561 V1.2.1 (2009-12), Electromagnetic compatibility and Radio spectrum Matters (ERM), Land Mobile Service; Radio Equipment using constant or non-constant envelope modulation operating in a channel bandwidth of 25 kHz, 50 kHz, 100 kHz or 150 kHz; Harmonized EN covering essential requirements of article 3.2 of the R&TTE Directive. European Standard. ETSI, 12/2009.

[3] ETSI EN 301 166-1 V1.3.2 (2009-11), Electromagnetic compatibility and Radio spectrum Matters (ERM), Part 1: Technical characteristics and methods of measurement. European Standard. ETSI, 11/2009.

4. Autospeed

Normally all radio modems in a network have to transmit with the same data rate on the same radio channel. The Autospeed feature of RipEX enables different speeds to be used simultaneously in a radio modem network.

The following picture gives an example of a network layout. Let us assume, that all signals are strong enough to ensure almost perfect operation:

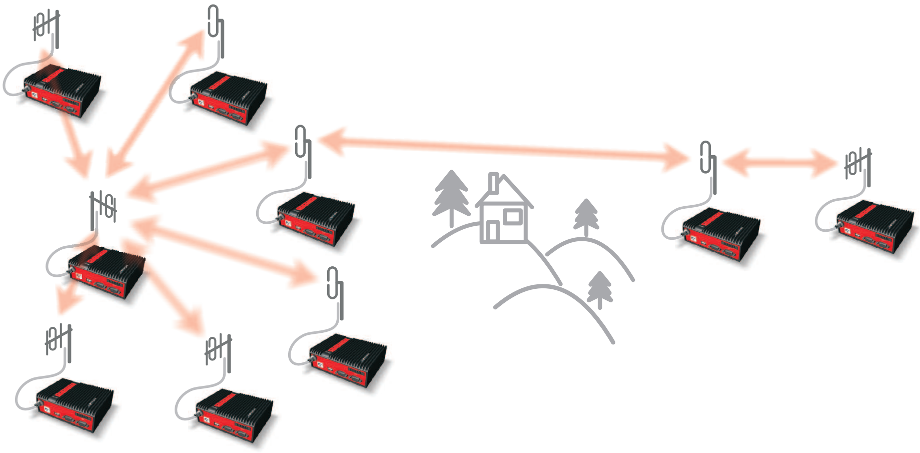


Fig. 4.1: Autospeed - initial situation

After some time situation changes and path loss on one of these links significantly increases, rendering the communication unreliable:

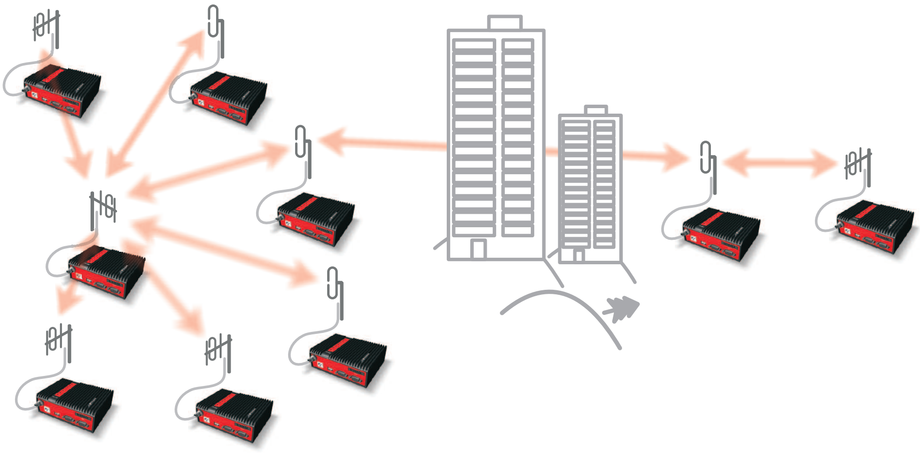


Fig. 4.2: Autospeed - problem

What can we do:

- Change antennas on one or both sides of the link
- Use higher masts on one or both sides of the link
- Build additional repeater(s)
- Lower the data rate significantly to increase the system gain

The first three possibilities require time and money, i.e. additional investment. The fourth possibility (when applied to whole network, as it normally is the case) would slow down the response time (two to four times) of the whole network, quite probably making it unusable for the application. RipEX Autospeed feature allows to change the transmission data rate at the affected radios only, the rest of the network may continue in full speed. Consequently the overall performance of network is maintained practically at the same level while no additional investment is required. More over, the whole fix can be done in minutes from behind a web-browser screen while sitting in your office.

Of course a similar scenario can be used right from the moment of planning a new network. The investment cost can be reduced by purposefully configuring the few „difficult“ radio links to a lower data rate.

The above scenarios are made possible by the unique capability of RipEX to automatically adjust its receiver to the data rate of the incoming frame. Note that when an ACK frame is sent by the receiving RipEX, it always uses the same data rate as the frame it acknowledges. The only limitation of this feature is that all the frames have to have the same symbol rate and the same principle of modulation (i.e. CPFSK or linear).

Modulation types which can be combined within one approval type (FCC , CE or CE-LBT):

2CPFSK & 4CPFSK & 8CPFSK with or without FEC
or
D2PSK & Pi/4DQPSK & D8PSK &16DEQAM with or without FEC

The improvement in system gain value using this technique may be more than 15 dB. Increasing gain of antenna system by that value would be impractical, often impossible – the „difficult“ hops are designed to use high-gain directional antennas from the beginning. Hence the Autospeed may make a radio modem network the optimum choice in situations where it could not be economically feasible before.

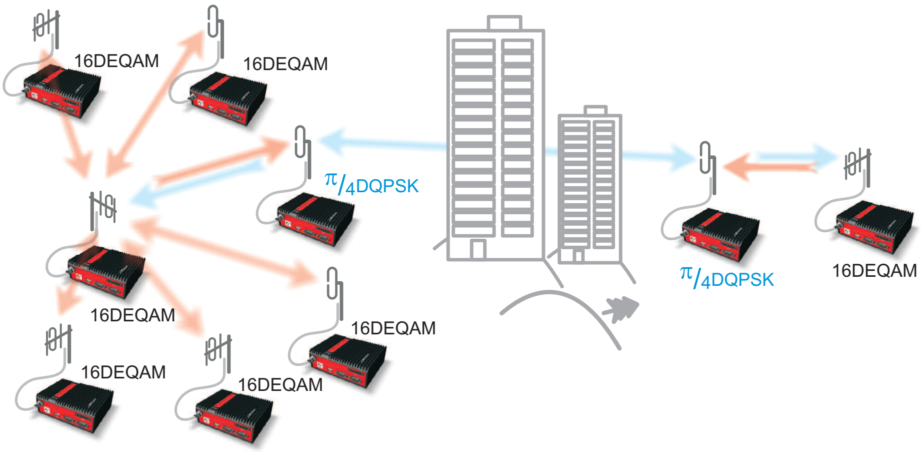


Fig. 4.3: Autospeed - solution

5. Back-to-Back repeater

This layout and settings may be used if you need to operate different parts of the radio network on different frequencies. Connection between these two parts is realised by Back2Back connection between two RipEX's (hereafter referred to as border RipEX's), each of which operates on different frequency.

5.1. Back to Back in Bridge mode

Ethernet

If end devices are connected to RipEX's over Ethernet, border RipEX's can be connected with an Ethernet cable. IP addresses of all RipEX's as well as connected devices must be within the same LAN. Ethernet interfaces must be interconnected for proper function of remote service access.

COM

If end devices are connected to RipEX's over COM interface, one (any of the two) COM port of a border RipEX must be connected to a COM port of the other border RipEX using RS232 crossover cable or null modem. Communication parameters of both connected ports must be set to the same values, we recommend using the highest available speed.

Important

Border RipEX's should be interconnected via one COM port only, connecting both COM ports would create a loop.

Limitation: If a device is connected to the free COM port of a border RipEX, it only sends data to its part of the radio network. Data from all other COM ports of other RipEX's throughout the entire network will be delivered to both COM ports of all other RipEX's.

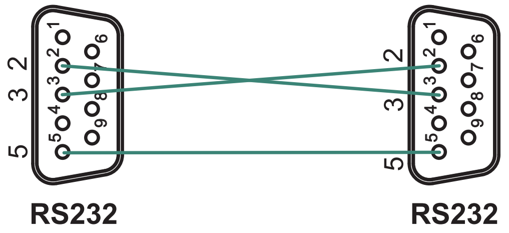


Fig. 5.1: Crosslink serial cable

Ethernet + COM

If end devices are connect to RipEX's both over Ethernet and COM ports, or if you require remote access to a network which uses COM ports, border RipEX's must be interconnected both via Ethernet (see 1.1) and COM (see 1.2).

5.2. Back to Back in Router mode

In Router mode border RipEX's are interconnected by Ethernet cable. Routing in both parts of the network must be set up so that communication passes through the Ethernet interface of the border RipEX's. We recommend splitting both radio networks to two separate LAN networks.

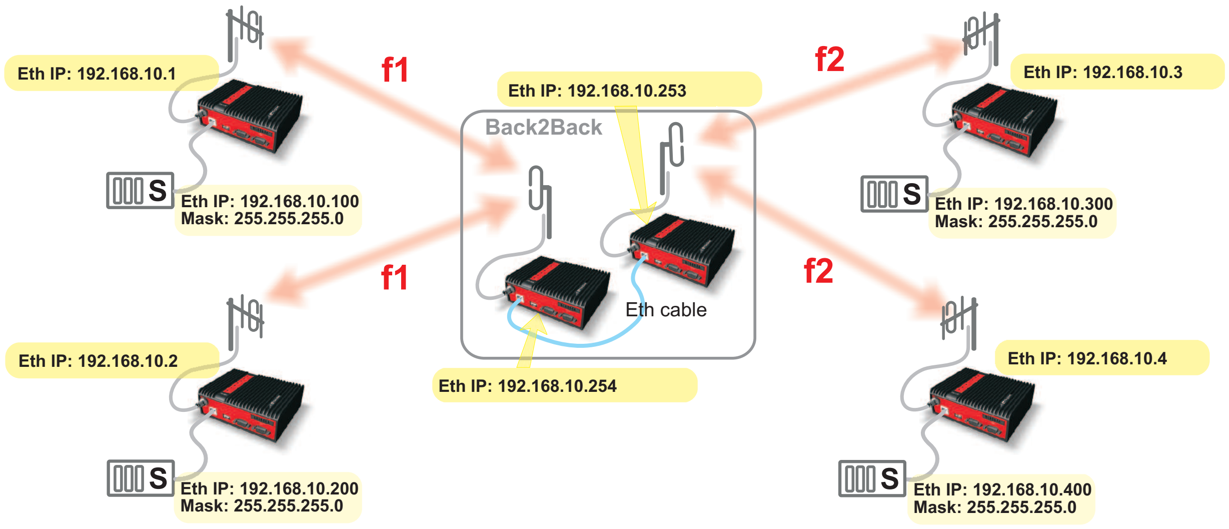


Fig. 5.2: Back2Back in bridge mode

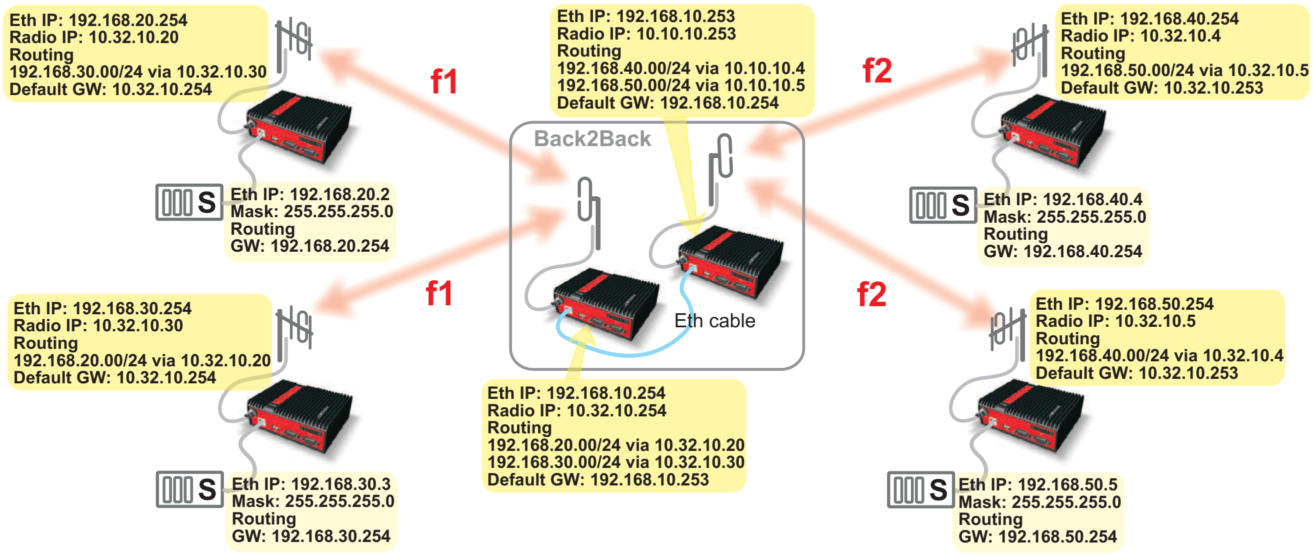


Fig. 5.3: Back2Back in router mode

6. Combining MORSE and RipEX networks

When expanding a MORSE network with RipEX radio modems, different arrangements are possible. In the following paragraphs we assume that the whole network is divided into two parts – the MORSE part and the RipEX part. The two parts are interconnected through two radio modems – one MRxxx and one RipEX, hereafter referred to as border radio modems. As RipEX and MRxxx radio channel protocols are not compatible, we strongly recommend you use different frequencies for either part of the network.

6.1. RipEX part in Bridge mode

There are two basic scenarios:

- Terminal devices are connected to Ethernet interface
- Terminal devices are connected to COM port

6.1.1. Terminal devices connected over Ethernet

If terminal devices are connected over Ethernet, the border RipEX and MRxxx should also be interconnected by an Ethernet cable. The IP addresses of all devices in the network should belong to a single LAN.

The picture shows MORSE network settings; note the use of Proxy ARP in IP-M-IP mode.

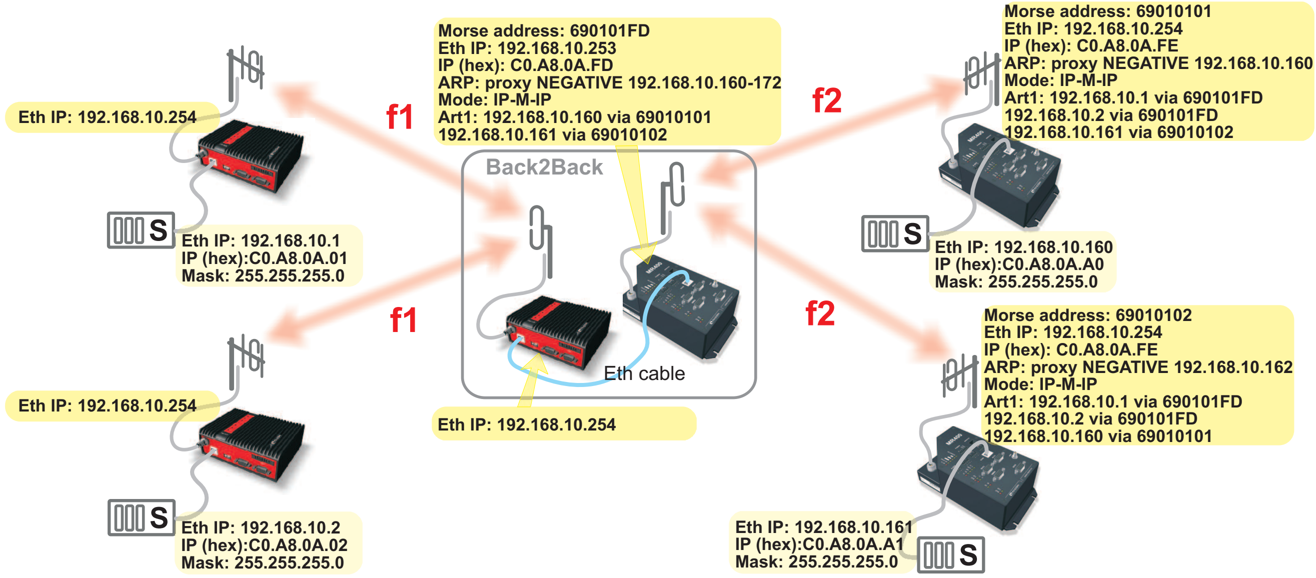


Fig. 6.1: RipEX - MR400 in Bridge mode

6.1.2. Terminal devices connected to COM

The COM port of the border RipEX and the RS232 of the border MRxxx are connected with a crosslink serial cable, see Fig. 6.2, “Crosslink serial cable”.

The COM port protocol at the border MRxxx must be the same as protocol used by the other MORSE devices in the network. In some special cases, the ASYNC LINK protocol can be used for the border interconnection.

If the Master is located on the side of the MRxxx, the border MRxxx should be set to Slave. Depending on the SCC interface used the MRxxx should use Multiaddressing with addresses of all the Slave units on the RipEX network.

If the Master is located on the side of the RipEX, the border MRxxx is set like it was connected to the Master and the Node of the connected SCC interface has to correspond to the Master's address.

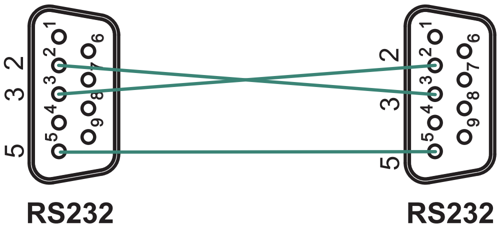


Fig. 6.2: Crosslink serial cable

6.2. RipEX in Router mode

There are two basic scenarios:

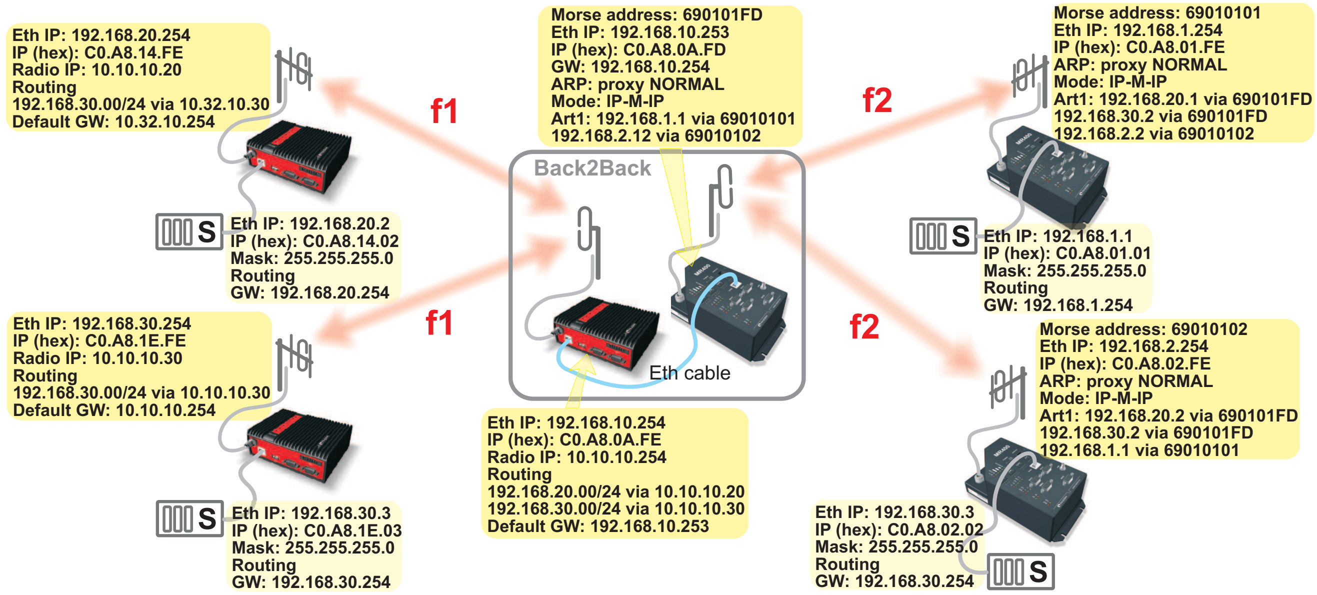
- Terminal devices are connected over Ethernet
- Terminal devices are connected over COM interface

6.2.1. Terminal devices connected over Ethernet

In this scenario the border RipEX and MRxxx should be interconnected with an Ethernet cable.

Routing in both parts of the network should be set up so that communication between them is channeled over the border modems. It is recommended that terminal devices in the two parts of the network are located on separate LAN's.

The picture shows MORSE network settings.



Eth IP: 192.168.20.254
IP (hex): C0.A8.14.FE
Radio IP: 10.10.10.20
Routing
192.168.30.00/24 via 10.32.10.30
Default GW: 10.32.10.254

Eth IP: 192.168.20.2
IP (hex): C0.A8.14.02
Mask: 255.255.255.0
Routing
GW: 192.168.20.254

Eth IP: 192.168.30.254
IP (hex): C0.A8.1E.FE
Radio IP: 10.10.10.30
Routing
192.168.30.00/24 via 10.10.10.30
Default GW: 10.10.10.254

Eth IP: 192.168.30.3
IP (hex): C0.A8.1E.03
Mask: 255.255.255.0
Routing
GW: 192.168.30.254

Morse address: 690101FD
Eth IP: 192.168.10.253
IP (hex): C0.A8.0A.FD
GW: 192.168.10.254
ARP: proxy NORMAL
Mode: IP-M-IP
Art1: 192.168.1.1 via 69010101
192.168.2.12 via 69010102

Eth IP: 192.168.10.254
IP (hex): C0.A8.0A.FE
Radio IP: 10.10.10.254
Routing
192.168.20.00/24 via 10.10.10.20
192.168.30.00/24 via 10.10.10.30
Default GW: 192.168.10.253

Morse address: 69010101
Eth IP: 192.168.1.254
IP (hex): C0.A8.01.FE
ARP: proxy NORMAL
Mode: IP-M-IP
Art1: 192.168.20.1 via 690101FD
192.168.30.2 via 690101FD
192.168.2.2 via 69010102

Eth IP: 192.168.1.1
IP (hex): C0.A8.01.01
Mask: 255.255.255.0
Routing
GW: 192.168.1.254

Morse address: 69010102
Eth IP: 192.168.2.254
IP (hex): C0.A8.02.FE
ARP: proxy NORMAL
Mode: IP-M-IP
Art1: 192.168.20.2 via 690101FD
192.168.30.2 via 690101FD
192.168.1.1 via 69010101

Eth IP: 192.168.30.3
IP (hex): C0.A8.02.02
Mask: 255.255.255.0
Routing
GW: 192.168.30.254

Fig. 6.3: RipEX - MR400 in Router mode

6.2.2. Terminal devices connected to COM

A MORSE network can only be expanded with RipEX modems if the application protocol is supported both by MORSE and RipEX, or if RipEX's UNI protocol can be used instead. If you want to use protocols which are not implemented in RipEX by default, please consult RACOM's technical support.

The COM port of the border RipEX and the RS232 of the border MRxxx are connected with crosslink serial cable, see Fig. 6.2, “Crosslink serial cable”.

If the Master is located in the MORSE part of the network, the border MRxxx should use Multiaddressing for addresses of all Slaves in the RipEX network. Protocol settings should reflect that. The border RipEX then should be set up as connected to the Master using the appropriate protocol (address translation using a mask or table, routing rules).

If the Master unit is located on the RipEX side of the network, rules for address translation should direct all the packets sent to Slave units of the MORSE network to the COM port connected to the border MRxxx. This COM port should then use an appropriate protocol in Slave mode. In the border RipEX the timeout for response from technology should be extended from 500 ms to several seconds (the response time will depend on the size of the MORSE network) – this parameter can only be set in CLI. On the MORSE side, the protocol should be set to Master.

7. Profibus

Radio modem RipEX supports the most widely spread Profibus (Process Field Bus) type designated Profibus DP (Decentralized Periphery) type 0 (see http://www.profibus.com/technology/profibus/).

Profibus DP is designed for fast master–slave communication. The central master unit communicates with the remote slaves using RS485 bus. They are typically connected by twisted pair cabling. The cable length between two RS485 repeaters is limited (from 100 to 1200 m), depending on the bit rate used. The RipEX Profibus DP implementation allows for RS485 to be replaced by radio network, either partially or entirely. This significantly increases the potential distance between the individual nodes or even enables you to get rid of cable links altogether.

7.1. Bridge and Router modes

RipEX operates in two basic modes, Bridge and Router. Network topology determines which one is the more suitable for your specific application (see chapter RipEX in detail[1] of the manual).

Apart from network layouts designed in this manual, we also recommend using Router mode if alongside the central RipEX some PLC Slaves are also connected to the PLC Master over RS485 while others connect over the radio network.

This is because in Bridge mode RipEX would broadcasts to radio channel each packet received on RS485. This could cause slower communication in some situations, and even collisions when a repeater is used. In Router mode only the packets destined for remote PLC Slaves are broadcast over the radio channel while packets sent to the PLC Slaves connected directly over RS485 are ignored.

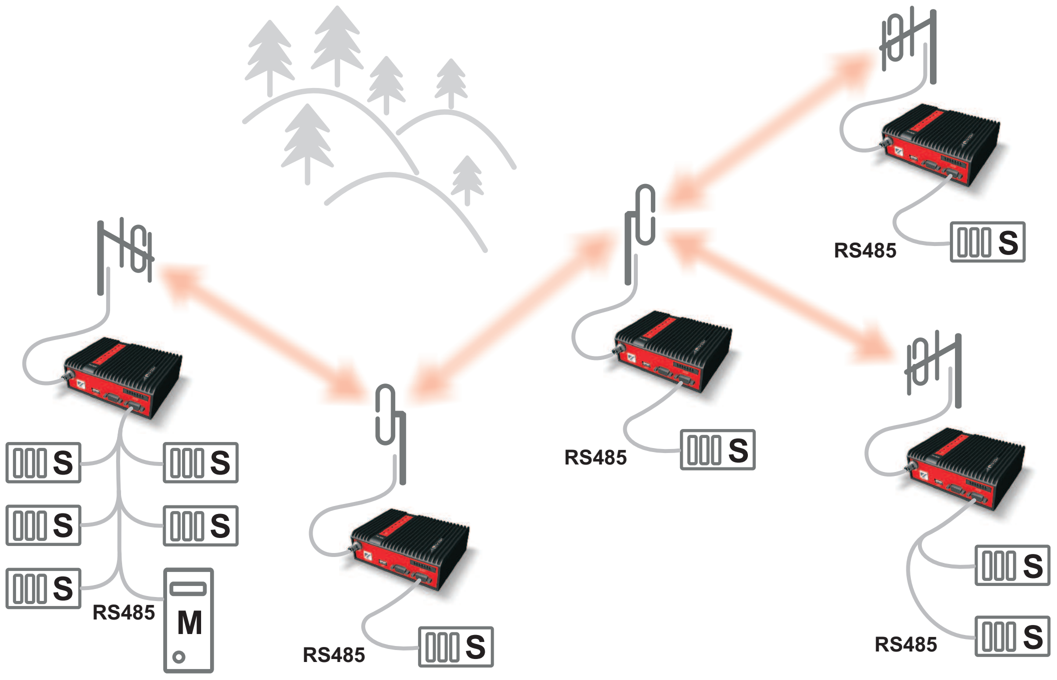


Fig. 7.1: RS485 and Radio network

[1] http://www.racom.eu/eng/products/m/ripex/ripex-detail.html

7.2. Profibus settings

We will only be looking at the basic communication parameters of the protocol – other parameters correspond to the standard Profibus DPV0. Profibus protocol is very sensitive to DP Slave response times. Delays are common in radio networks; this should be taken into account when setting up Profibus communication parameters.

Recommended default Profibus settings for data transfer using RipEX radio modems:

Tslot_Init:	16 383 t_bit
Max. Tsdr:	50 t_bit
Min. Tsdr:	11 t_bit
Tset:	1 t_bit
Tqui:	0 t_bit

Explanation of acronyms:

Tslot_init (Slot-time): This indicates how long a DP Master should wait for a response from a DP Slave before it repeats a packet or sends another. The maximum value is 16 383.

Max. Tsdr (Maximum Station Delay of Responders): Sets the maximum DP Slave response time. This value is the same for all DP Slaves and is distributed from the DP Master at the beginning of their communication. This value must be lower than Tslot_init (Slot-time).

Min. Tsdr: Sets the minimum DP Slave response time. 11 to 255 bit values are permitted. This value is the same for all DP Slaves and is distributed from the DP Master at the beginning of their communication. This value must be lower than Max. Tsdr.

Tset: Sets delay. This is used to postpone broadcasting of the next packet. This parameter enables you to create space for other communication on RipEX network.

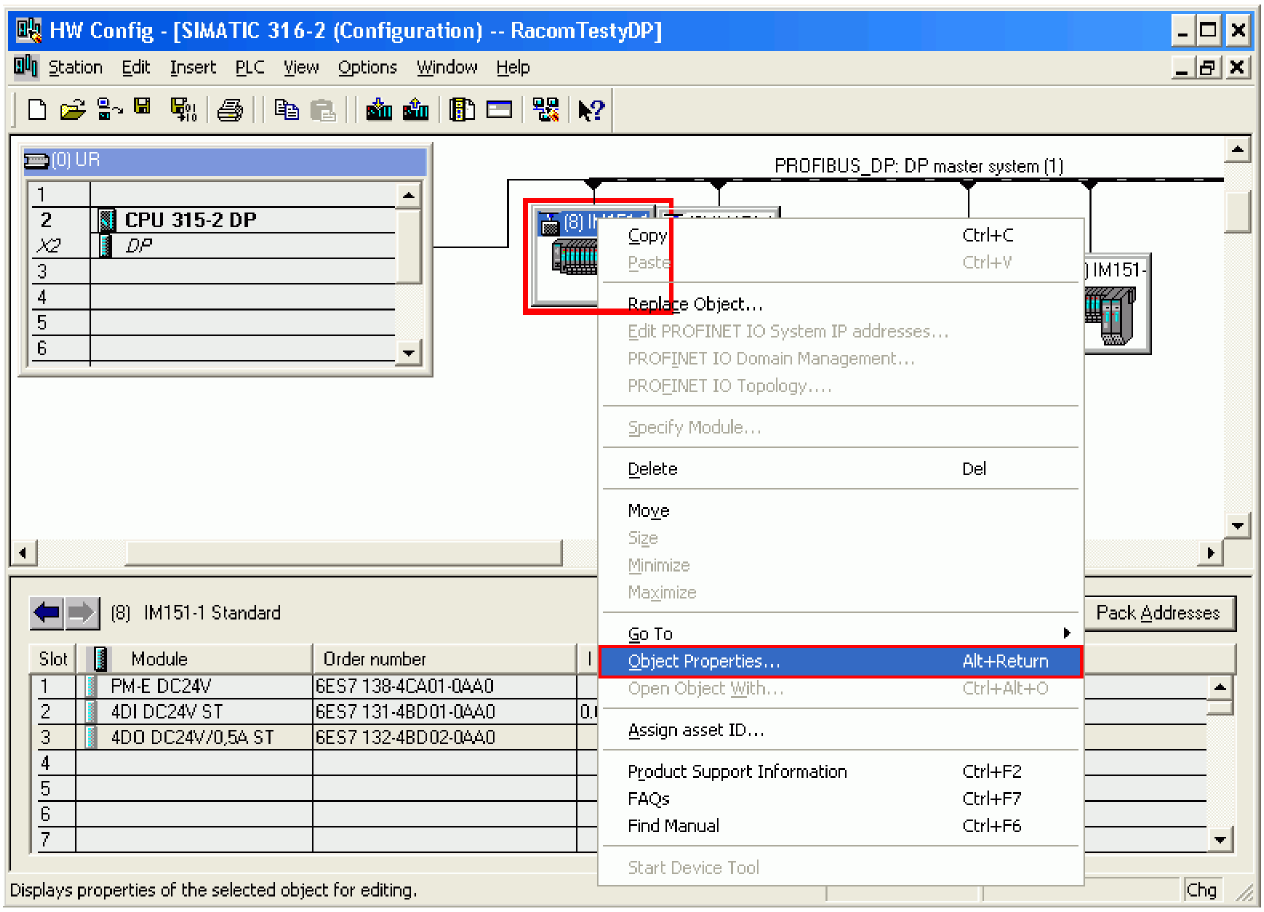
Tqui (Quit time): Sets the switching time between reception and broadcasting. This must be lower than Min. Tsdr.

Note: All times are given in bits. 1 t_bit = 1 / Baud rate [seconds]

A single bit time	Baud rate – data transfer speed
104.2 µs	9600 bps
52.1 µs	19200 bps

Example of Profibus DP settings in STEP 7

Under network layout click the right mouse button to open Object Properties:


HW Config - [SIMATIC 316-2 (Configuration) -- RacomTestyDP]
Station Edit Insert PLC View Options Window Help
(0) UR
PROFIBUS_DP: DP master system (1)
(8) IM151-1
IM151-
Copy Ctrl+C
Paste Ctrl+V
Replace Object...
Edit PROFINET IO System IP addresses...
PROFINET IO Domain Management...
PROFINET IO Topology....
Specify Module...
Delete Del
Move
Size
Minimize
Maximize
Go To
Object Properties... Alt+Return
Open Object With... Ctrl+Alt+O
Assign asset ID...
Product Support Information Ctrl+F2
FAQs Ctrl+F7
Find Manual Ctrl+F6
Start Device Tool
Pack Addresses
Chg

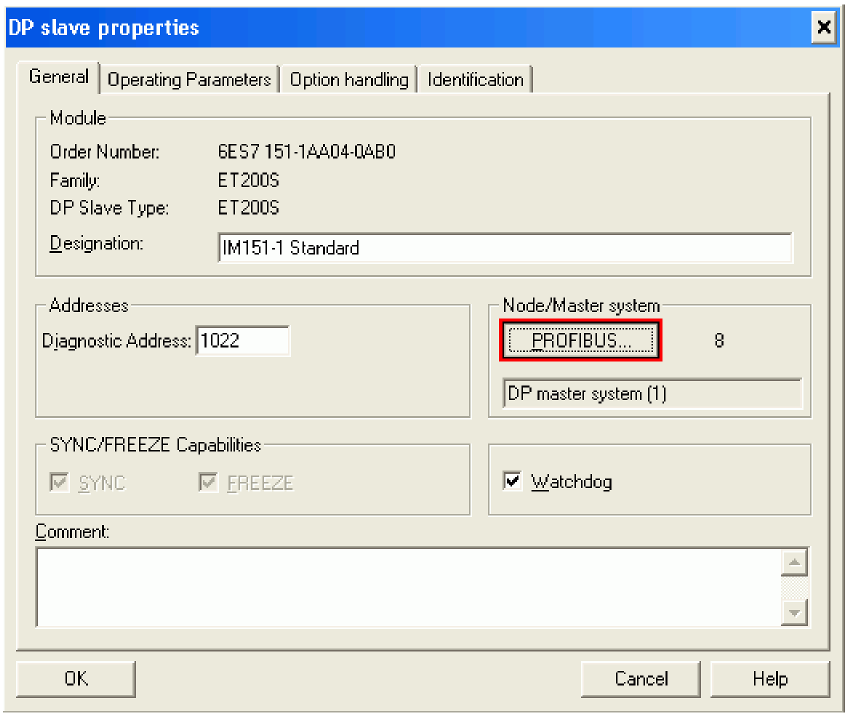
1	
2	CPU 315-2 DP
X2	DP
3	
4	
5	
6	

(8) IM151-1 Standard

Slot	Module	Order number
1	PM-E DC24V	6ES7 138-4CA01-0AA0
2	4DI DC24V ST	6ES7 131-4BD01-0AA0
3	4DO DC24V/0,5A ST	6ES7 132-4BD02-0AA0
4		
5		
6		
7		

Displays properties of the selected object for editing.

DP slave properties window opens. Click on the PROFIBUS button:



Module

Order Number: 6ES7 151-1AA04-0AB0

Family: ET200S

DP Slave Type: ET200S

Designation: IM151-1 Standard

Addresses

Diagnostic Address: 1022

Node/Master system

8

DP master system (1)

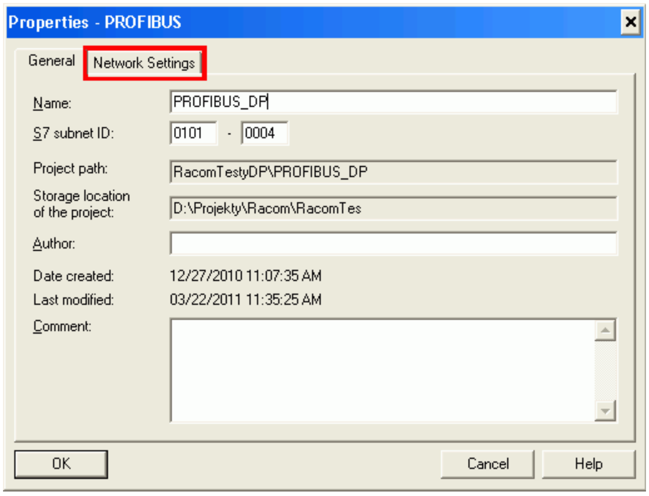
SYNC/FREEZE Capabilities

☑ SYNC ☑ FREEZE

☑ Watchdog

Comment:

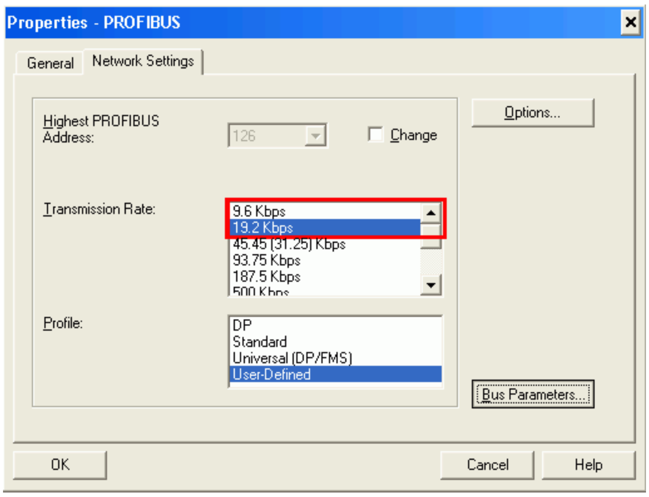
Properties – PROFIBUS window opens. Select the Transmission Rate (19.2 Kbps or 9.6 Kbps) under the Network Settings tab. The recommended value is 19.2 Kbps. Under Profile select User Defined and click Bus Parameters.



Properties - PROFIBUS

General | Network Settings

Name:	PROFIBUS_DP
S7 subnet ID:	0101 - 0004
Project path:	RacomTestyDP\PROFIBUS_DP
Storage location of the project:	D:\Projekty\Racom\RacomTes
Author:	
Date created:	12/27/2010 11:07:35 AM
Last modified:	03/22/2011 11:35:25 AM
Comment:	



Properties - PROFIBUS

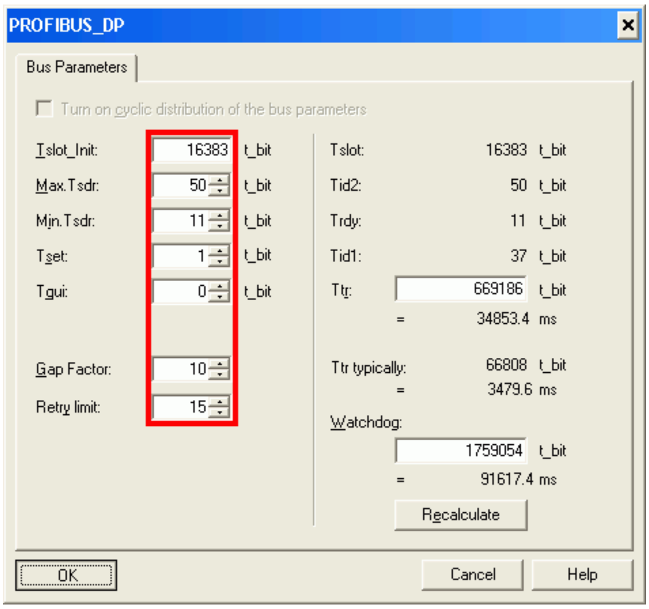
General | Network Settings

Highest PROFIBUS Address: 126 □ Change

Transmission Rate: 9.6 Kbps, **19.2 Kbps**, 45.45 (31.25) Kbps, 93.75 Kbps, 187.5 Kbps, 500 Kbps

Profile: DP, Standard, Universal (DP/FMS), **User-Defined**

PROFIBUS_DP is the most important settings window; fill in settings as shown below, click Recalculate and confirm by clicking OK. Confirm the values in all open windows and click the icon Download to Module. Tslot_Init is a value which fundamentally influences operation of the entire device. 16 383 t_bit is the maximum value which helps test radio transmission. We recommend setting as described in chapter "Advanced Settings – Calculation of minimum slot time".

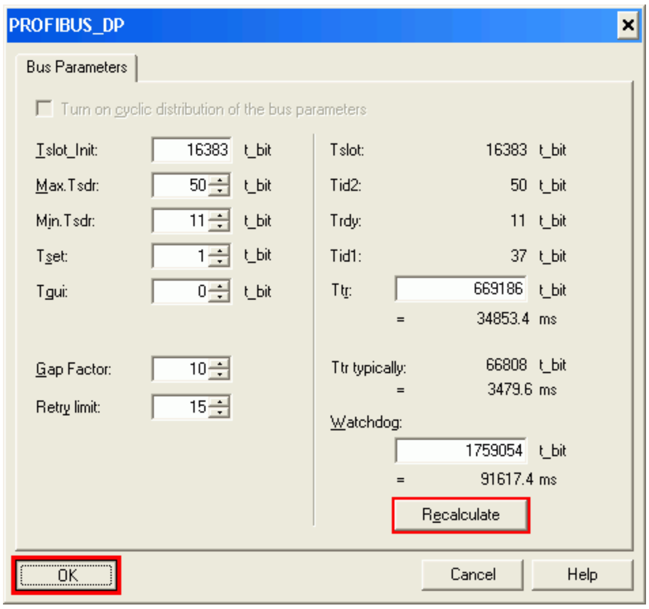


PROFIBUS_DP

Bus Parameters

□ Turn on cyclic distribution of the bus parameters

Tslot_Init:	16383	t_bit	Tslot:	16383	t_bit
Max.Tsdr:	50	t_bit	Tid2:	50	t_bit
Min.Tsdr:	11	t_bit	Trdy:	11	t_bit
Tset:	1	t_bit	Tid1:	37	t_bit
Tqui:	0	t_bit	Ttr:	669186	t_bit
			=	34853.4	ms
Gap Factor:	10		Ttr typically:	66808	t_bit
			=	3479.6	ms
Retry limit:	15		Watchdog:	1759054	t_bit
			=	91617.4	ms



PROFIBUS_DP

Bus Parameters

□ Turn on cyclic distribution of the bus parameters

Tslot_Init:	16383	t_bit	Tslot:	16383	t_bit
Max.Tsdr:	50	t_bit	Tid2:	50	t_bit
Min.Tsdr:	11	t_bit	Trdy:	11	t_bit
Tset:	1	t_bit	Tid1:	37	t_bit
Tqui:	0	t_bit	Ttr:	669186	t_bit
			=	34853.4	ms
Gap Factor:	10		Ttr typically:	66808	t_bit
			=	3479.6	ms
Retry limit:	15		Watchdog:	1759054	t_bit
			=	91617.4	ms

7.3. RipEX settings

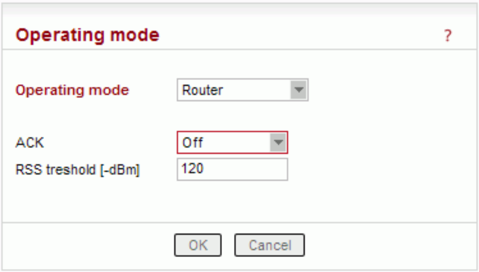
7.3.1. Operating mode

See chapter Advanced configuration[2] of the manual.

If there is no more than a single repeater on your network, we recommend using Bridge mode. Profibus DP is always a master–slave type network in which there is no danger of radio channel collisions.

[2] http://www.racom.eu/eng/products/m/ripex/h-menu.html

Router mode should only be used where network topology does not allow for Bridge mode to be used (see page YY of the manual). If you choose to use Router mode we recommend switching off acknowledgement on the radio channel. This speeds up packet transmission on the radio channel. Repetition of undelivered packets is ensured through the application layer of the DP Master.



Operating mode

Operating mode: Router

ACK: Off

RSS treshold [-dBm]: 120

Fig. 7.2: ACK Off

7.3.2. COM 2

Profibus DP utilises RS485 interface. This interface can only be set to COM2 in RipEX. COM2 functionality is conditioned by using the appropriate software key, see chapter Maintenance[3] of the manual.

COM2 settings must correspond to PLC device settings. We recommend setting port speed to 9600 for complex networks or 19200 bps for networks without re-translation (the timing is derived from the length of a single bit).

Idle state can be reduced to as little as 1.

In Router mode, set Protocol to Profibus.

For explanation of the individual parameters refer to on-line help in the web interface or chapter Settings[4] of the manual.

Note: If Profibus IP's do not correspond to RipEX IP's (e.g. several PLC Slaves are connected to a RipEX over a single bus), addresses must be **translated using a table**.

7.4. Advanced settings

7.4.1. Calculation of minimum slot time

Setting the appropriate (minimum) Tslot_Init value for a given network may significantly shorten the total DP Slave polling cycle. If one of the DP Slaves is out of order or if its response is lost, the DP Master will only wait for a set minimum time before sending another query. The value should be set to maximum to prevent problems.

The calculator on http://www.racom.eu/eng/products/radio-modem-ripex.html#calculation enables you to calculate the RTT (round trip time).

Set the PLC Master to Ethernet interface in the calculator (Profibus protocol timing is based on the last sent byte; time on Master's RS485 does not figure in this calculation).

RTT for Bridge mode can be used directly; for Router mode the resulting average RTT needs to be multiplied by constant 1.25 to receive the maximum achieved RTT.

Calculate the recommended Tslot_Init as follows:

Tslot_Init = RTT * (Port speed in bps) / 1000

[3] http://www.racom.eu/eng/products/m/ripex/h-menu.html

[4] http://www.racom.eu/eng/products/m/ripex/h-menu.html

7.4.2. Router mode - timing

Router mode web based settings may cause time problems in more complex networks. CLI lets you adjust radio channel access parameters and set up repetition taking into account the number of re-translations in your radio network.

If you only use the Profibus protocol with RipEX and no other broadcast interferes with your network, you can configure certain parameters to shorten the access time to channel using CLI. If you want to use packet acknowledgement on the radio channel, you can shorten the repetition timeout if ACK is turned off.

Set up using CLI:

cli_cnf_set_device_mode:

-ack n	Turns on ACK
-retries 2	Number of retries 2
-rto-prog f	Turns off progressive retries
-rto-fix 10	Shortens the retry timeout to the minimum value of 10 Bytes
-rto-var 10	Shortens the variable retry timeout to the minimum value of 10 Bytes
-slots-rx 0	Will receive immediately after request – random channel access is not used
-slots-tx 0	Will transmit immediately after request – random channel access is not used

Same settings should be used for all devices.

To find out more about CLI, see RipEX manual chapter CLI Configuration[5].

Set the following in Profibus parameters:

Tslot_Init 16383

Note: This setting is only appropriate for certain types of networks; changes should only be made by experienced users!

Connecting RS 485

Connector layout of RipEX COM 2 for RS 485 and the corresponding PIN's on Siemens Simatic S7.

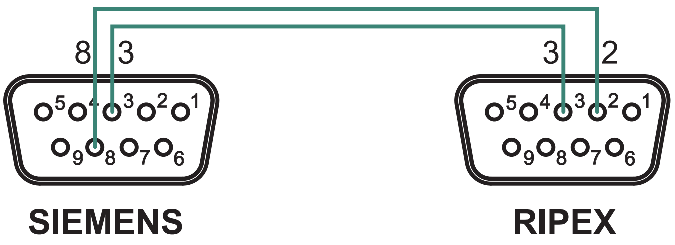


Fig. 7.3: RS485 connection

[5] http://www.racom.eu/eng/products/m/ripex/cli-conf.html

8. Modbus TCP/RTU

Use of Modbus in RipEX.

RipEX supports Modbus RTU, Modbus TCP as well as their combinations:

Tab. 8.1:

	Centre protocol	**Remotes' protocol**	**Radio network behaviour**	**Available with Operating mode**
1	RTU	RTU	Modbus RTU over Radio channel	Bridge, Router
1.1	Multiple Masters RTU	RTU	Modbus RTU over Radio channel	Router
2	TCP	TCP	TCP/IP protocol over Radio channel	Bridge, Router
3	TCP	TCP	TCP/IP protocol locally between Modbus device and RipEX. TCP/IP overhead is not transferred over Radio channel	Router
4	TCP	RTU	Conversion of Modbus TCP to Modbus RTU on the remote units	Router
5	TCP	Combination of TCP and RTU	Using 3 and 4	Router
6	Multiple TCP and multiple RTU masters	Combination of TCP and RTU	TCP master communicates with TCP or RTU slaves, RTU Master only communicates with RTU slaves, utilising 1.1 and 5	Router

8.1. Modbus RTU

A standard simple network design with a single Master and several Slaves running Modbus RTU.

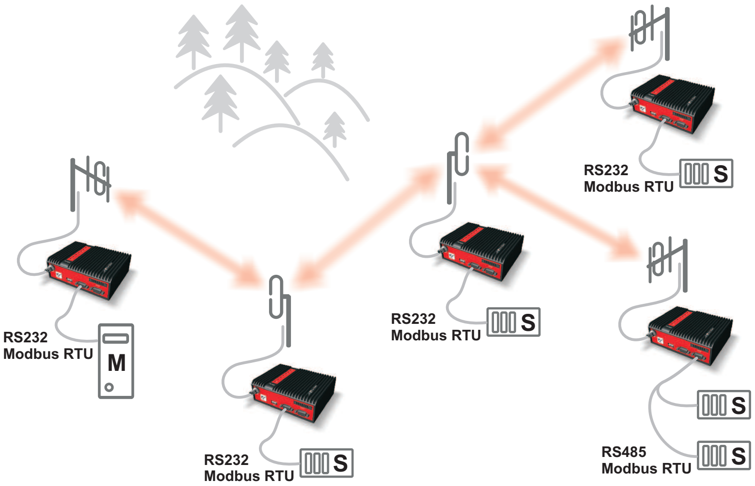


Fig. 8.1: Modbus RTU

In Bridge mode, set the type of communication interface (RS232 or RS485) for the COM port as well as the parameters of the serial interface, both for the Master and Slave.

In Router mode, set the COM port of your Master RipEX to Modbus (Mode of Connected device). To translate Modbus addresses to RipEX format and vice versa either use a mask (if RipEX addresses mirror the Modbus ones) or table. A table must be used if there are several Modbus slaves behind a single RipEX (RS485 or both COM1 and COM2). For more information refer to on-line help or chapter XX of the manual.

In addition, set Modbus to Slave on all remote units. If you intend to broadcast in Modbus, set the required parameters. For more information refer to on-line help or chapter XX of the manual.

8.1.1. Modbus RTU with multiple Masters

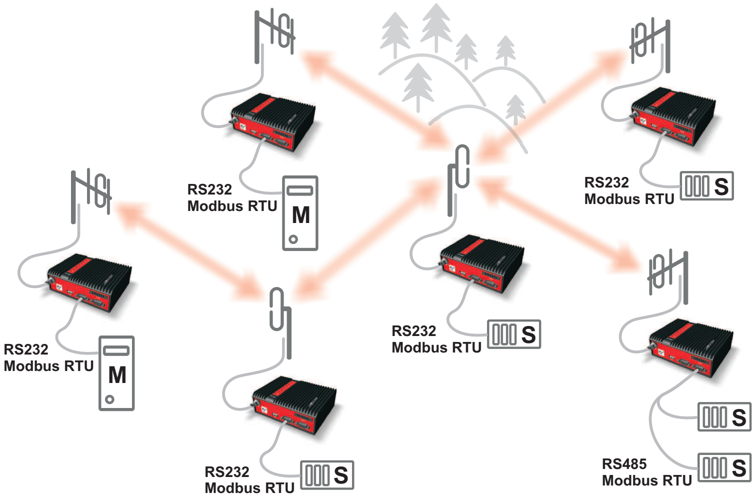


Fig. 8.2: Modbus RTU with multiple Masters

RipEX allows for several Masters to operate at the same time and to communicate with the same Slaves. Router mode is presumed in this design. RipEX settings remain the same as above. Each Slave responds directly to the Master unit which queries it – i.e. if Master A issues a query to a Slave, the response is sent exclusively to Master A. If a single Slave is queried by two Masters at once, queries are resolved one by one. Query from the second Master is queued inside RipEX until it receives a response from Slave RTU on its serial interface or until 500 ms timeout has passed.

8.2. Modbus TCP

A standard simple network with a single Master and several Slaves running Modbus TCP. A TCP/IP connection is established and maintained between Master PLC and Slave RTU across the entire radio network.

In Bridge mode, no special setup is required. RipEX operates as an intelligent Bridge. For more information refer to on-line help or chapter XX of the manual.

In Router mode, routing must be set up in the radio network. Communication between the IP address of the Modbus Master and IP addresses of all Modbus Slaves is necessary. Remember to set Modbus TCP/RTU and Terminal Servers (under Settings/Ethernet) to Off.

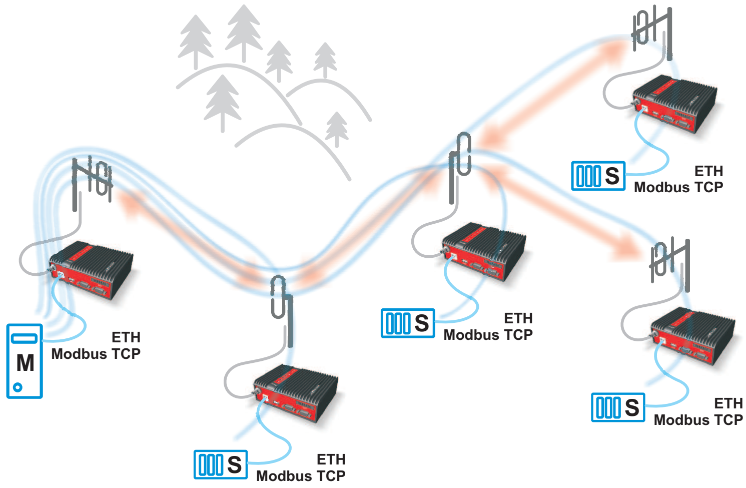


Fig. 8.3: Modbus TCP

8.3. Modbus TCP, local TCP/IP connection

Note - Only works in Router mode.

TCP connection is established only locally between Modbus devices and the connected RipEX units. TCP protocol overhead is not transmitted over the Radio channel. Secured TCP/IP transfer is not necessary because in Router mode every packet in the Radio channel is acknowledged on every radio hop. A packet is therefore repeated directly in the part of the network where it is lost, not across the entire radio network as in TCP/IP. This improves latency and increases network throughput.

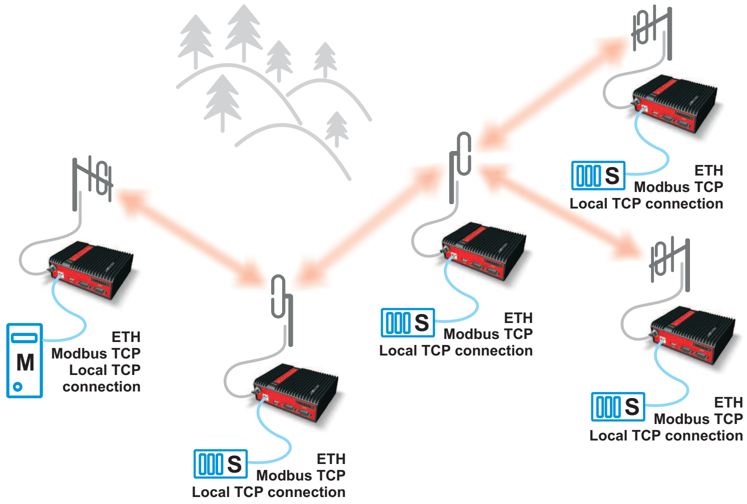


Fig. 8.4: Modbus TCP local

Set your Modbus TCP Master to use a single IP to communicate with Modbus TCP Slaves (RipEX ethernet IP) and set TCP port to 504. Communication begins on port 504 from where it is redirected to other RipEX ports, corresponding to the individual RTU's, based on negotiation with the Modbus TCP Master.

To set up RipEX connected to Modbus TCP Master:

- Set Modbus TCP/RTU to On. Type the port number on which the connected Modbus TCP Master initiates communication, by default 504, into “My TCP Port” field.
- Select how you want to translate Modbus addresses to RipEX IP addresses (using mask or table). Set the UDP interface to Terminal server (TS1-TS5). Set the same TS for remote RipEX’s too.

Select how you want to translate Modbus addresses to RipEX IP addresses (using mask or table). Set the UDP interface to Terminal server (TS1-TS5). Set the same TS for remote RipEX’s too.

Note

The maximum number of concurrent TCP/IP connections between a Modbus TCP device and RipEX is set to 10 due to limited computing capacity. (Note: The number of concurrently open TCP/IP connections can be increased using CLI if necessary.) Modbus TCP Master must be set to not open more than 10 TCP/IP connections at any given time.

To set up RipEX connected to Modbus TCP Slave:

- Modbus TCP/RTU - Off
- Terminal Servers - On
- Set the Terminal Server (see RipEX Master settings) to TCP and set My Port to 504. Use the address of the connected Modbus Slave as the destination IP and fill in the destination port number which the connected Modbus Slave device scans for incoming communication.
- Set Protocol to UNI and Mode of Connected device to Slave.

8.4. Master - Modbus TCP, slaves - Modbus RTU

Note - Only works in Router mode.

Master establishes a local TCP connection to RipEX using Modbus TCP protocol, as described in chap. 3. A packet is securely sent over the Radio network to RipEX to which the destination Slave is connected by COM port. The RipEX translates the packet to Modbus RTU format and sends it to the connected Slave using Modbus RTU protocol.

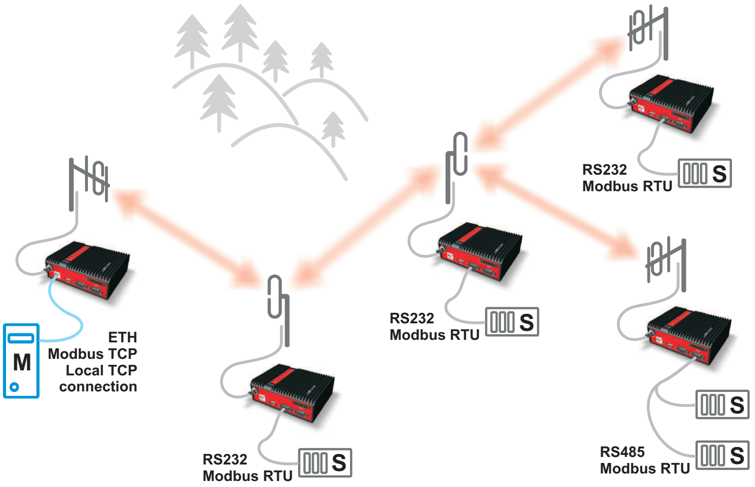


Fig. 8.5: Modbus TCP - RTU

To set up RipEX connected to Modbus TCP Master:

- Select the type of translation from Modbus to RipEX IP address (mask or table), as described in chapter 3.
- Set the UDP interface to COM1 or COM2 depending on the port that the remote RipEX uses to connect to the Slave device.

To set up RipEX connected to Modbus RTU Slave:

- As described in chapter 1 set the appropriate COM to Modbus and the Mode of Connected to Slave.

8.5. Master Modbus TCP, slaves Modbus RTU or Modbus TCP

RipEX radio modems enable full featured cooperation between the Master using Modbus TCP and slave devices using Modbus RTU or Modbus TCP within a single network.

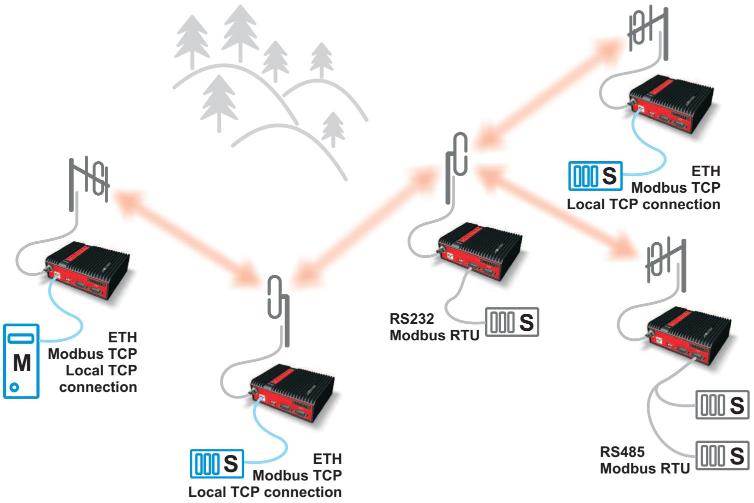


Fig. 8.6: Modbus TCP, Slave RTU or TCP

To set up RipEX connected to Modbus TCP Master:

- Set the translation from Modbus to RipEX IP addresses to table-based, as described in chapter 3.
- For devices connected over Modbus RTU, set the UDP interface to COM1 or COM2 (as in chapter 4).
- For devices connected over Modbus TCP, set the UDP interface to TS1-TS5, as described in chapter 3.
- You can define address ranges in the table for greater ease of use.

To set up RipEX connected to Modbus RTU Slave:

- See chapters 4 and 1 respectively.

To set up RipEX connected to Modbus TCP Slave:

- See chapter 3.

8.6. Multiple Modbus TCP or Modbus RTU Masters and Slaves

Any combination of network designs described in chapters 1–5 is possible. The only limitation is that a Master with Modbus RTU cannot communicate with a Slave using Modbus TCP.

A Slave with Modbus RTU protocol may simultaneously communicate with masters using Modbus TCP and Modbus RTU. The network will deliver responses only to the Master which issued the queries using the appropriate protocol.

The individual settings are described in chapters 1–5.

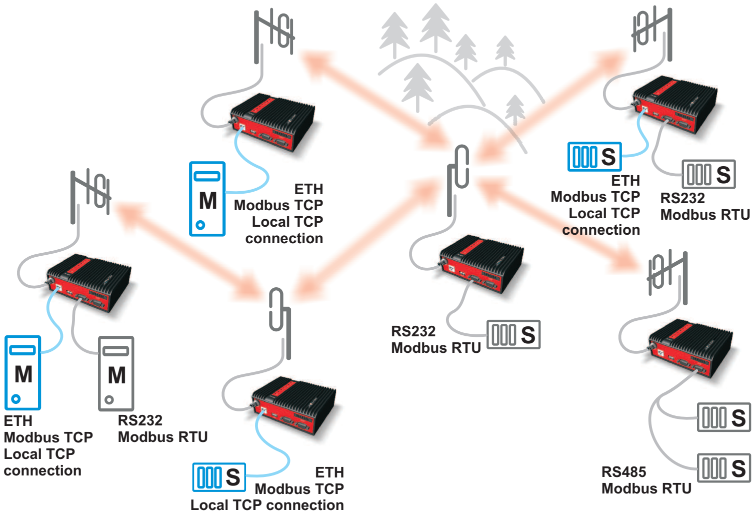


Fig. 8.7: Modbus TCP, Slave RTU or TCP

9. UNI protocol

UNI is the "Universal" protocol utility designed by RACOM. It is not a new SCADA protocol, it can actually process different protocols of different vendors. It supports both the standard MASTER -SLAVE and the MULTI MASTER types of communication. At least one Master is required in the network.

The SCADA protocol to be handled by the UNI has to meet solely the following condition: There has to be an 8 or 16 bit* protocol address in every message generated by a Master station and the address position in all messages has to be the same. The position of address in the reply from an RTU is not relevant, because the reply is always send back to the address where the request originated.

Note

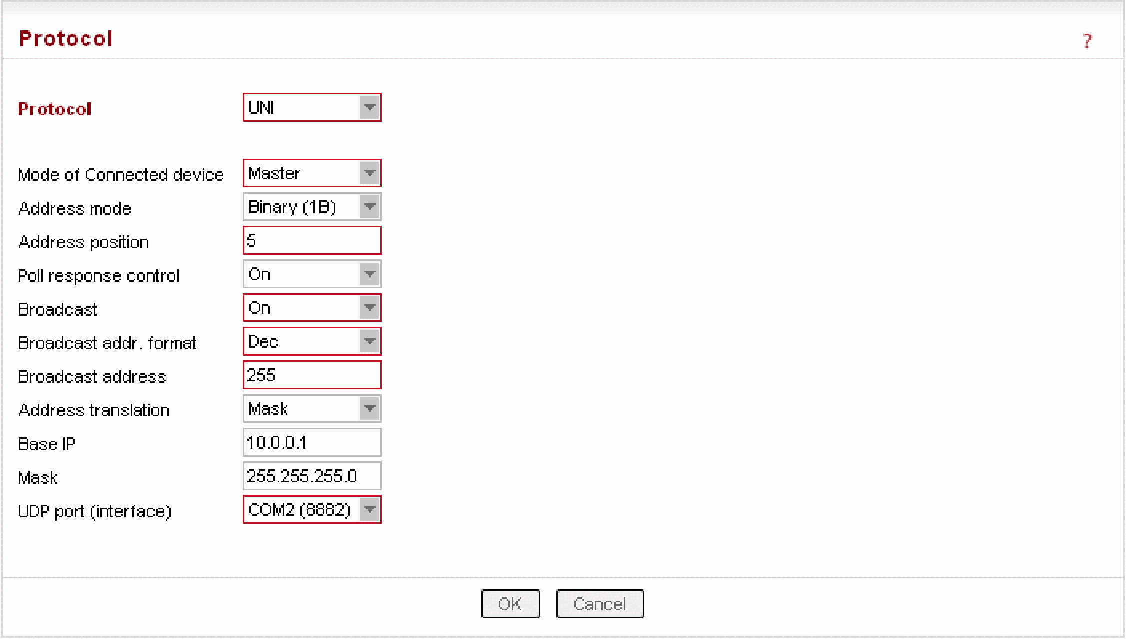
Some SCADA protocols use two byte ASCII address, which is an ASCII representation of an 8 bit address in the hexadecimal format (e.g. "8C" means 8-bit value 0x8C in hex / 140 in decimal notation).

Address bytes for some protocols:

PR2000	3rd Byte
RDS	2nd Byte
Mars-A	8th Byte (without local ACK)
Hirsch	2nd Byte

9.1. MASTER – SLAVE communication

Master reads the address byte defined by configuration and generates the destination IP address using the mask or the translation table. The message is then delivered to that IP address and the respective UDP port (e.g. the port No 8882 which is assigned to the COM2 interface).



Protocol

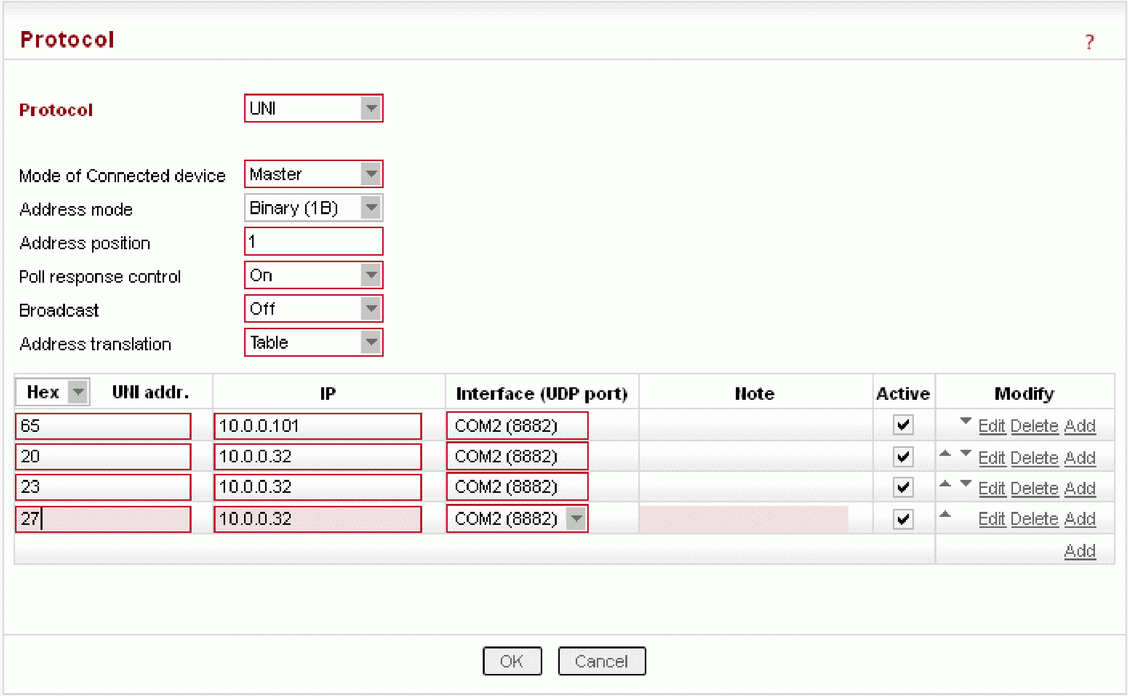
Protocol: UNI

Mode of Connected device: Master
Address mode: Binary (1B)
Address position: 5
Poll response control: On
Broadcast: On
Broadcast addr. format: Dec
Broadcast address: 255
Address translation: Mask
Base IP: 10.0.0.1
Mask: 255.255.255.0
UDP port (interface): COM2 (8882)

An example of Master configuration is in the picture above. The address translation then proceeds as follows:

The 5th byte from the incoming message from SCADA centre is used to replace the last byte of the Base IP and the resulting IP address is used as the destination of the UDP datagram which contains the original SCADA message.
Let assume that the content of 5th byte is 0x65 - then the IP destination address will be 10.0.0.101 and the UDP port 8882.

The translation by a table is more versatile, however it requires an extra line of configuration for every remote in the network. The table has to be used when addresses of RipEX radiomodems and SCADA RTUs do not match or different ports (interfaces) at different remotes have to be configured.



Protocol: UNI

Mode of Connected device: Master

Address mode: Binary (1B)

Address position: 1

Poll response control: On

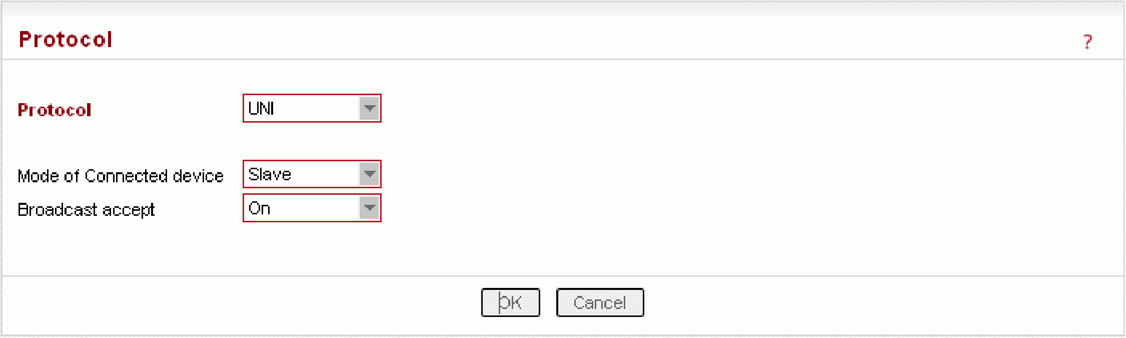
Broadcast: Off

Address translation: Table

Hex UNI addr.	IP	Interface (UDP port)	Note	Active	Modify
65	10.0.0.101	COM2 (8882)		☑	▼ Edit Delete Add
20	10.0.0.32	COM2 (8882)		☑	▲ ▼ Edit Delete Add
23	10.0.0.32	COM2 (8882)		☑	▲ ▼ Edit Delete Add
27	10.0.0.32	COM2 (8882)		☑	▲ Edit Delete Add
					Add

The example of table in the picture above demonstrates a situation when there are three SCADA devices connected to the COM2 of a single RipEX unit over a RS485 bus.

The configuration of a Slave radiomodem is very simple, as demonstrated in the picture below. When a UNI Slave receives the UDP datagram from RF channel, it takes the original SCADA message and transmits it over the respective interface (the COM2 in our example).



Protocol: UNI

Mode of Connected device: Slave

Broadcast accept: On

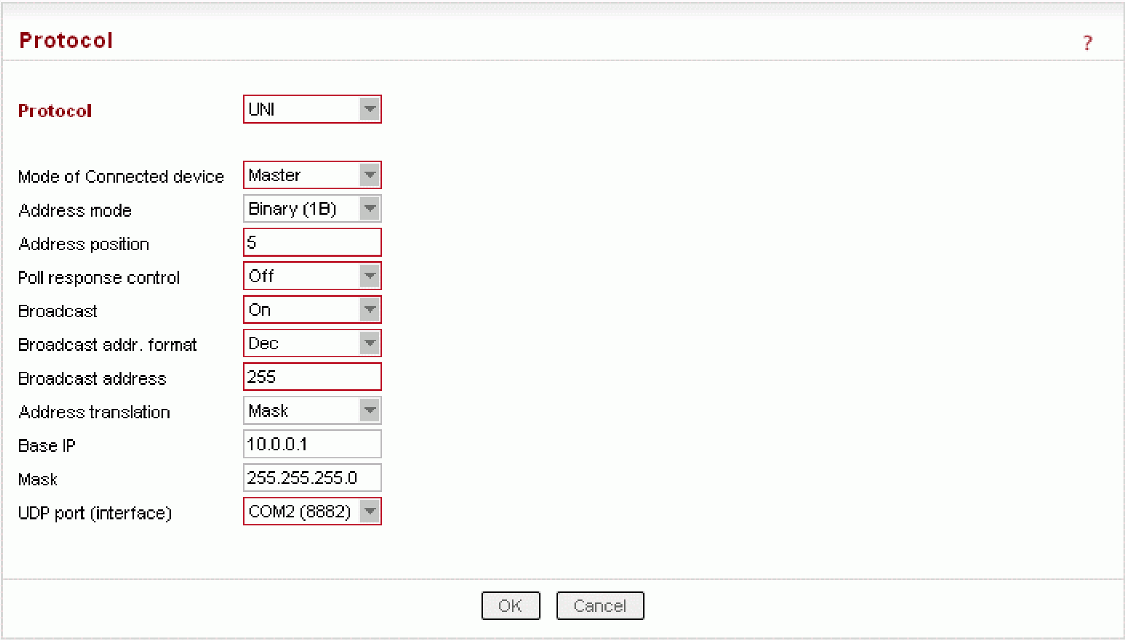
If the SCADA device connected responds to the message within a timeout of 500 ms, the source IP address of the received UDP datagram is used as the destination for the response.(Note only one packet is accepted as a response). When the timeout expires, all messages received by the serial interface are discarded.

9.2. MASTER – SLAVE with several Masters

The behaviour of Master and Slave is exactly the same as in the previous scenario, i.e. a Slave always responds to the address from which the request was sent. If by chance two simultaneous requests from different Masters are received by a slave radiomodem, the RipEX radio modem waits for the first reply from the connected SCADA device before transmitting the request which arrived second. The 500 ms timeout applies again, i.e. when there is no reply for the first request, the second one is transmitted after the timeout expires.

9.3. MASTER – MASTER

The Master - Master communication is possible. The translation of addresses is proceeded with every packet incoming to the RipEX radio modem from connected SCADA equipment, thus it is suitable for SCADA protocols containing the destination address in all packets.



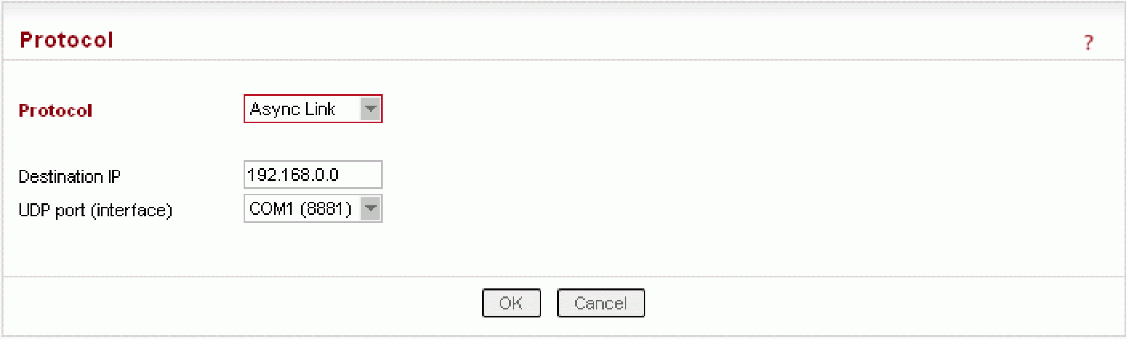
Protocol

Protocol	UNI
Mode of Connected device	Master
Address mode	Binary (1B)
Address position	5
Poll response control	Off
Broadcast	On
Broadcast addr. format	Dec
Broadcast address	255
Address translation	Mask
Base IP	10.0.0.1
Mask	255.255.255.0
UDP port (interface)	COM2 (8882)

The Poll Response Control has to be set to OFF for the MASTER-MASTER type of communication.

9.4. MASTER UNI – ASYNC LINK SLAVES

The combination of the UNI and the ASYNC LINK protocols is useful for networks where one application master communicates with many slaves and the slaves are allowed to spontaneously send messages to the master. The UNI-Master RipEX§s address is configured as the ASYNC LINK protocol destination address at all the slaves. This arrangement makes the syntax of application protocol messages generated by slave completely arbitrary. All slave messages are transparently delivered to the application master.



Protocol ?

Protocol: Async Link

Destination IP: 192.168.0.0

UDP port (interface): COM1 (8881)

Note that, similarly to the MASTER-MASTER mode, the Poll Response Control at the Master RipEX has to be set to Off.

10. Channel access

Method of accessing the radio channel may significantly affect the overall reliability of packet transmission. Even in a simple polling-type application, which never generates more than a single packet at a time, collisions may occur when repeaters are used. The goal of channel access is either to eliminate collisions completely, or to reduce their probability while ensuring that systematic repeated collisions never happen. RipEX provides different channel access methods in different modes and optimum configuration can be found for every communication scheme and network layout.

10.1. Collisions

What is so special about collisions that they deserve that much attention? Well, they are a special case of interference (“friendly fire”, a military reporter would say), which may very seriously harm network performance.

A collision happens, when two (or more) transmissions in the network overlap in time. Radio modem A transmits a packet for B, C transmits for D. In well designed network the respective signal levels (i.e. A received at B, C received at D) do ensure error-less reception. For the period of time when these two transmissions overlap, signal from C at receiver input B and signal A at D act as interference signals, reducing the SNR (Signal to Noise Ratio). If B and D are in the same area, the difference in signal strength is small and so is the resulting SNR at both receivers. Consequently the BER (Bit Error Rate) at both receivers jumps to unacceptable level and none of the packets is successfully received. That is the basic principle of a collision.

There are two very harming features of collisions:

The first is a systematic repeated collision. No application generates a totally random traffic pattern. So it may happen (and it does happen), that a certain sequence of packets in a certain network layout generates a collision and it generates this collision repeatedly, in fact always. The result is that certain specific packets are never delivered, regardless of number of retries set at the application level. Imagine a SCADA system never capable of performing one specific task, while all communication tests report that links are in perfect shape. It would be very tempting to blame the SCADA, while the true problem is a systematic collision, i.e. wrong network design. Ways to avoid such collisions are described further in this document.

The second dangerous feature of collisions is just a direct consequence of probability laws. The most effective communication scheme for many applications is the report-by-exception mode, which can vastly reduce the amount of mainly useless traffic generated by polling-type systems. Report-by-exception means though, that collisions can never be ruled out completely, hence a collision-solving system must be an integral part of the protocol in the radio channel (RipEX in router mode provides such protocol of course). Solving a collision means retransmission, typically a delayed retransmission. Consequently the probability of another packet being generated by the application in the meantime increases by the delay, and it increases at both parties involved in the collision. That results in an increased probability of next collision to happen...and so on. This principle makes report-by-exception networks very sensitive to bursty loads. Whenever the load increases over certain limit (we may say “normal” network capacity), number of collisions grows exponentially, reducing the instant network capacity well below normal situation. Series of lost packets and very long delivery times are the result from the application point of view. While the network for report-by-exception application has to be designed to provide maximum capacity possible, it is recommended to take measures to avoid burst load generation at the application level. Limiting the possible load generated by a single device can help to avoid the whole network collapse just because one remote unit goes suddenly “crazy” (e.g. generates hundreds of “exceptions” per second).

10.2. Bridge mode

In Bridge mode, a packet is transmitted to the radio channel immediately, without any checking whether the radio channel is occupied or not. If other radio was transmitting simultaneously, a collision would occur and both packets would be lost. Consequently Bridge mode can be used only for applications which never generate more than a single message at a time, e.g. master-slave polling applications. Still appropriate measures have to be taken to avoid collisions in special situations.

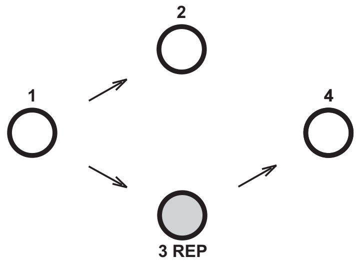
10.2.1. Bridge mode with Repeaters

Repeaters can be used in the Bridge mode in order to extend the radio coverage. Considering the repeated packets, it is necessary to schedule the access to the radio channel to avoid systematic collisions. In a polling-type network, there is a request packet from centre to remote, to which the remote responds immediately. When a remote receives the request directly from the centre, its immediate response would collide with the repeated request, so it would be never received by the centre – a perfect example of a systematic collision.

Packet header contains information about the number of Repeaters on the route, i.e. how many times the packet can be possibly repeated. This number is decremented when passing through each Repeater. The remote radio modem which receives the packet must delay its own transmission for a period. This delay is calculated from the number of the remaining repetitions, the packet length and the modulation rate in the radio channel. Repeaters always transmit immediately, without any delay.

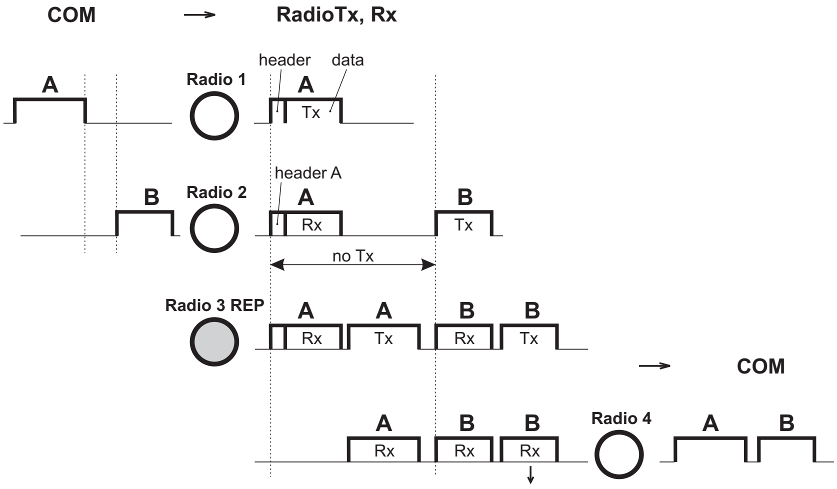
Example:

There are 4 radios in the network operated in the Bridge mode. Everyone can receive each other except Radio 4, which is not able to receive Radio 1 and vice versa. Therefore, in the Radio 3 the Repeater function is turned on, and it mediates the connection between 1 and 4.



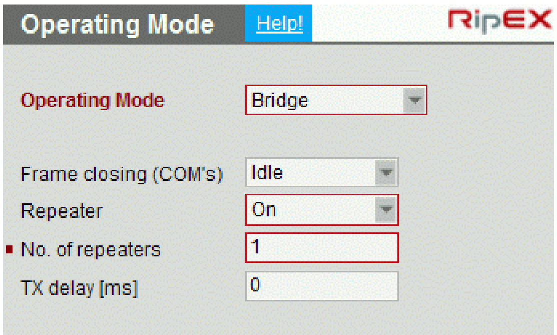
First, packet A is broadcast from Radio 1.

Radio 2 receives Packet A and sends it to its COM. In the instant when it starts the reception of Packet A, Radio 2 calculates (from information in the received packet header and from number of repeaters in its own setting) the time delay which is needed for the delivery of Packet A through the repeater (repeaters). When the response from the connected device arrives via COM (Packet B), the Radio 2 postpones its transmission for the delay.



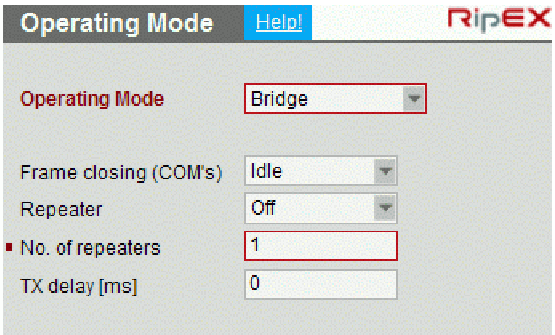
In the meantime, Radio 3 (Repeater) receives Packet A and repeats it to the radio channel immediately. Radio 4 receives the Packet A and then Packet B and sends them both to the COM. Packet B is also received by Radio 3 and immediately repeated. Whenever a radio receives a copy± of the same packet during the calculated delay, it discards it as a repeated packet. Note that the picture does not show all the packets at all the radios.

Repeater is configured in the Settings / Device / Operating Mode menu, for Radio 3 (left) and Radio 1, 2, 4 (right):


Operating Mode
Help!
RipEX

Operating Mode: Bridge

Frame closing (COM's): Idle
Repeater: On
■ No. of repeaters: 1
TX delay [ms]: 0


Operating Mode
Help!
RipEX

Operating Mode: Bridge

Frame closing (COM's): Idle
Repeater: Off
■ No. of repeaters: 1
TX delay [ms]: 0

The delay period based on number of repeaters solves the collision between a repeated packet and a possible response. When more than one repeater is used in a Bridge-mode network, collisions between repeated packets from different repeaters may occur. These cannot be solved by simple delays, rather a sophisticated anti-collision protocol is required. The RipEX Router mode is recommended to be used in more complex networks with multiple repeaters. Nevertheless if certain conditions on signal coverage (sometimes non-coverage) among repeaters, centre and remotes are met, the Bridge mode for a polling-type application can be used. See the chapter Bridge mode[1] in RipEX Manual.

[1] http://www.racom.eu/eng/products/m/ripex/ripex-detail.html

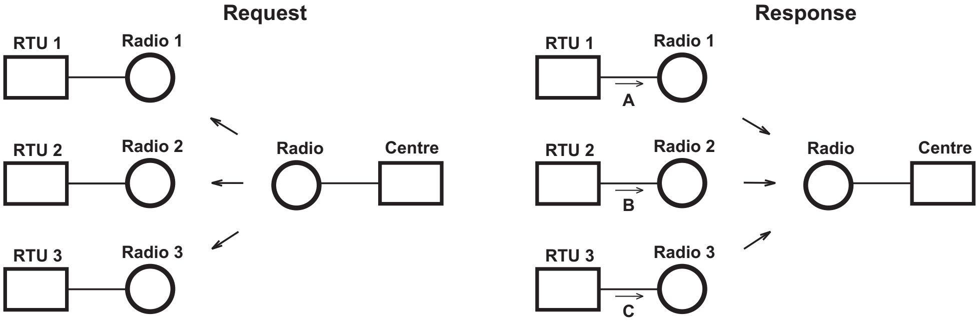
10.2.2. Time division of responses in Bridge mode

There is also the Tx delay setting in the menu. It shall be used in Bridge mode if multiple RTUs connected to slave stations reply to a broadcast query from the centre. It is necessary to spread out their replies to the radio channel in terms of time, otherwise a massive collision occurs. It can be achieved by setting the TX delay parameter to an adequate sequence of TX delays (e.g. 0, 100, 200 ms as in the example below) in individual remote RipEXes. The slave RipEXes will enter the radio channel successively and no collisions will occur.

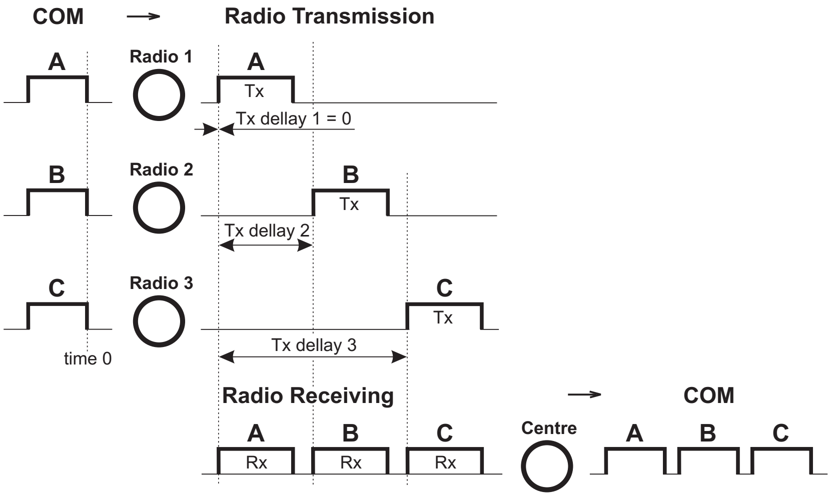
Note: The TX delay applies to every packet that is sent out to the radio channel.

Example:

The Centre broadcasts request and the RTUs 1, 2 and 3 generate the response and send it out to their respective RipEX.



Radios 1, 2 and 3 have the TX delay parameter set to 0, 100 and 200 ms, respectively. Therefore, Radio 1 starts transmitting just after reception of the frame from COM port. Upon 100 ms later, when Radio 1 has completed transmission, Radio2 starts transmitting. Finally, 200 ms after the reception of the packet from RTU, Radio 3 starts its transmission. All three responses are thus sequentially sent to the Centre and no collision happens.



The TX delay parameter coresponds to multiples of maximum packet length expected and shall be set in miliseconds. The packet transmission time through radio channel can be calculated as follows:

```
t = (n + 12). 8/(b . fec)
```

where:

`t [ms]` - time needed for the packet transmission

`n [ - ]` - number of bytes transmitted (consider the longest possible reply from RTU)

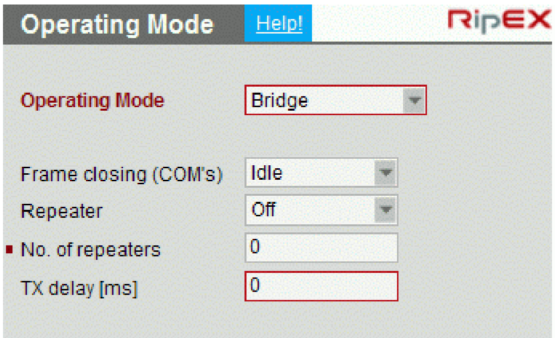
`b [kbps]` - Modulation rate

`fec [ - ]` - Forward Error Correction

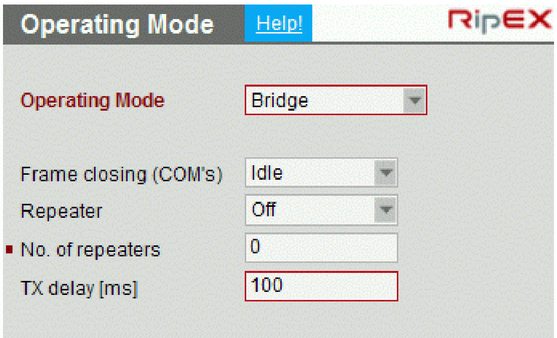
```
fec = 1.00 if FEC = Off
fec = 0.75 if FEC = On
```

This calculation gives approximate results (± 3ms). When more accurate calculation is necessary, please check the calculation tool on Racom web pages http://www.racom.eu/eng/products/radio-modem-ripex.html#calculation

TX delay is configured in the Settings / Device / Operating Mode menu, for Radio 1 (left) and Radio 2 (right):



Operating Mode: Bridge
Frame closing (COM's): Idle
Repeater: Off
No. of repeaters: 0
TX delay [ms]: 0



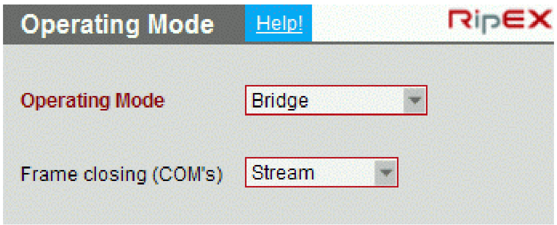
Operating Mode: Bridge
Frame closing (COM's): Idle
Repeater: Off
No. of repeaters: 0
TX delay [ms]: 100

10.3. Bridge mode and COM stream

The COM port in Bridge mode can be switched into the Stream mode. In any other mode, a packet/frame coming to RipEX over any interface has to be received completely before any further processing. In Stream mode the incoming bytes are transmitted to radio channel with minimum possible delay, byte by byte. Consequently nor checks neither processing of the data can be done. All the bytes are simply broadcast to the radio channel and every radio modem which can receive them forwards them immediately to its COM port(s).

Obviously there can not be any repeaters in the Stream mode and no measures against possible collisions can be taken. The responsibility for collision-free communication remains solely with the application. Consequently only simple master-slave polling-type applications, which never respond to broadcasts, can use the Stream mode. This mode should be used solely in applications which would not work when the normal store-and-forward regime is used because of the inevitable delays involved.

The Stream mode is configured in the Settings / Device / Operating Mode menu:



Operating Mode: Bridge
Frame closing (COM's): Stream

10.4. Router Mode

10.4.1. Channel access in Router mode

The protocol in the radio channel in the Router mode of RipEX uses several methods to prevent and solve collisions. The first is Listen Before Transmit. Not a simple CSMA (Carrier Sensing Multiple Access), but a sophisticated LBT with configurable threshold. Only a valid data packet with RSS above threshold or a data packet destined for the RipEX itself is evaluated as a busy channel. RipEX's own transmission is regarded a busy channel as well.

When RipEX evaluates the channel as free, it calculates the Access period – time for which it has to continue monitoring the channel before starting a transmission. Only when the channel stays free for the Access period or more, RipEX starts transmitting whenever a packet destined to radio channel arrives. If channel gets busy, the arriving packets have to wait in a queue and whole process starts from the beginning.

The Access period calculation follows quite complex algorithm, which takes into account RipEX settings, properties of the last packet sent or received and there is – very important – random element. The result is an optimum performance of RipEX's in a report-by-exception network.

10.4.2. Solving collisions in Router mode

When report-by-exception application or multiple-master polling-type one loads the network, collisions can not be avoided completely despite the sophisticated channel access method used. Then a collision-solving algorithm becomes equally important.

The standard protocol feature of sending an Acknowledgement (ACK) to every data packet and retransmitting it when no ACK comes takes care of all possible reasons for packet non-delivery, collisions included. However retransmitting a packet increases the network load and so increases the collision probability. Moreover, it is possible to create a systematic collision by e.g. a regular retransmissions after the initial random collision. Thus the calculation of the retransmission time-out requires a sophisticated approach again. RipEX uses its settings, packet parameters, sequence number of the retransmission and the necessary random element to calculate the time-out.

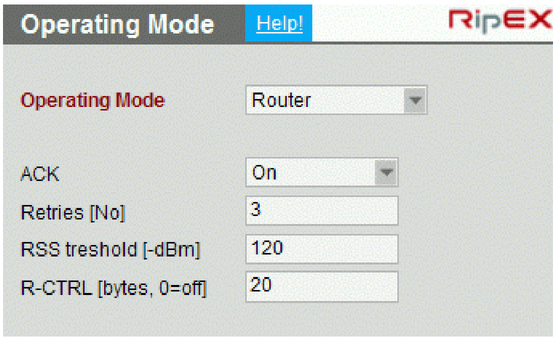
Retransmission feature is enabled by selecting “On” in the ACK listbox. By deciding on number of Retries you define the very important compromise between the longest possible delivery time and the probability of a packet being lost. Note that this setting does not normally affect the typical (most probable) delivery time in the network, since a typical packet is delivered without retransmissions.

Most applications require their data to be delivered completely and error-free, hence there are message retransmissions at the application level. Note that the RF protocol (i.e. RipEX's) retransmissions are always more effective than the application ones, since the radio modem can use more information from the channel when calculating the retransmission time-out. Moreover, when repeaters are involved, retransmitting over a single hop is always faster (and has a greater chance to succeed) than retransmitting over the whole path. Consequently a reasonable approach is to set application time-out to maximum value possible and use an adequate number of Retries in RipEX's in the network. Though the application engineers may find it difficult to understand, such setting will make the application run faster.

There are few exceptions and hitches though. There are applications which rather send a fresh data instead of simply retransmitting the original message. In such case, depending on the frequency of fresh data from the application, the Retries should be set to 1 or ACK switched off completely. Sometimes the application is hard-wired and the retransmission time-out cannot be changed – then it is better to minimize or switch off RipEX's retransmissions again. The trickiest case is when the application centre generates messages to non-existent or switched-off remotes (for any reason). When a remote site is without power (including the RipEX) and the centre continues sending requests to that remote, the last repeater will keep retransmitting these requests for full number of Retries set. More importantly, a long retransmission time-out at the application level is not desirable any more, since it keeps the centre from continuing the polling cycle. Nevertheless in any case it is beneficial to keep the number of application retransmissions at the lowest setting available, i.e. zero if possible, and leave the RipEX network to use the time available for the possible retransmitting.

To calculate the typical and maximum possible delivery time for different settings, please use the calculator on Racom web pages, http://www.racom.eu/eng/products/ripex.html#calculation

The parameters discussed above are configured in the Router operating mode menu. Kindly see the Help pages for further information.



Operating Mode	Router
ACK	On
Retries [No]	3
RSS treshold [-dBm]	120
R-CTRL [bytes, 0=off]	20

Appendix A. Revision History

Revision 1.1	2011-09-02
First issue	
Revision 1.2	2012-01-31
New chapter – UNI protocol	