...the broadest narrowband money can buy

RACOM
RADIO DATA NETWORKS

Introduction in the MORSE system

MORSE Guide 1

version 7.45
9/17/2007

Table of Contents

List of Figures

1. Introduction

1.1. Properties of the MORSE system

MORSE is a packet telecommunication system. It transfers data packets over radio channels and IP networks.

- Network created for a given application exclusively for the transmission of data

- Packet network is effective for very frequent transmission of short messages

- High transfer speed = high transfer capacity and reliability. The fact that individual messages are transferred over a short time reduces the risk of messages being affected by interference.

- Modulation transfer speed of radio modems in the MORSE system is 21.68 kbps for a 25 kHz channel in the 140-470 MHz range and up to 192 kbps for a 500 kHz channel in the 900 MHz band

- Rx/Tx switching speed < 1.5 ms

- Integration of any transmission media - LAN, WAN, GPRS, satellite, etc.

- 5 user ports + Ethernet available on each modem, integrated digital/analog I/O module

- Tens of supported industrial protocols

- Options include remote configuration, hot-line servicing, remote supervision and network administration, network diagnostics using RANEC supervisory SW - it is possible to continuously monitor the quality of transmission, optimise the network and react in time to deteriorations in the properties of connections before there is an actual break in connection. Remote servicing allows the causes of most faults to be uncovered, to decide whether there is a defect in the communication network or in the connected device, and to quickly and cheaply remove the majority of defects. Records are accessible in HW components of the MORSE system which contain extensive information about operation, the status of transmission channels (repeated packets, lost packets, number of packets, etc.), interference, etc. typically for the past 3 days of operation. Testing and monitoring the quality of individual connections

- MORSE system radio modems are installed with special software designed for testing and measuring the quality of data transfer, simulating operating loads, etc. The MORSE system constantly (evening when listening to user operation messages) measures and stores in memory the audibility of stations, the strength of signals, data quality, and the average and pulse level of interference, etc.

- Possibility of creating very extensive mobile networks

Further information about the Morse system can be found on www.racom.eu[1].

---

[1] http://www.racom.eu/cz/products/morse_main.html

1.2. What you will need

Pages www.racom.eu contain a lot of useful informations for the work with MORSE system. You can find them in this way:

Article content	*The way to it*
Setr - detailed description	Support, Firmware, Setr-firmware[2]
Setr for MR25 - detailed description	Support, Firmware, Setr-firmware for MR25[3]
Applications Setr, Netcnf, Memcp, Netlock	
description	Support, Utils, MORSE Utilities[4]
download	Download, Archive, Firmware&Utils[5]
Application Ranec, detailed description	
description	Support, Network Management[6]
download	Download, Archive, Ranec[7]
Download firmware in modem	
description	Support, Utils, Download firmware[8]
download	Download, Archive, Firmware&Utils[9]
Communication protocols	
descriptions	Support, Interface Protocols[10]
download	part of the firmware in modem
MORSE Guide - learning manual:	
Intoduction, routing, examples	Support, MORSE system-desc., MORSE Guide 1[11]
Mobile mode, diagnostics, protocols	Support, MORSE system-desc., MORSE Guide 2[12]
Ethernet in MORSE	Support, MORSE system-desc., MORSE Guide 3[13]
Manuals for products	
MR400	Support, Radiomodems, MR400, Manual[14]
MS2000	Support, Power supplies, MS2000, Manual[15]
labeling	Download, Production code[16]

---

[2] https://www.racom.eu/eng/support/firmware/mr400/index.html

[3] https://www.racom.eu/eng/support/firmware/mr25/index.html

[4] https://www.racom.eu/eng/support/utils/index.html

[5] https://www.racom.eu/eng/download/archiv-fw.html

[6] https://www.racom.eu/eng/support/ranec/index.html

[7] https://www.racom.eu/eng/download/archiv-ranec.html

[8] https://www.racom.eu/eng/support/download/index.html

[9] https://www.racom.eu/eng/download/archiv-fw.html

[10] https://www.racom.eu/eng/support/protocol.html

[11] https://www.racom.eu/eng/support/morse-m1/index.html

[12] https://www.racom.eu/eng/support/morse-m2/index.html

[13] https://www.racom.eu/eng/support/morse-m3/index.html

[14] https://www.racom.eu/eng/support/hw/mr400/index.html

[15] https://www.racom.eu/eng/support/hw/ms2000/index.html

[16] https://www.racom.eu/eng/download/morsecode.html

2. MORSE Guide

2.1. What does a simple MORSE network look like.

The following picture is used to give an initial idea about the MORSE network. Information transmitted in the form of data packets comes from the connected device A along the wire link to "radio" CU1. This "radio" is a complicated piece of equipment that checks the packet, modifies it and supplements it with records necessary for further transmission over the radio channel. This is why it is labelled as a communication unit (CU).
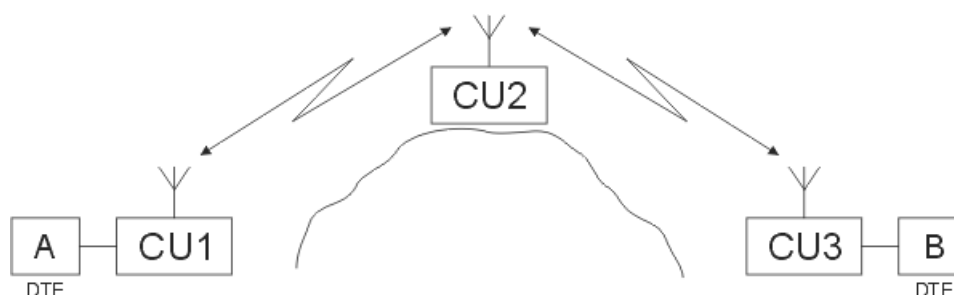

Fig. 2.1: Simple network MORSE

Amongst information added to the packet is the first section of the route which the packet will travel through via the net environment to the destination CU3. In our example CU3 is out of reach of direct audibility of CU1, therefore the packet is routed through CU2. This communication unit, as for all the others, is equipped with intelligence, which is used to take care of the packet along the next section of its route. CU2 thus receives the packet from CU1 and sends it on to CU3. This method of transmitting packets can be called store and forward relaying. At the destination CU3 the packet is sent along the wire link to the external device B.

A radio network is generally more branched. Connection is made between individual points in the direction of lines marked in the drawing.

Besides these routes it is often possible to achieve connection by further combinations, for example between CU3 and CU6. Whichever links will be used during operation is determined by the quality of the connection, according to loading of the network and according to the structure of addresses used in the network. Each CU has information about the choice of routes in its routing table (also store and forward table) and according to these it then decides about routing obtained packets.
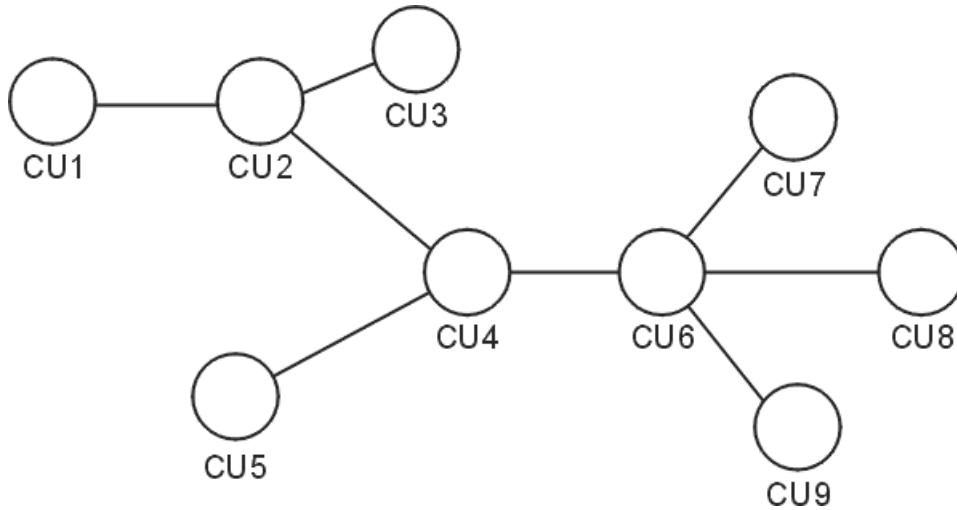
Fig. 2.2: Branched network

2.2. MORSE addresses

Connection-oriented and Connection-less Transmission of Data

The network can transmit data in two ways. For connection-oriented transmission a route is created in advance by connecting individual sections of the network and the route created is then available for bi-directional transmission of data. An example is the telephone network. This method is suitable for the transmission of a large amount of data in one transmission. Other participants must wait during this period.

The second method is connection-less transmission which is also used in the MORSE network. Packets with data are sent to the network here. Each packet is provided with the destination address and travels through the network by itself. The length of the packet is limited to 1500 bytes, so transmission between two CU's runs from 0.1 to 1 second depending on the length of the packet and other conditions. Another packet can then pass through the same section of the route to another destination. The suitability of non-linked transmission for frequent routing of short messages to a large number of participants comes from this.

Non-linked transmission including addressing can be likened to the postal service. In this case information is sent with the address of the network of post offices which gradually regulate its progression to the addressee. Let's try and compare analogical terms:

Letter	Packet
Postal address	Packet header with address
Letter text	Data
Signature	CRC
Registered letter	Acknowledged receipt of packet
Letters can overtake each other	The same order of registered packets is not guaranteed
Sorting of letters before delivery	Routing of packets

Address

The MORSE address contains 4 levels for determining the destination in its 4 bytes. They are termed:

```
global - net - wide - local
```

An example of an address is given below:

```
690F0A11
```

It contains 4 bytes with each created from two hexadecimal symbols:

- `69`– global (country specification, 69=CR, 6A=Slovakia...)

- `0F` - net (area in the country or other closer definition of the network)

- `0A` - wide (subnetwork, further specification)

- `11` - local (specific CU user)

Each byte can acquire a value of 0 to 0xFF (0 to 255 dec), the overall theoretical number of addresses is thus 256^4 which is more than 4 billion. The address architecture is important for the accessibility of individual networks, especially at the remote supervision of the network functionality. If not stated otherwise the MORSE address is recorded using eight hexadecimal symbols.

Work with the address is described in paragraph routing.

The internal layout of the modem from the point of view of configuration is contained in paragraph Communication unit.

2.3. Explanation of terms used

Packet

A packet is a sequence of bytes ordered according to precise rules.

Fig. 2.3: Data packet
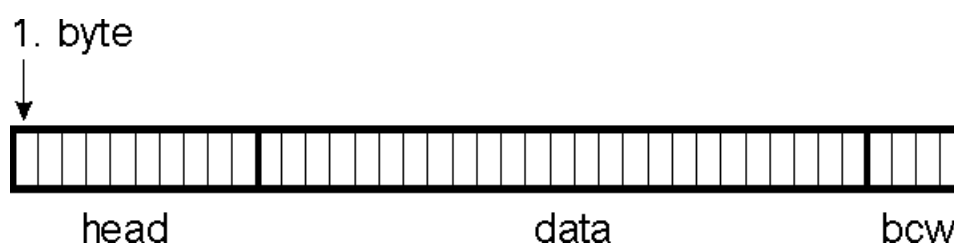
- At the front of this sequence is a group of bytes which contains basic information about the packet (type, length, who determined for, etc) and this is called the head.

- Behind this other bytes follow which carry necessary information (data monitored by measuring devices, GPS position of vehicle, text message, command for remotely operated pump etc.). This part is labelled as data.

- A packet is generally ended with a CRC which serves for data integrity checking.

The particular form of the packet depends from the communication protocol used.

What is the range of the MR400

The range or audibility of the radio signal in the 400 MHz band is influenced by the shape of the terrain and other obstacles. Under unfavourable conditions (city, valley) the range of the MR400 is several hundred metres or several kilometres. Under favourable conditions (direct visibility, good antenna) good quality operation is possible to distances over 100 km.

Network Medium

The medium used for packet transmitting enables such broadcasting of messages where the transmitted message can be picked up by more receivers. Therefore the message must carry information about which of the receivers must process it further. In our case it is the medium of radio signals or the Ethernet network medium.

The other media type only allow signals to be transmitted between two participants and can be called a link media. An example is the wire communication line protocol RS232 which is used for the connection between the CU and external devices.

Dispersed Intelligence

Operation in the network medium can be controlled in principle in two ways:

The network can only contain one centre which controls routing of messages, for example a telephone exchange. Participants then only request the centre to mediate the routing of messages and the centre undertakes everything necessary.

The second possibility is that the network is composed of elements from which each is able to process the message in the necessary way. This means deciding where the message should be routed and route it to the appropriate channel (which here is the radio channel, Ethernet, wire communication line, output port, service channel) and check the data integrity. The MORSE system is composed of a CU equipped in this way. All CU are the same and differ only in the setting of configuration parameters. Therefore the MORSE network is easily expandable or adjustable.

3. Communication Unit

The MORSE communication unit layout is in principle equal for different devices like MR400, MC100, MR900, MR25, MR25ET, sw MORCE or WALRUS. This devices can differ with some blocks presence or it's number but the configuration procedure stays the same. The diagrams and descriptions presented corresponds with the products family MR400 if it is not written differently.

CU (Communication Unit) can be represented from the point of view of configuration as:
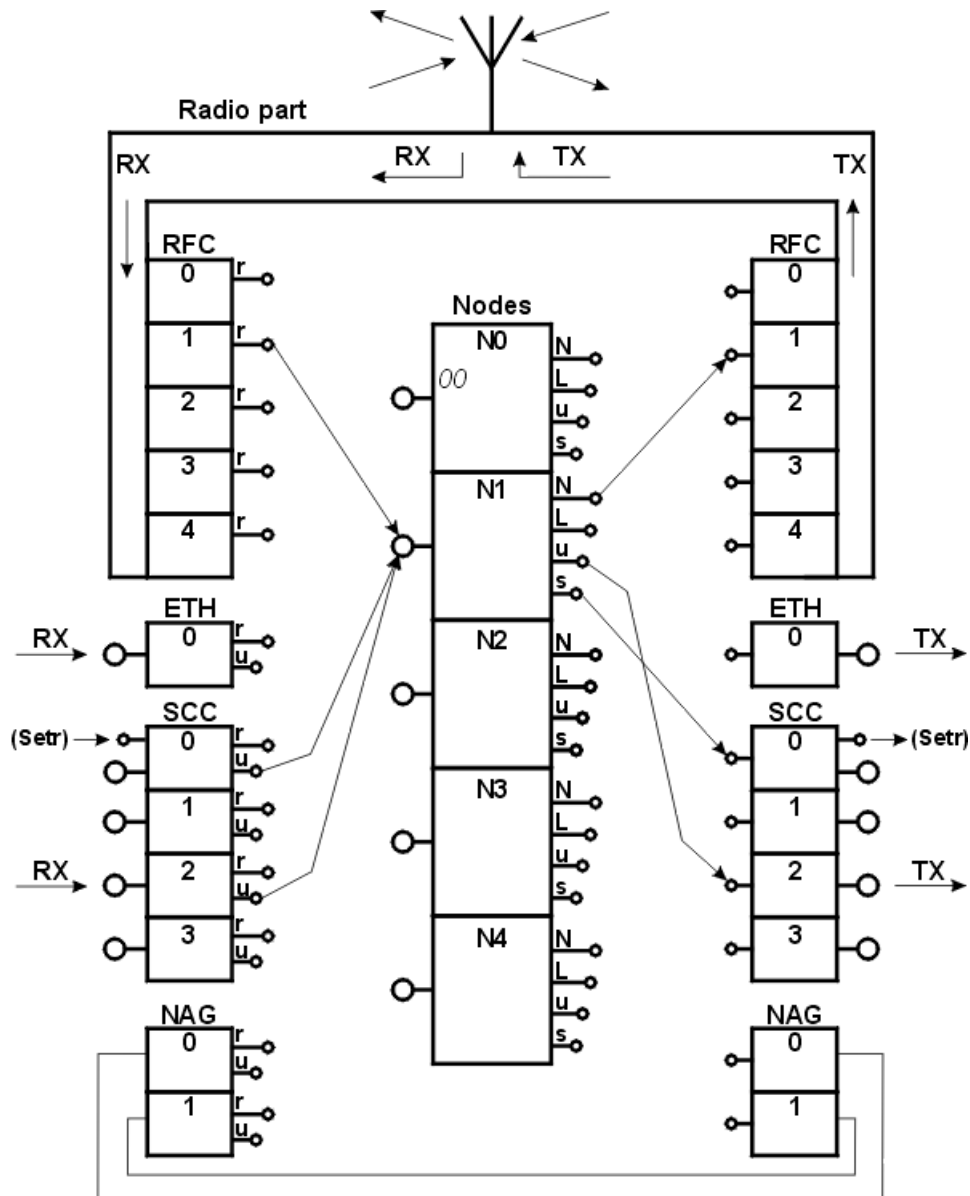

Fig. 3.1: Communication Unit MR400 layout

Individual blocks are always fitted with one input (from the left) and one or more outputs (to the right). Packets pass through the diagram from left to right. An exception is the antenna terminal above, which the received signal passes through into the CU and alternately the sent signal passes out through to the antenna.

In the centre there are **5 nodes**, which are the basic centre controlling communication. In reality they aren't boxes with terminal but parts of the program in the processor. From the point of view of configuration it is 5 independent control units, each with 1 input point and 4 outputs. Inside they are equipped with decision-making logic for controlling the flow of packets (routing). Each node can have a different address assigned to it under which it enters the MORSE network (thus one CU can be part of several independent networks).

The **Radio part** is marked in the upper part of the picture. It is common for the whole CU and individual nodes use it alternately. Its outputs pass through the radio-frequency channel (RFC), which according to the set rules enable routing of packets from nodes to transmitting radio parts. The receiving part of the RF channel transfers packets received from the RF medium to the node to which it is connected by the set configuration.

Serial Communication Channels (SCC) drawn in the lower part receive packets from external devices, modify them to the standard MORSE packet form and route them to the node. In its output part it then enables the output of packets from the node to the user device.

Ethernet Channel (ETH) enables input and output of packets between the Ethernet network and the MORSE network node.

Network agent (NAG) is a virtual channel without the physical output. It receives the packets from the node similarly like other channels, proceeds them and sends them again to the input of some node.

The mentioned blocks are connected via communication lines according to the selection in configuration. Here, for example, at the input of node 1, 3 lines converge from RFC1, SCC0 and SCC2, and from its outputs 3 lines lead out in various directions. As a rule more lines can converge at the input however at the output only one direction of further progression of the packet can be defined. Lines are configured independently for receiving and transmitting. For example the connection from SCC0 to node 1 is different than from node 1 to SCC0

Notes on configuration

Nodes are numbered 0 to 4. Node 0 has an address firmly assigned. It is the serial number, which always starts with 00, for example 0049B897. According to this number (address) it is possible to identify a CU at any time and by means of a so-called path-packets it can be used for establishing a connection. The address can be arbitrary assigned to other nodes and used for normal communication. There are four node outputs:

N	net	Network output to the radio network medium, Ethernet or net lines
L	line	Line output to wire link between two CU´s
u	user	User output for user data
s	service	Service output for Setr service terminal

SCC, ETH and NAG channels have a routing output, which is used for transferring packets inside the MORSE network and a user output used for packets which have come from outside to the MORSE network.

RFC channels have only a routing output for transmitting packets through the radio network. The user output would be of no use to the RFC.

Configuration parameters, i.e. the above-mentioned connections between channels and nodes and many other parameters that influence functions of the MORSE network are set using the service terminal. It is a normal PC connected by serial cable or ethernet and running the SETR.exe program, freely supplied by Racom.

4. Service terminal

The Service terminal is a program used for handling the CU of the MORSE system by a personal computer. It allows the setting of CU parameters, the monitoring of traffic on individual channels, the running of tests, the reading of records about past traffic and other functions. It is part of the applications group for handling a MORSE network and is accessible on www.racom.eu.

Choose **Download** ▷ **Software** on the main page or **Download** ▷ **Archive** for the older recommended versions and choose the appropriate directory:

- **utils/actual/** - here we find the current control applications file for the CU. According to the operating system in use we select the appropriate file; for Windows the file is `utils-win-gui-9.0.19.0.exe`. Number 9.0.19.0 indicates the software version and this number incrementally increases with development.

- **firmware/actual/** - firmware modules contained in radio modems; these do not need to be loaded as one of these modules is already loaded at manufacturer. Next work with the firmware see the article Download[1].

- **utils/release-candidate/** - these directories, accessible at higher levels of access via the Internet, contain new, not fully tested software

After choice utils/actual/ next menu appears:

Fig. 4.1: Menu utils/actual

Since the 9.0.16.0 version the .exe file is avilable, here `utils-win-gui-9.0.19.0.exe`. It is suitable for the automatic installation into PC. Copy the file in the PC and start it. If you reply NO after request about backup, then the old Utils set will be overwritten. If the reply is

---

[1] https://www.racom.eu/eng/support/download/index.html

YES, then it is saved in the directory Utils-xxxx. The directory `c:\Racom\Utils\` containing all important files is created. Setr can be started also from the Windows Start menu **Start ▷ Programms ▷ Racom ▷ Utils ▷ Setr**.

The other possibility is to use the file `utils-win-gui-9.0.19.0.zip`. Unpack it into suitable directory. This sub-directory contains more files, however the absolute minimum requirement for work with the CU is `setr.exe`.

Files with firmware modules for the CU can be downloaded from directory `firmware/actual/`. The directory contains two sets of modules:

- files `m1r-757-mx.zip` (formerly mr25-721-mx.zip) are designed for MR25, MR25ET, MCM302

- files `m2r-9.0.19.0-cx10.fkl.zip` (formerly mr900-721-mx.zip) are designed for MR900, MR400, MR300, MR160, MC100, MG100

Create the subdirectory `\fkl\` into directory `c:\Racom\Utils\` and unpack in it the needed files from `m2r-9.0.19.0-cx10.fkl.zip`. Here it is e.g. `ce10.fkl` and `cb10.fkl`. Example of the location of files:

For controlling the CU we use utility:

Utility	Description
Setr	Software for controlling the CU
Memcp	Loading software to the CU
Netcnf	Upload/download CU configuration
Netlock	Locking the CU

Note

For a description of downloading a firmware program module to the CU please see www.racom.eu, section Support/Software/Utils/Download firmware[2].

Recommendation

For problem-free configuration and control of the CU we recommend using the same version of firmware in the CU and control utilities (Setr, Memcp and others).

---

[2] https://www.racom.eu/eng/support/download/index.html

A service cable serves for connecting the PC to the CU and has contacts connected in the following manner:
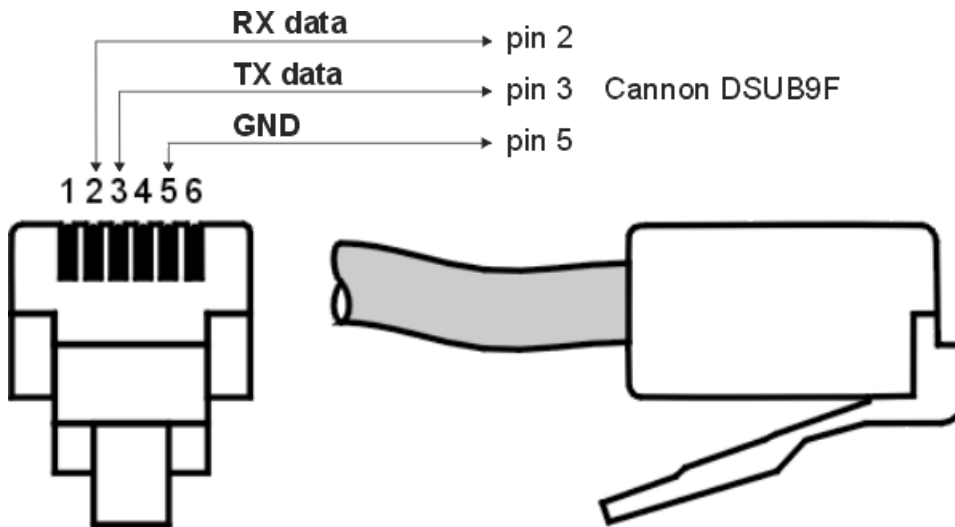
Fig. 4.2: Service cable DKR23 for MR400, rx and tx signals
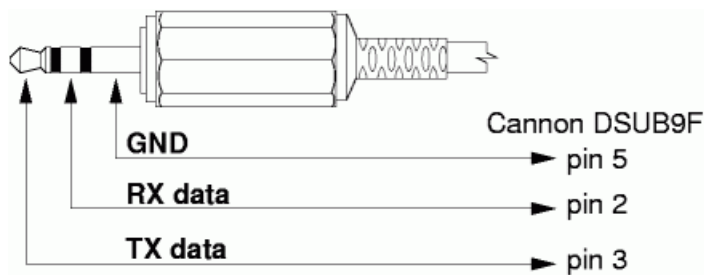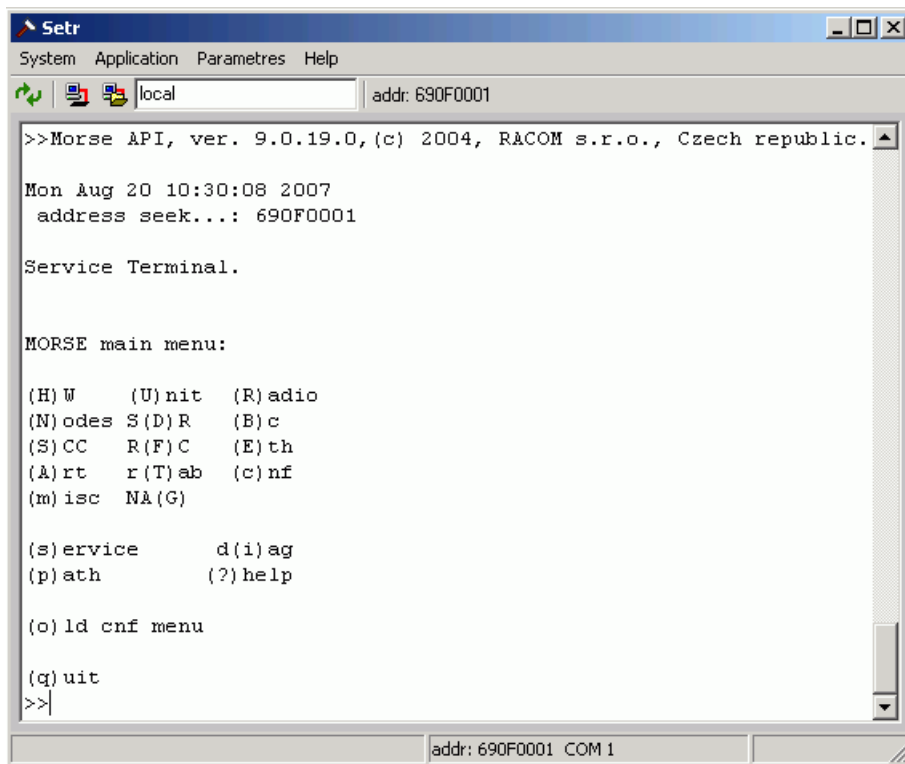
Fig. 4.3: Service cable DKR12 for MR25

For connecting the CU to the PC using a serial cable connect the SERVICE connector of the modem to port COM1 or COM2 on the PC. Also see Service manual[3], chapter service connector[4].

Now we can start `Setr.exe`. If the service cable is connected to the PC via another port than com1 then we run Setr with the appropriate parameter, e.g. `setr -p2`. The service terminal screen then appears. Above there is the Setr version number (`ver. 9.0.19.0`), at the lower edge there is the address of the connected node in the CU and the number of the PC port. The main content of the screen is the MORSE main menu, from which it is possible to fully control the CU.

---

[3] https://www.racom.eu/eng/support/hw/mr400/index.html
[4] https://www.racom.eu/eng/support/hw/mr400/ch03.html#serv_konektor

Fig. 4.4: Setr after connecting to CU

Warning

For the duration of the connection of the service cable port SCC0 is disconnected in the CU in the case of MR400, or SCC2 is disconnected in the case of MR25.

5. MORSE Main menu

MORSE main menu contains all functions for controlling the CU.

```
MORSE main menu:

(H)W     (U)nit  (R)adio
(N)odes S(D)R    (B)c
(S)CC    R(F)C    (E)th
(A)rt    r(T)ab   (c)nf
(m)isc   NA(G)


(s)ervice      d(i)ag
(p)ath         (?)help


(o)ld cnf menu


(q)uit
>>
```

5.1. Managing the menu

Work with the menu and its options is described in the *MORSE Firmware Documentation*. The first part of the menu serves for setting configuration parameters, the second for service and diagnostic work and the last allows older versions of software to be controlled. Individual commands are selected by pressing the letter given in brackets. For example, entry of the node address is done as follows:

- press N for selection of command Node, press Enter

- press e for selection of command edit, press Enter

- message ending with O.K. provides notification that the existing parameters were transferred from the CU to Setr

- press Enter to display them

now we can see the state of parameters of menu Nodes:

```
  Nodes:
                         retab
Nid|address |M | u   s | L   N |l w n g H|sTO Err  Cent vTO hTO
(0) 004AE97E    -  S00| -  R00|0 0 0 0| 15 SERV  OFF 304  30
(1) 690F1109   S02 S00| -  R01|1 1 1 1| 15 SERV  OFF 304  30
(2) 00000000   S01 S00| -  R02|0 0 0 0| 15 SERV  OFF 304  30
(3) 00000000   S02 S00| -  R03|0 0 0 0| 15 SERV  OFF 304  30
(4) 00000000   S03 S00| -  R04|0 0 0 0| 15 SERV  OFF 304  30


de(f)ault (r)ead (w)rite
(I)nit (S)ync
```

```
(q)uit
>>
```

We can see that node 1 already has the address `690F1109` inserted and routing tables local, wide, net and global assigned, each with serial number 1. The user output from node `690F1109` (row (1), column u) leads to port SCC2. Other parameters have a recommended initial value, which is the value that appeared after selecting the command `de(f)ault`.

In order to insert for example address `69112233` to node 4:

• press `4 Enter`, then follow help on screen

• press `a Enter`

• insert address `69112233 Enter Enter`

• return back 1 level `q Enter Enter`

• now the new address which is currently only in Setr can be transferred to S-RAM permanent memory in the CU `w Enter`

• after receiving confirmation of entry `O.K.` we can write the new address using command `I Enter` also into RAM memory, thus setting it into operation

The whole operation does not seem very transparent; however after gaining a little experience we can do this from main menu as follows:

```
Ne  Enter (O.K.) Enter
4a  69112233  Enter Enter
w   Enter (O.K.)
I   Enter (O.K.)
Q   Enter (return to MORSE main menu)
```

Basic principles of configuration

are quite evident from the CU layout diagram:
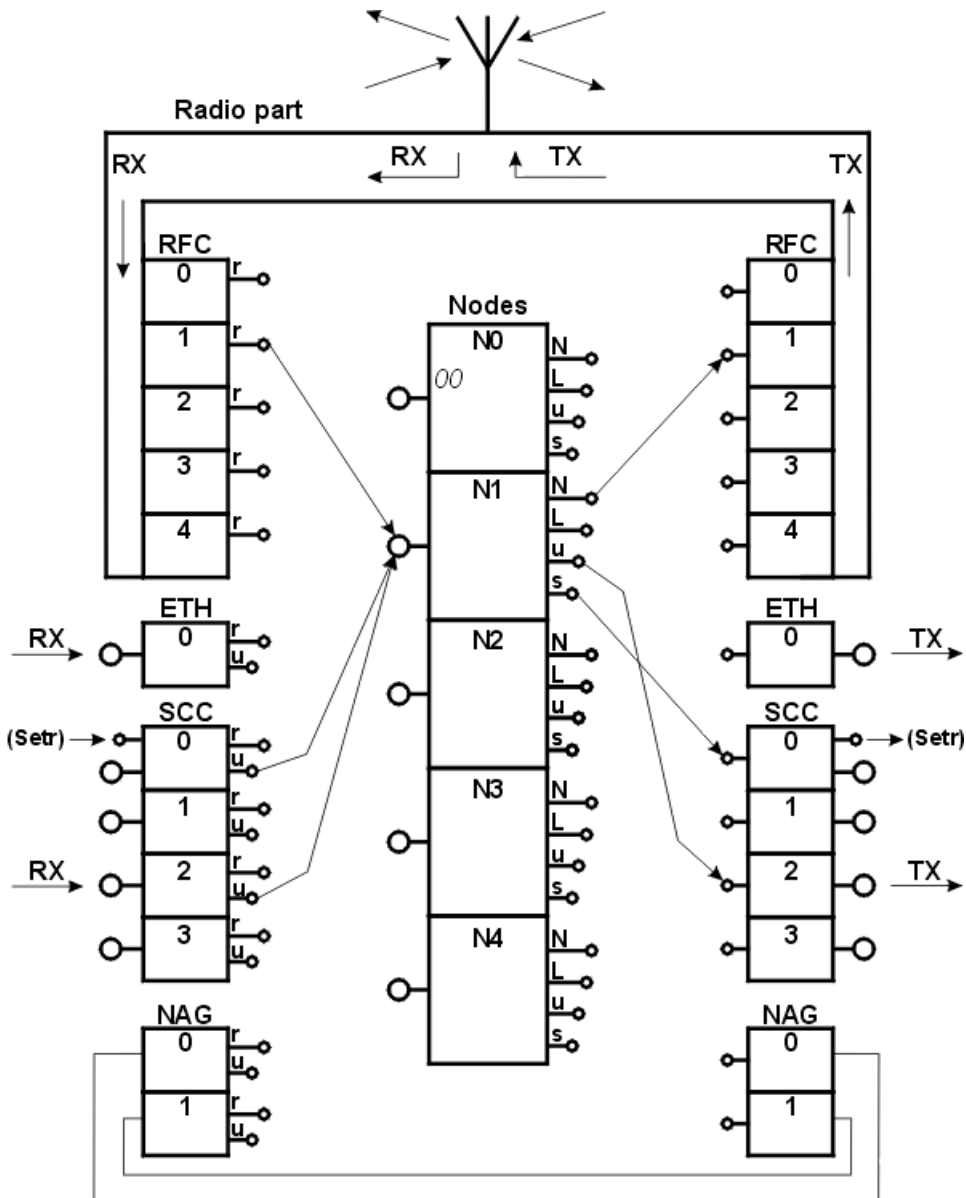
Connection from the node to the communication channel is set in the previously mentioned menu `Ne`, i.e. `(N)odes (e)dit`:

```
 Nodes:
                             retab
Nid|address |M | u    s | L    N |l w n g H|sTO Err   Cent vTO hTO
(0) 004AE97E      -   S00| -   R00|0 0 0 0 -| 15 SERV  OFF  304  30
(1) 690F1109     S02 S00| -   R01|1 1 1 1 -| 15 SERV  OFF  304  30
(2) 690F1205     S03 S00| -   R02|2 2 0 0 -| 15 SERV  OFF  304  30
(3) 00000000     S02 S00| -   R03|0 0 0 0 -| 15 SERV  OFF  304  30
(4) 00000000     S03 S00| -   R04|0 0 0 0 -| 15 SERV  OFF  304  30
```

Each node has one row and its columns `u,s,L,N` define where the respective output is directed. Row (1) corresponds to connections in the figure. Another 4 columns `l,w,n,g` determine using numbers 1 to 4 which of the prepared routing tables will be used by the node when resolving routing.

Connection from the SC channel to the node is selected in menu `SIe` :

```
Channel to Node Interface:
    retranslation     |   user+service              lim
 id N  A t         m | N  A t Base      m  sec brc S   e
(0) 0   NO AR        | 1   NO AR            usr OFF NONE
(1) 0   NO AR        | 2 MASK 00000000/08 usr OFF NONE
(2) 0   NO AR        | 1 MASK 00000000/08 usr OFF NONE
(3) 0   NO AR        | 0 MASK 00000000/08 usr OFF NONE
```

Again each channel has 1 row; the `retranslation` group of columns serves for routing output `r` and the `user` columns are determined for routing output `u`. Basic information is carried in column `N`, which determines which node the output is routed to. Rows (0) and (2) correspond to connections in the figure. Other items in the table will be explained in the SC channel properties definition.

Similarly menu `FIe` is used for routing outputs from the radiofrequency channel:

```
Channel to Node Interface:
    retranslation     |   user+service              lim
 id N  A t         m | N  A t Base      m  sec brc S   e
(0) 0   NO AR        | 0   NO AR            usr OFF NONE
(1) 1   NO AR        | 0   NO AR            usr OFF NONE
(2) 2   NO AR        | 0   NO AR            usr OFF NONE
(3) 3   NO AR        | 0   NO AR            usr OFF NONE
(4) 4   NO AR        | 0   NO AR            usr OFF NONE
```

The only difference is in the number of rows, according to the number of RF channels, and in that the user columns are not used here. Row (1) corresponds to connections in the figure. In defining routes from the RF or SC channel to the node the address of this node is simultaneously assigned to the channel. If a packet is then received through a radio part then according to its address it is determined whether is pertains to any of the RF channels. If not then it is not processed any further. If yes then it is transferred to the node along the route configured in this menu.

Menu `EIe` behaves in the same way for the Ethernet channel and menu `GIe` for the Network Agent channels.

For a better understanding of the above-mentioned configuration connections

see the figure below of three communication units and connections, forming a simple radio network. Connections used along the packets route are also included:
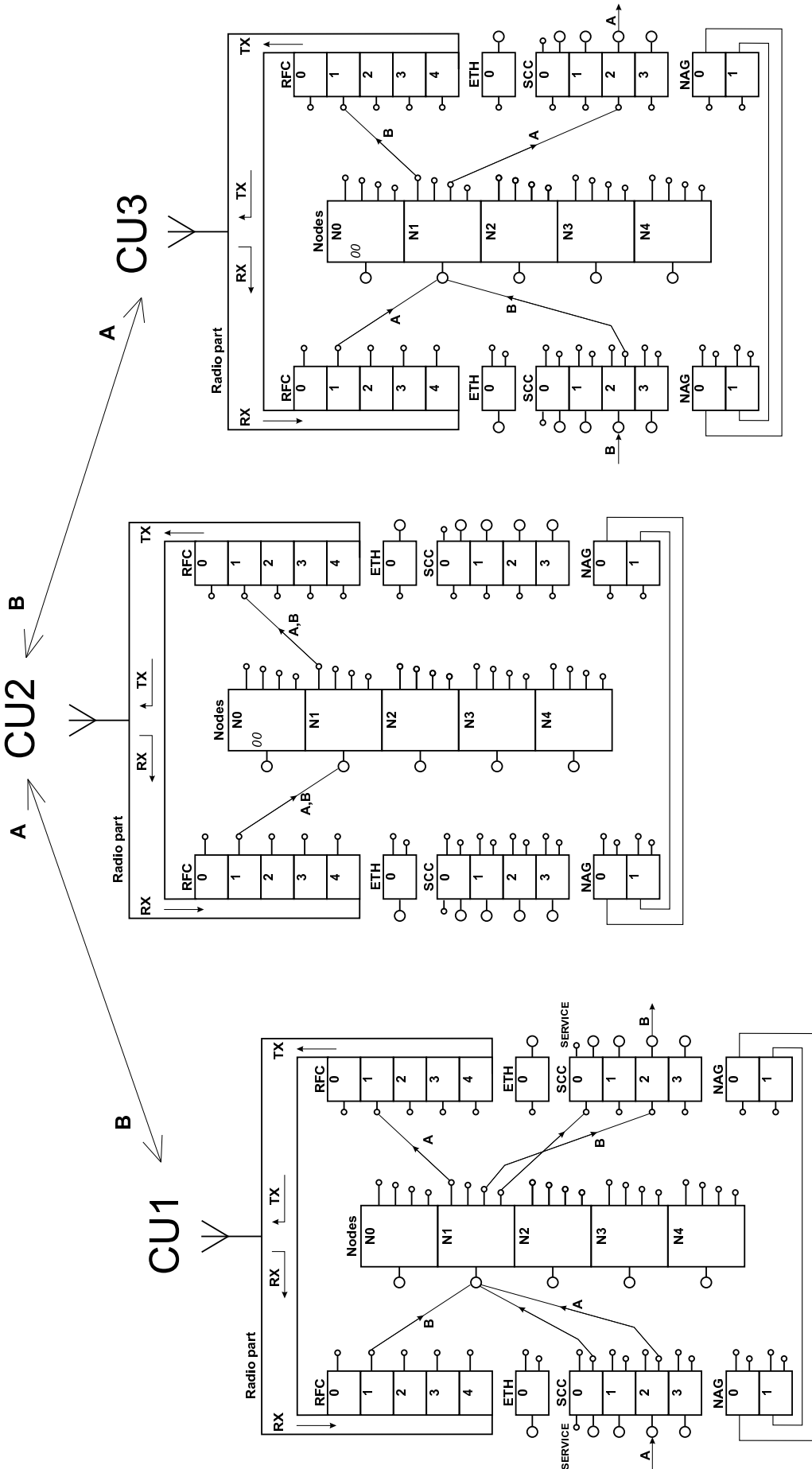
Fig. 5.1: Packet transfer via the configured connections

The packet A is sent from CU1 to CU3:

- The packet passes through channel SCC2 (received =RX) to CU1. Node N1 sends it via radio channel RFC1.

- Radio part CU2 receives the packet, channel RFC1 sends it to node N1 and from there it continues through the transmitting part RFC1 to the antenna. This step is termed store and forward relaying.

- At the destination CU3 the packet is received by the radio part, it is transferred to node N1 by channel RFC1 and from there to the serial channel SCC2, through which the packet leaves the MORSE network (sent through channel SCC2 =TX).

The packet B is sent from CU3 to CU1:

- The packets route in the opposite direction from CU3 to CU1 is analogical. It starts with the arrow entering SCC0 in CU3 and ends with the arrow leaving SCC0 in CU1.

- Connections connecting SCC0 with node N1 are also drawn. These serve for routing service packets (for example upon configuration setting) between Setr and node 1, because the service cable on MR400 is connected inside CU to SCC0 for this purpose. Similar connections are in each CU, so that access from Setr to each communication unit is possible using the service cable.

As shown in this example the node is the centre of the design-making operations for controlling the route of packets through the network. Introduction to routing gives a definition of these operations.

6. Introduction to Routing

A packet entering the MORSE network must be provided with a destination address. This address is assigned to it in the user device A or B. This external device can determine the address for example as only a serial number from 1 to 5 if it works in a network containing 5 stations. Transfer of addresses to the MORSE form is then taken care of in the input part of the CU, most often the SCC serial communication channel.
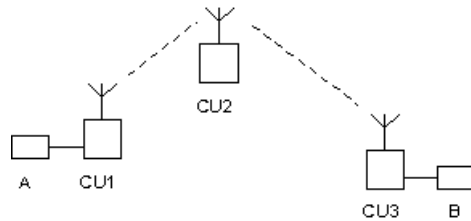
Fig. 6.1: Simple network diagram

This destination address (shortened to dest or dst) is carried in the header of the packet along the whole route through the network. It serves as initial information for each CU during routing, i.e. upon determining the next step of the packets route. The packet also carries a source address with it (src), according to which the destination CU recognises whom it should send a reply. For example, see next drawing, a packet sent in the subnetwork `69AABB00` from CU1 to CU6 thus has the src address `69AABB01` and dst address `69AABB06`. The route of the packet is divided into a number of steps between neighbouring CU, here 1 > 2, 2 > 4, 4 > 6. For each of these steps routing is resolved separately and the packet takes these two addresses of the current step with it - **from** is the transmitting CU and **to** is the receiving CU. Addresses contained in the header of the MORSE packet are thus:

src, from, to, dst
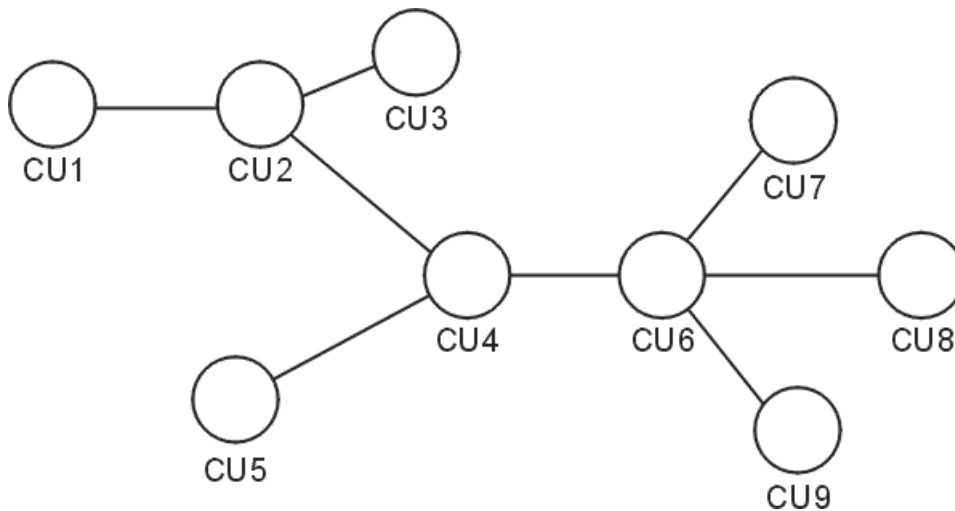
Fig. 6.2: The branched network

All routing operation run in any of the 5 nodes contained in the CU. In the node the address **to** is assigned to the passing packet in the manner described below and the packet is sent via any of the four node outputs on its next route through the MORSE network. The function of the node is described in the chapter Routing 1.
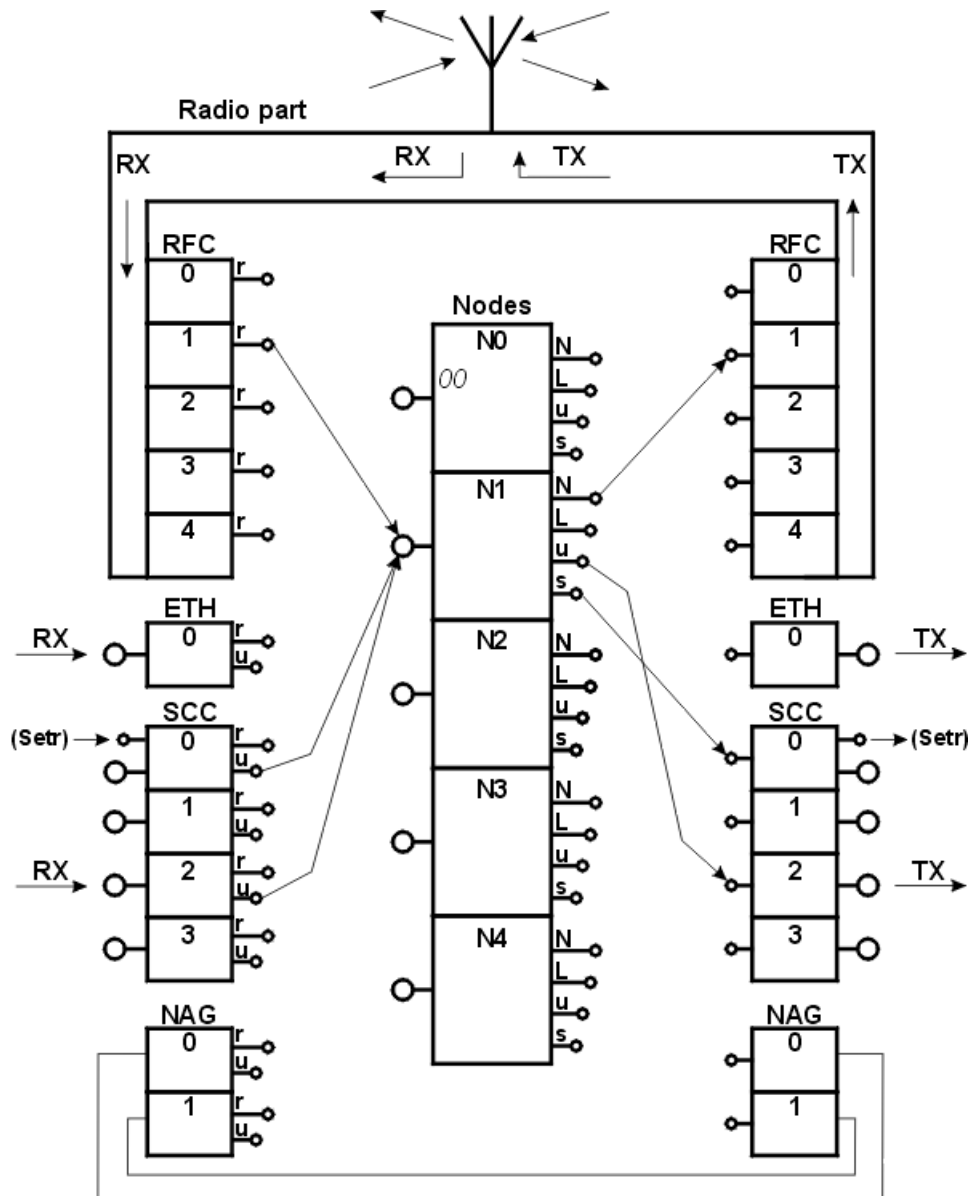
Fig. 6.3: The communication unit CU

For illustration here we give a record of the route of the signal through three nodes from address `690F120F` to address `690F1209` and back:

```
u S00   690F120F    R01
30/ 46  690F1200    R01
31/ 88  690F1209   serd


 serd   690F1209    R02
30/ 41  690F1200    R01
30/ 65  690F120F  u S00
690F1209h>
```

In the middle the route is marked by the sequence of addresses, on the right the channel is marked through which the packet left the node and on the left there is the strength of the received signal.

From the point of view of the internal construction an address is composed of 4 bytes called **global, net, wide** and **local**, see introductory chapter MORSE addresses. According to these parts of the address the node searches for the further information in the routing (or store and forward) tables. The order of these node activities is discussed in the following chapter Routing 1. Here you can see work with the routing tables.

From the MORSE main menu selecting:

```
>> T Enter
```

we get:

```
Retranslation table:
  (l)ocal
  (w)ide area
  (n)et
  (g)lobal
(q)uit
>>
```

We select which type of table we would like to work with, for example for table wide:

```
>> w  Enter

Wide retranslation table No:
  (1) (2) (3) (4)
(q)uit
>>
```

The table can be prepared in up to 4 versions so that each node can have a different table available. Select table number one:

```
>> 1 Enter

Retranslation table:
  (r)ead nontrivial paths
  (p)ath:0 via (n)ode:0
  (g)et p(u)t
  (c)lear  (e)dit
(q)uit
>>
```

Using the function `(r)ead` we can read the contents of the table:

```
>> r Enter

Wide retab. No 1
14to:1105 15to:150A
>>
```

Using the function `(c)lear` we can delete the whole table. An entry is made using `(p)ath` and `(n)ode` with a subsequent `(p)ut`. If an entry is made using the standard procedure as follows:

```
>> p 14 Enter
>> n 1105 Enter
```

then it is not necessary to use the command `p(u)t`.

Upon writing to individual tables this length of address is used for the item `(n)ode`:

```
 global
   69112233
 net
   69112233
 wide
   2233
 local
   33
```

The item `(p)ath` always has the length of one byte, e.g. 22.

Rules for creating the content of routing tables are described in the following chapter Routing 1.

7. Routing 1

Simplified description

Packet incoming to the node, contains it's target address DST. The address of immediate next step NN is determined in the node according to address DST.

The address DST is compared with address of the node (NODE ADDRESS), where the process runs. If they are equal then the packet is in the target node and will be sent, according to its type, to the user or service port or to the service processing. If they are different, then the routing tables processing follows.
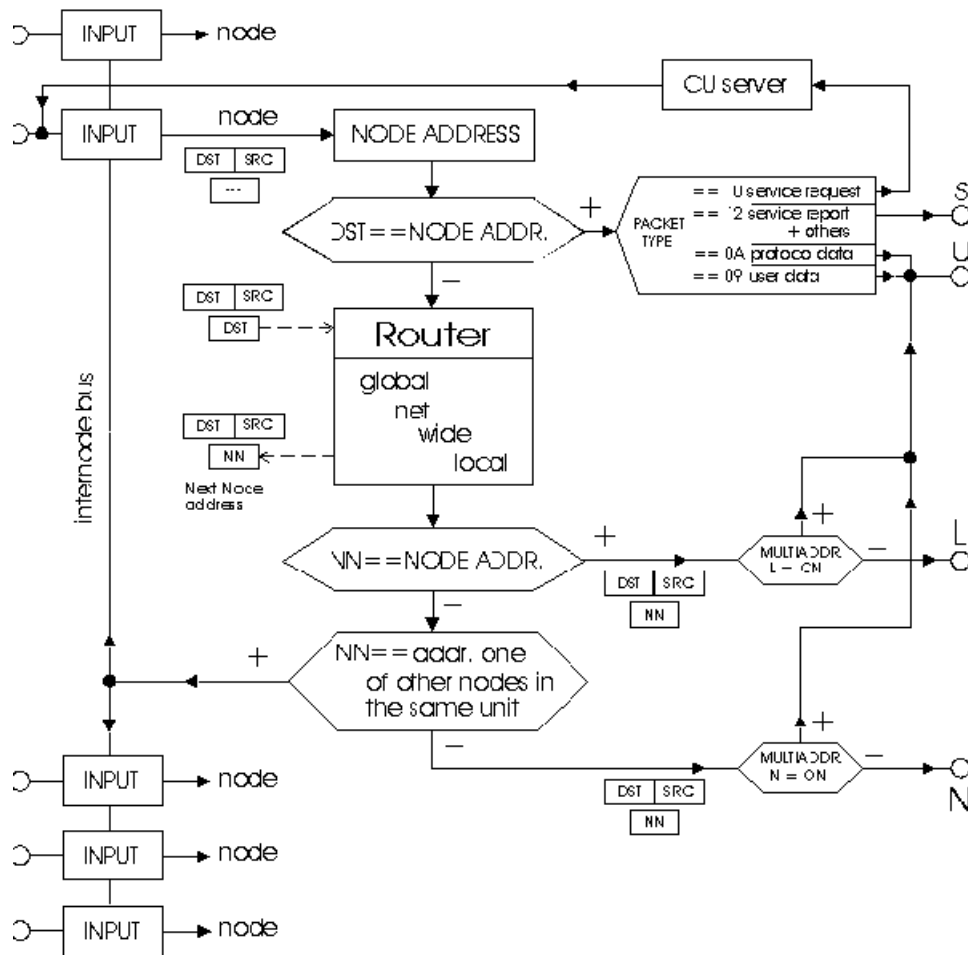
Fig. 7.1: Simplified node diagram

The packet address can be divided to the 4 parts called (from left) global, net, wide and local. The "global" part represents a big unit where the address belongs (country), part "net" determines more exactly the region inside the country (geographically or according to other points of view). Next specifying is in part "wide" and part "local" defines finally the actual address of node. At the choice of routing table the addresses DST and NODE ADDR are compared. The parts global, net, wide and local are compared gradually. If the difference is found (for example in

"net" part), then in the respective table (net) will be the address of next step found. The differences in lower parts (here wide and local) are then not important.

The routing table contains the pairs `(p)ath` and `(n)ode`. The `(p)ath` represents the compared part of the address (here net) and `(n)ode` is address of node NN where leads the next step of packet on it's way to DST address. The tables wide and local contains in `(n)ode` the lower address part only. The item `(n)ode` is the resultant information found in the routing tables and is inserted into address NN.

The packet equipped now by the NN address goes to the next processing according to diagram *Block circuit of MORSE node*. If the NN address equals to NODE ADDR, then the packet is sent to the link output L.

In next step is the address NN compared with addresses of other nodes in own CU. If the concordance is found, then the packet is sent through inner bus to the respective node.

If the packet is no diverted to the link output or inner bus, then it is sent to the net retranslation output N. The address NN becomes then the next step adrress TO in the MORSE network.

In this way the packet is sent to the net medium (most frequently radio channel) and for it's next processing takes charge this node, which address is equal to TO address contained in the packet.

The more detailed description is in chapter Routing 2. The particular routing examples follows in the next chapter.

8. Routing examples 1

8.1. Direct connection between two CU.

Let suppose, that in the communication unit (modem, radio) there is set the working frequency, power and that all parameters have the default value. Then we set the address `690F1230` in the menu `Ne` in one of them:

```
  Nodes:
                            retab
Nid|address |M | u    s | L    N |l w n g H|sTO Err  Cent vTO hTO
(0) 004AE97E     -  S00| -   R00|0 0 0 0 -| 15 SERV  OFF 304  30
(1) 690F1230    S00 S00| -   R01|0 0 0 0 -| 15 SERV  OFF 304  30
(2) 00000000    S01 S00| -   R02|0 0 0 0 -| 15 SERV  OFF 304  30
(3) 00000000    S02 S00| -   R03|0 0 0 0 -| 15 SERV  OFF 304  30
(4) 00000000    S03 S00| -   R04|0 0 0 0 -| 15 SERV  OFF 304  30


de(f)ault (r)ead (w)rite
(I)nit (S)ync
(q)uit
>>
```

Save all by the command **(w)rite** and **(I)nit**. After new address setting initialize the connection with Setr by command **Alt+I**. The same things make in the opposite station with this different, that the putted address is `690F1233`. The resulting state:

```
            _____        radio    _____
           |          |      channel  |          |
     PC ---| 690F1230 |..............| 690F1233 |
    Setr   |_____|              |_____|
```
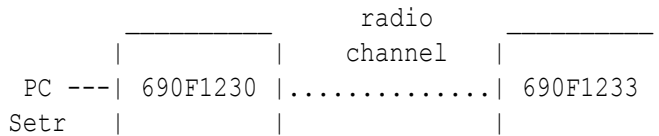
Then give back the service cable in the first CU, perform **Alt+I** and write:

```
    >>!h690F1233 Enter
```

Suffice it to write the last part of address only, which is different from the own or the last called address:

```
    >>!h33 Enter
```

Now put the command for the packet sending:

```
    >>! Enter
```

The packet is sent there and back and the connection report appears:

```
u S00    690F1230    R01
31/ 77   690F1233    serd

 serd    690F1233    R01
30/ 75   690F1230  u S00
690F1233h>
```

The address column describes the way of packet. On the left there is the incomming signal description (signal source or for radio signal it's quality and strength DQ/RSS). On the right there is the direction in which the packet leaved the node.

8.2. Store and forward connection.

In this case we need three CU. The assigning of routing tables put in the CU `690F1230`:

```
Nodes:
                              retab
Nid|address |M | u    s | L    N |l w n g H|sTO Err  Cent vTO hTO
(0) 004AE97E    -  S00| -   R00|0 0 0 0 -| 15 SERV  OFF 304  30
(1) 690F1230   S00 S00| -   R01|1 0 0 0 -| 15 SERV  OFF 304  30
(2) 00000000   S01 S00| -   R02|0 0 0 0 -| 15 SERV  OFF 304  30
(3) 00000000   S02 S00| -   R03|0 0 0 0 -| 15 SERV  OFF 304  30
(4) 00000000   S03 S00| -   R04|0 0 0 0 -| 15 SERV  OFF 304  30
```

We prescribe it the local table in CU `690F1230` , that the packets designated for the address `690F1233` should be sent to address `690F1200` which is audible for both station:

```
Local retab. No 1
33 to:00
>>
```

After command **!h33 Enter** we obtain next report:

```
u S00    690F1230    R01
30/ 68   690F1200    R01
31/ 68   690F1233   serd

 serd    690F1233    R01
30/ 71   690F1230  u S00
690F1233h>
```

We can see, that the packet from `690F1200` to `690F1233` runs through the store and forward station `690F1200` but in the opposite direction remained the direct route, because the routing table is not fullfiled in `690F1233` and the stations can hear each other directly. The connection scheme:

```
        PC---690F1230...>...690F1200...>...690F1233
               :                     :
               :...........<...........:
```

Now put also in CU `690F1233` the local routing table:

```
Local retab. No 1
30 to:00
>>
```

and assign the set of tables 1 also to the address `690F1233`. The packet runs now in both directions through `690F1200`:

```
u S00    690F1230    R01
27/ 68   690F1200    R01
30/ 68   690F1233   serd

 serd    690F1233    R01
31/ 69   690F1200    R01
30/ 65   690F1230  u S00
690F1233h>
```

The connection line is similar in both directions:

```
    PC---690F1230.......690F1200.......690F1233
```

Now both end points can be out of touch (the radio signal between them is too weak) but if each of them can hear the station `690F1200`, they can communicate together.

8.3. Distant access

The configuration modifications can be done by service cable but more often the distant access through the radio channel is used. We remain connected by the service cable to the local station and hand on the service packets by radio channell to the distant station. If we obtained in the last example after command:

```
>>!h33 Enter
```

Enter this response, then the last line

```
690F1233h>
```

announces, that we are switched in the `690F1233` station and all commnads, we are putted in, will be performed in this station. Notice the change shape of prompt from

```
>>
```

which refers to local access to

```
690F1200h>
```

denoting the access in the distant station. To return to the local access type:

```
690F1233h>!l Enter
```

Notice the longer response to service command for reading and writing into memory in the distant CU. Switch to the distant station

```
>>!h33 Enter Enter
```

and choose from the main menu

```
690F1233h>Ne Enter
```

Now takes place the service packet exchange between the stations, which takes according to the connection conditions from several tenth of seconds to several seconds and then the report appears

```
get  NODE 0 O.K.
get  NODE 1 O.K.
get  NODE 2 O.K.
get  NODE 3 O.K.
get  NODE 4 O.K.
690F1233h>
```

which says, that the menu `Node edit` was succesfully transfered from the distant station into our program Setr. Display this menu by `Enter` and write e.g. the new address and routing table assignement in the Node 2.

```
Nodes:
                          retab
Nid|address |M | u   s | L   N |l w n g H|sTO Err  Cent vTO hTO
(0) 004A1E27     -  S00| -  R00|0 0 0 0 -| 15 SERV  OFF 304  30
(1) 690F1233    S00 S00| -  R01|1 0 0 0 -| 15 SERV  OFF 304  30
(2) 690F1234    S01 S00| -  R02|0 0 0 0 -| 15 SERV  OFF 304  30
(3) 00000000    S02 S00| -  R03|0 0 0 0 -| 15 SERV  OFF 304  30
(4) 00000000    S03 S00| -  R04|0 0 0 0 -| 15 SERV  OFF 304  30


de(f)ault (r)ead (w)rite
(I)nit (S)ync
(q)uit
690F1233h>
```

These changes are yet done in our Setr and only by command **(w)rite** we send them into distant station. We obtain the acknowledgement about succesful writing in the flash memory:

```
write NODE 0 O.K.
write NODE 1 O.K.
write NODE 2 O.K.
write NODE 3 O.K.
write NODE 4 O.K.
690F1233h>
```

Now we perform similarly the writing in the RAM memory by command **(I)nit** and in this way we initialize the putted parameters in the distant station.

```
put   NODE 0 O.K.
put   NODE 1 O.K.
put   NODE 2 O.K.
put   NODE 3 O.K.
put   NODE 4 O.K.
690F1233h>
```

It could happend at bed connection or heavy radio traffic, that the acknowledgement `get`, `write` or `put` doesn't come. We repeat the command in this case. We return by command

```
    !l Enter
```

in the local station, refill the local routing table to the state

```
Local retab. No 1
33 to:00  34 to:00
>>
```

and by command

```
    !h34 Enter
```

examine the new connection:

```
u S00    690F1230    R01
30/ 68   690F1200    R01
31/ 68   690F1233   serd

 serd    690F1233    R01
30/ 69   690F1200    R01
29/ 65   690F1230  u S00
690F1233h>
```

The connection through more CU is presented in the chapter Routing examples 3.

9. Routing examples 2

9.1. Serial line

The packet transmittion between two CUs by a serial cable (called the wire retranslation too) is used to connect two parts of a network working on different frequencies.

In our example, the packet comes through the network on the frequency 410 MHz into CU5, then it goes through RFC1 to the node N1 having the address `690F1230`. From there by the link output L to SCC0 and then by the serial link to CU6. It inputs CU6 through SCC1, by the retranslation output r of SCC1 to the node N1 having the address `690F1240`, by the net output N through RFC1 to the antena and then continues on the frequency 420 MHz.
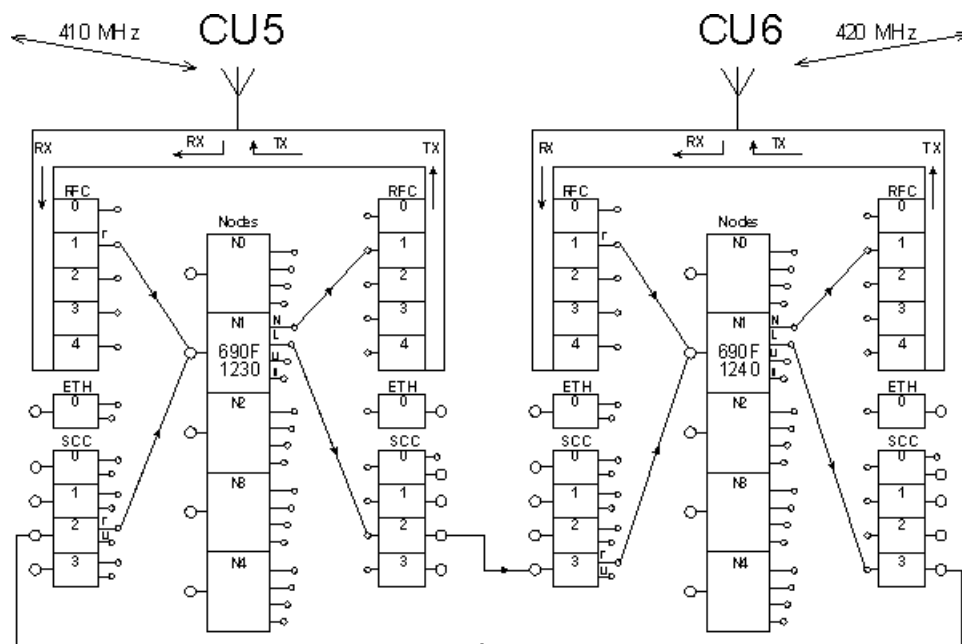
Fig. 9.1: Retranslation link connection

The necessary configuration:

• The link output is routed to chosen SCC. The retranslation output from SCC is routed to the node.

• The protocol MARS-A is configured on the respective node.

• The way to the own address is written in the routing tables, see the description in the Routing 1.

• The serial channels of both CUs are connected by the three-wire link (RX,TX,GND). The RX clamp on one CU is connected to the TX clamp on the others, so called crossed line.

The example of configuration CU5:

```
Nodes:
                             retab
Nid|address |M | u    s | L   N |l w n g H|sTO Err  Cent vTO hTO
(0) 004AE97E      -  S00| -  R00|0 0 0 0 -| 15 SERV  OFF 304  30
```

```
(1) 690F1230    S00 S00|S02 R01|1 0 0 0 -| 15 SERV  OFF 304  30
(2) 00000000    S01 S00| -  R02|0 0 0 0 -| 15 SERV  OFF 304  30
(3) 00000000    S02 S00| -  R03|0 0 0 0 -| 15 SERV  OFF 304  30
(4) 00000000    S03 S00| -  R04|0 0 0 0 -| 15 SERV  OFF 304  30


Channel to Node Interface:
    retranslation    |  user+service          lim
 id N  A t        m | N  A t Base     m  sec brc S   e
(0) 0   NO AR       | 1   NO AR          usr OFF NONE
(1) 0   NO AR       | 2 MASK 00000000/08 usr OFF NONE
(2) 1   NO AR       | 1 MASK 00000000/08 usr OFF NONE
(3) 0   NO AR       | 0 MASK 00000000/08 usr OFF NONE


Local retab. No 1
40 to:30
```

The example of configuration CU6:

```
Nodes:
                          retab
Nid|address |M | u   s | L   N |l w n g H|sTO Err  Cent vTO hTO
(0) 0049B897    -  S00| -  R00|0 0 0 0 -| 15 SERV  OFF 304  30
(1) 690F1240    S00 S00|S03 R01|1 0 0 0 -| 15 SERV  OFF 304  30
(2) 00000000    S01 S00| -  R02|0 0 0 0 -| 15 SERV  OFF 304  30
(3) 00000000    S02 S00| -  R03|0 0 0 0 -| 15 SERV  OFF 304  30
(4) 00000000    S03 S00| -  R04|0 0 0 0 -| 15 SERV  OFF 304  30


Channel to Node Interface:
    retranslation    |  user                 lim
 id N  A t        m | N  A t Base     m  sec brc S   e
(0) 0   NO AR       | 1 MASK 00000000/08 ON  OFF NONE
(1) 0   NO AR       | 2 MASK 00000000/08 ON  OFF NONE
(2) 1   NO AR       | 1   NO AR          ON  OFF NONE
(3) 0   NO AR       | 0 MASK 00000000/08 ON  OFF NONE


Local retab. No 1
30 to:40
```

The next response after sending of command **!** is valid for the simple case, when the Setr is connected to CU5 and the CU6 is called:

```
690F1240h>!

u S00   690F1230    S02
  S03   690F1240    serd


 serd   690F1240    S03
  S02   690F1230  u S00
690F1240h>
```

9.2. The serial line connected to the network

An example with the serial line and with the addresses containing two different wide items.

The record of packet passing is contained in its left column. The entry DQ/RSS appears when the packet arrived through radio channel or the marking of serial channel, when the packet arrived by serial channel. In the right column is the marking of channel through which the packet left the node.

```
u S00   69802001    R01
30/ 56  69802002    S02
  S03   69802005    R01
25/ 46  6980300A   serd

 serd   6980300A    R01
27/ 42  69802005    S03
  S02   69802002    R01
30/ 56  69802001  u S00
6980300Ah>
```
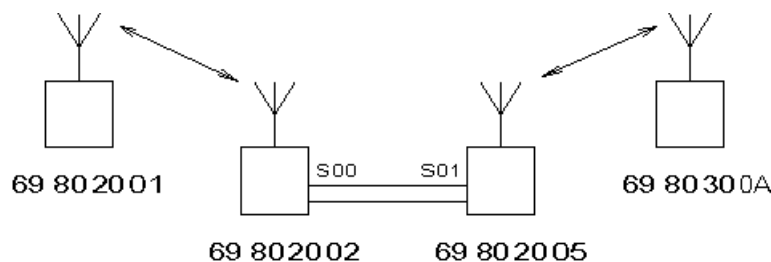

Fig. 9.2: The network layout

The comment to the content of the routing tables:

Tables in the node `69802001`:

- it is not necessary to route packets to node `69802001`

- the routes from node `69802005`, which is behind the serial link (so called "behind wires"), to the nodes `69802005` and `6980300A` must be routed to its own address, even if the addresses are different in the local part only:

        Local retab. No 1
        05 to:02

- the route to node 6980300A will be looked for in the wide table, because the address `6980300A` differs from the own one in the wide part:

        Wide retab. No 1
        30to:2002

Tables in the node `69802002`:

- it is not necessary to route packets to node `69802002` because their addresses are different in local part only and there is a direct radio connection.

- the way to the node `69802005` is routed to `69802002`, the record is:

```
Local retab. No 1
05 to:02
```

- the route to node `6980300A`, whose address is different in the wide part, will be looked for in the wide table:

```
Wide retab. No 1
30to:2002
```

Tables in the node `69802005`:

- the routes to the nodes `69802001` and `69802002` are identical to that of the input to the wire link, so the routess in the local table are routed to own address:

```
Local retab. No 1
01 to:05  02 to:05
```

- the route to the node `6980300A` is looked for in the wide table:

```
Wide retab. No 1
30to:300A
```

Tables in the node `6980300A`:

- all nodes to which we will route the packets are lying in other wide than `30`, so routes in wide table is required. Since all other nodes are lying in a common wide, `20` , only one route suffices:

```
Wide retab. No 1
20to:2005
```

After this practice of routing we give again the description of process how the packet is going through the node completed with more details.

10. Routing 2

Packet through Morse node - enlarged description.

The routing packet entering the CU carries a quadruple MORSE address - source, SRC, destination, DST, node address from where it is coming, FRM, and a node address to where it is directed, TO. The procedure of processing is represented in the diagram *Block circuit of MORSE node* and includes the content of the mentioned addresses.

The access module, represented in the left part of the diagram is assigned by configuration to some of the nodes. The module only accepts packets whose TO address is the same as the address of that node. From there the packet continues with only the SRC and DST addresses.

Packet incoming to the node, contains it's target address DST. The address of immediate next step NN is determined in the node according to address DST.

The address DST is compared with address of the node (NODE ADDRESS), where the process runs. If they are equal then the packet is in the target node and will be sent, according to its type, to the user or service port (including the SRC and DST addresses) or to the service processing. If they are different, then the routing tables processing follows. Into address NN is at first inserted the address DST, which is later rewrited by address found in respective routing table.
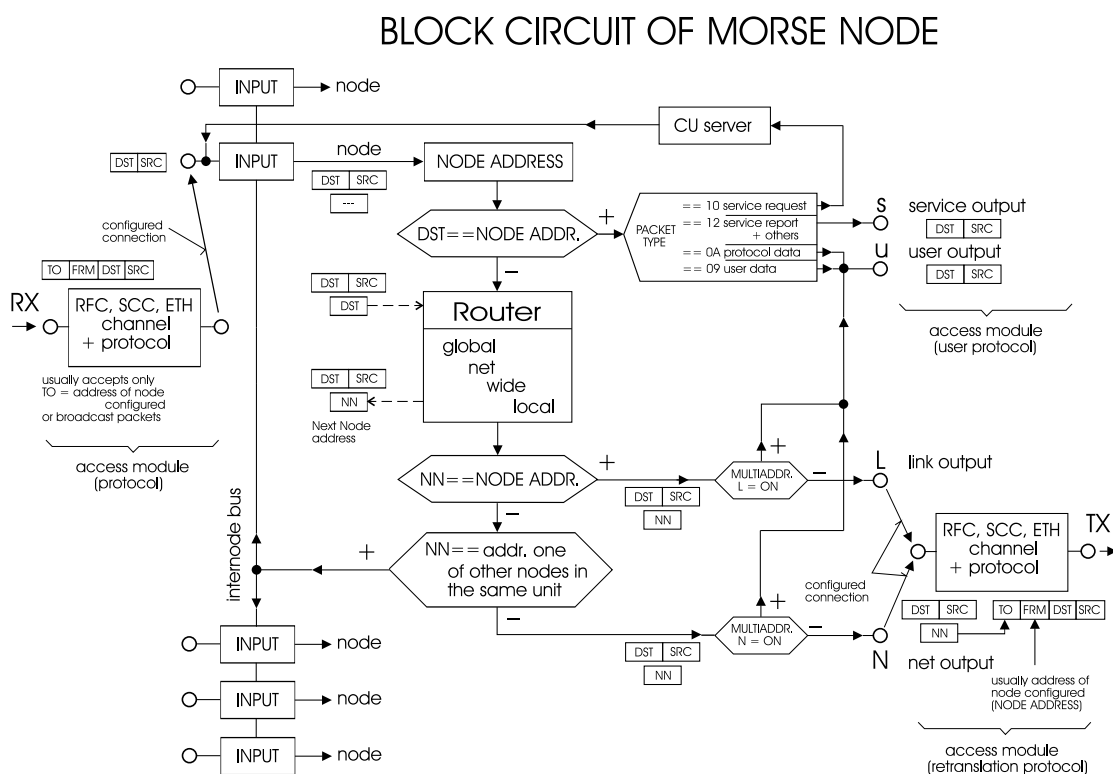

Fig. 10.1: Node block diagram

You can get the diagram in pdf format[1].

---
[1] http://www.racom.eu/images/support/nod2.pdf

The packet address can be divided to the 4 parts called (from left) global, net, wide and local. The `global` part represents a big unit where the address belongs (country), part `net` determines more exactly the region inside the country (geographically or according to other points of view). Next specifying is in part `wide` and part `local` defines finally the actual address of node. At the choice of routing table the addresses DST and NODE ADDR. are compared. The parts global, net, wide and local are compared gradually. If the difference is found (for example in `net` part), then in the respective table (net) will be the address of next step found. The differences in lower parts (here wide and local) are then not important..

The routing table choosed in this way, can be in the CU (Communication Unit) prepared in the versions 1 to 4. This version is used, which number is written in the menu `(N)odes (e)dit` in the row of respective node and in the column according to table type (here `net`). If in this place is not the number 1 to 4 written then on the place of address NN remains the address DST and the packet will be sent directly to this address. This connection will be successful only rarely, because the node having address DST is located usually out of the area of direct audibility.

The routing table contains the pairs `(p)ath` and `(n)ode`. The `(p)ath` represents the compared part of the address (here `net`) and `(n)ode` is address of node NN where leads the next packet step on it's way to DST address. The tables wide and local contains in `(n)ode` the lower address part only. The item `(n)ode` is the resultant information found in the routing tables and is inserted into address NN. More this situations can happen in the routing tables:

- In the local table there are not displayed the equal items, e.g. `(p)ath 5E (n)ode 5E`. This we can employ at removing of the unhelpful items in this way, that we put in `(n)ode` the same item like is in the `(p)ath`.

- It is not necessary to put in the local table the trivial path that is the ways to nodes, which are directly audible and which address is different in part `local` only. (In this case is `to=dest` and `(n)ode=(p)ath`, see last note)

- In the tables global, net and wide are not displayed the items, where is `(n)ode = 0`. In this way we can remove the unhelpful items by putting 0 in the item `(n)ode`.

- The next result is, that the zero items are not used in the net and wide parts. In the global part only there are used the addresses beginning by 00 for the zero node N0 in each CU. These addresses (production numbers) are unique and unchangeable for each CU produced. The are created in different hierarchy then standard addresses and are not suitable for the work with routing tables. However it is possible create the connections via this addresses using the Path packets.

- If the table doesn't contain the item `(p)ath` equal to compared part of address, then the packet is lost. The exception see next article.

- In the tables global and net there it is possible to use the item `(p)ath`, which is equal to respective part of address of node, in which the process runs (NODE ADDR). Then the relevant item `(n)ode` is the default address, where are sent the packets, which haven't own item `(p)ath` in the table.

The packet equipped now by the NN address goes to the next processing according to diagram "Block circuit of MORSE node". If the NN address equals to NODE ADDR, then the packet is sent to the link output L. The actual routing of the link output is written in the menu `(N)ode (e)dit`, in the column L. Some of serial channels or output to the Ethernet is used. The link

output is used above all for the connection between two CU working on different frequencies. The way of packets can go through nets with different frequencies and can be fully described in routing tables. On the other end of wire line there is next node, which accomplish the assignment of next packet step according to address DST, which the packet carries all time.

In next step is the address NN compared with addresses of other nodes in own CU. If the concordance is found, then the packet is sent through inner bus to the respective node. It is not necessary to have other setting in menu `Ne` or `SIe`. In this node there begins the packet processing from start point again include using (different) set of routing tables.

Before sending to the net output the setting of flag `link/net to user addr` is checked. It is contained in the menu `Ne` in column `M` (Multiaddressing). If it is not set, which is the common case, then is the column M empty. By its setting the latter N appears here and then all packets, originally going to the net output, are sent to the user output. The similar opportunity is to change routing of packets going originally to link retranslation output to the user output. Both options are used in special cases of routing only.

If the packet is no diverted to the link output nor to inner bus nor by multiaddressing, then it is sent to the net retranslation output N. It could be the radio or ethernet output assigned in menu `Ne` in column `N`.

Behind the network or link output of the node the output part of the access module continues in the connected communication channel. Here the packet (amongst other things) is again given a quadruple address. SRC and DST addresses remain unchanged, the value NN is inserted into the TO address, and the address of the node, in which the routing process took place, is (generally) written into the FRM address.

In this way the packet is sent to the net medium and for it's next processing takes charge this node, which address is equal to TO address contained in the packet.

This description is the more detailed version of the article Routing 1. The example showing the routing possibilities can be found in the article Routing examples 3.

11. Routing examples 3

More complicated network and the routing tables.

The network is made up of ten nodes which differ in addresses in terms of wide and net. The node `69805000` and `69804000` are connected by a serial link. The node `69A01000` is connected to other networks through node `69A02000` where routing continues.

It is good to remember the basic principle - each node seeks for the next hop according to the destination address, which is always contained in the packet. The node which receives the packet processes the packet in similar fashion. This process continues until the packet arrives at the destination.

```
( 69 A0 20 00 )
   :
  _:_____        _____        _____        _____
 |          |      |          |      |          |      |          |
 | 69 A0 10 00 |......| 69 80 50 00 |-S00=S00-| 69 80 40 00 |........| 69 80 40 10 |
 |_____|      |_____|      |_____|......  |_____|
    T1 -              : T1 -              : T1 -              :
    Tw -              : Tw 30 to 5000     : Tw 30 to 3000     :
      Tn 80 to 69805000  :    40 to 5000     :    50 to 4000     :..|          |
        B0 to 69805000   : Tn B0 to 69B01000 : Tn 80 to 69804000 | 69 80 40 20 |
        C0 to 69805000   :    C0 to 69B01000 : Tg 69 to 69804000 |_____|
        A0 to 69A02000   :    80 to 69A01000  _:_____          T1 10 to 00
      Tg 69 to 69A02000  : Tg 69 to 69A01000 |          |            20 to 00
                         :                   | 69 80 30 00 |        Tw 30 to 4000
  _____            :                   |_____|            50 to 4000
 |          |           :                    : T1 02 to 01          Tn 80 to 69804000
 | 69 B0 10 00 |........:                    : Tw 40 to 4000        Tg 69 to 69804000
 |_____|                                :    50 to 4000
   : T1 -                                    : Tn 80 to 69804000
   : Tw -                                    : Tg 69 to 69804000
   : Tn C0 to 69C01000                        _:_____
   :    B0 to 69805000                       |          |
   : Tg 69 to 69805000                       | 69 80 30 01 |
  _:_____                                 |_____|
 |          |                                 : T1 -
 | 69 C0 10 00 |                              : Tw 40 to 3000
 |_____|                                 :    50 to 3000
    T1 -                                      : Tn 80 to 69803000
    Tw -                                      : Tg 69 to 69803000
      Tn C0 to 69B01000                        _:_____
      Tg 69 to 69B01000                       |          |
                                              | 69 80 30 02 |
                                              |_____|
                                                T1 00 to 01
                                                Tw 40 to 3001
                                                   50 to 3001
                                                Tn A0 to 69803001
                                                   B0 to 69803001
                                                   C0 to 69803001
                                                   80 to 69803001
                                                Tg 69 to 69803001
```

Comments on the contents of the routing tables:

- The nodes `69804010` and `69804020` are contained in common CU. The necessary tables are almost identical - the only difference being in the local tables where node `69804010` needs the item `20` to `00` and node `69804020` needs the item `10` to `00`. Both nodes use the common set of tables in `(N)ode (e)dit`.

- The node `69803002` contains all net address, which appear in the network in it's net tables. All these net addresses are pointed to node `69803001`, so it is possible to use the default

item `80` to `69803001`, as was explained in the article Routing 2. The fact is used in the net and global tables of all other nodes.

- The node `69A01000` has it's default item in net table as `A0` to `69A02000`. i.e it's default route in net table is pointed to `69A02000`, which connects our network to other networks. This is the reason the table contain explicit written net addresses which appear in our network even though they are all directed to a common node.

- At the wide table level, the default route option is not possible. This is why all wide table items which occur in the common net are written here. Similar reason is valid at the local table level as well, but this occur only rarely in our example.