



MiniNET protocol for MORSE

verze 9.0.17.0
8/7/2007

1. Introduction

The MiniNet protocol is used in Bernecker & Rainer (Austria) equipment. Communication in networks with this equipment is always of the Master - Slave variety.



Important

The terminology is unified from Setr version 9.0.17.0 according next schedule:

PLC Master - CU RADIOSLAVE ... CU RADIOMASTER - Slave PLC

CU (radiomodem) connected via SCC to PLC Master is called Radioslave (RS),
CU connected to PLC Slave is called Radiomaster (RM).

Changes appear at a new Setr version, the firmware version has not influence.

The older Setr uses the reverse terminology, also:

(PLC Master - CU RADIOMASTER ... CU RADIOSLAVE - Slave PLC)

2. Data Format

Most of packets in the MiniNet protocol are characterised with this structure:

| STX/8 | LEN/8 | NODE/8 | INDEX/8 | DATA/8*N | CHK/8 |

where:

STX beginning of packet 02h. If 02h occurs in the data, 00h has to come after it. i.e. 02h is transmitted as 0200h

LEN length (from STX up to and including CHK), supplemented 00h in the data (after 02h) is not counted. LEN can acquire values from 05h to FFh.

NODE address, see also the parameter (s) lave adr

INDEX identification of packet for understanding requests and replies at the master (numbering, repeating, errors etc.)

DATA sequence of data bytes

CHK check byte (for calculation see below)

An exception to the above mentioned structure is the single byte packet ACK (06h), which is only sent by the Slave in reply to certain master requests.

The protokol MININET-MORSE selects the above data structure from any incoming frame. In practice most PLCs send a differing number of synchronisation bytes FFh before STX and after CHK.

Protokol MININET-MORSE vybírá výše uvedenou strukturu dat z jakéhokoliv došlého rámce. V praxi většina PLC vysílá před STX a za CHK různý počet synchronizačních bytů FFh.

3. Implementation in Morse

Addressing:

CU (MR400) RS always sends data to the address which it creates by overwriting the lowest byte of its own address (on the relevant node) with the NODE byte, see Data Format. In the case that NODE is 10h (broadcast) the packet is discarded. The CU RM always sends data to the address that the request came from.

a) PLC -> MR400

1. In case of RM device check whether data is expected. The RM can only send data as a reply to a request - unexpected data is discarded.
2. Check whether the received packet has min. size of 5 bytes (exception only at RM device for ACK). Shorter packets are discarded.
3. STX is searched for in the received packet STX, i.e. the packet does not need to start with STX. If STX is not in the packet (not followed by 00h), the packet is discarded.
4. Check length of LEN. If it is larger than the size of the received packet the packet is discarded.
5. According to the given length of LEN, CHK is searched for, i.e. packet does not need to end with CHK.
6. Check CHK.

Only INDEX and DATA are transmitted through the air (see Data Format). STX,LEN,NODE,CHK are not transmitted. Packets are transmitted though the air as USER DATA types. In case of ACK the USER DATA packet is transmitted with a data length of zero.

b) MR400 -> PLC

1. STX,LEN,NODE,CHK are supplemented to the relevant position (calculated) of the received data "from the air".
2. Only the RS can receive a packet with zero length. In this case a single byte packet 06h is created.
3. Packet sent to port.

From the above it is obvious that only service data and index secured with MORSE system algorithms are transmitted through the air. Data on the wire link is protected by MiniNet protocol algorithms.

There is no handshake used on the CU-PLC link - not even SW (via ACK or other characters in the link layer of the protocol) not even HW (only three connections are used - RXD, TXD, GND). In case of overflow of queues of packets for sending into the air (4 packets) there is no way to inform the PLC, and therefore other possible incoming packets are discarded.

3.1. Communication example

```
PLC Master - RADIOSLAVE ..... RADIOMASTER - PLC Slave
  MiniNet      690F0011           690F0022      MiniNet
```

```
PLC Master ----> RS 11 (RADIOSLAVE 690F0011)
FFFFFF 02 07 22 40 1B52 4B FFFF request

RS 11 ----> RM 22
40 1B52                      source address 11 is aved in the CU RM 22

RM 22 ----> PLC Slave
02 07 22 40 1B52 4B

PLC Slave ----> RM 22
FF 02 06 22 C080 DA FFFFFF      response

RM 22 ----> RS 11
C0 80                      RM 22 sends the packet to the address 11

RS 11 ----> PLC Master
02 06 22 C0 80 DA
```

4. Configuration Parameters

MININET parameters:

```
PLC Master - CU RADIOSLAVE ... CU RADIOMASTER - Slave PLC
(m)ode :RADIOSLAVE (wired to master) (s)lave adr :FFFF
(q)uit
```

- (m) • RADIOSLAVE - connected via the service cable to PLC Master - sends data on the RF channel immediately after receiving on the SC channel. Cannot receive ACK (from side of SCC).
- RADIOMASTER - connected via the service cable to PLC Slave - data received on the SCC sent into space only in the case that the CU RS is expecting it (has to be preceded by a request from the PLC Master), i.e. spontaneous communication of the Slave unit is not possible. Incoming data from the RFC is sent immediately on the SCC, i.e. does not wait for a reply to the previous request from the PLC. Can receive ACK (from side of SCC) and sends it over the RF channel to the address from which the request came from.

ATTENTION - The Setr older than 9.0.17.0 uses the reverse labelling !

(s) For CU RADIOSLAVE only

- FFFF - the byte NODE in the response from RS to PLC Master contains the lowest byte of the MORSE address of CU RM which sent the message
- other content - the low byte of parameter (s)lave adr is inserted in the byte NODE

5. Example of CHK calculation

```
Byte Minet::calc_chk(Byte *pb, Word size)
{
    Word     chk,
            i;

    chk = 0;
    i = 0;
    while (i < size) {
        if ((*pb - 1) == (Byte)STX) && (*pb == 0) && (i>4)) {

//4=STX|LEN|NODE|INDEX
            i++; //supplementary 00 after STX in data area is not included
            pb++;
            }
        else {
            chk <= 1;
            if ((chk & 0x100) == 0x100) {
                chk += 1;
                chk &= 0xFF;
                }
            chk += *pb++;
            if ((chk & 0x100) == 0x100) {
                chk += 1;
                chk &= 0xFF;
                }
            i++;
            }
        }
    if ((Byte)chk == (Byte)STX) chk = ~chk;
    return((Byte)chk);
}//Minet::calc_chk
```

6. History

9.0.17.0 - 06/2007 - unification of terms RS/RM in the Setr