RKT protocol.

Used to transfer variable values in RKT controllers between systems.

Systems.

A system is a program (usually FBD) in a controller that services a technological system, for example S1 (supply system), EX1 (exhaust system). There may be several systems in one controller.

The system contains a client and a server. The client of one system connects via the SCTP protocol to the server of another system.

SCTP.

SCTP is a reliable message transfer protocol. It ensures that message boundaries are preserved.

Client and server socket parameters:

```
socket(AF_INET, SOCK_SEQPACKET, IPPROTO_SCTP)
```

SCTP IP addresses and port numbers.

The IP address of the controller and the system number are located in the net.cfg file, which is created in the "Network Configuration Setup" editor.

SCTP port number of the system server = 50000 + system number.

Message format.

Text messages in UTF-8 format.

Messaging.

For the client, there are variables that he wants to write to the server - outgoing, and receive from the server - incoming.

1. The client subscribes to the server immediately after connecting. The following messages are sent:

```
variable_name_on_server1 variable_name_on_client1
variable_name_on_server2 variable_name_on_client2
…
```

For example:

```
Toutdoor_air Tout
T_water T_water
```

2. Subscription end messages are sent (2 messages):

```
#
#
```

3. The client then sends a list of output variable values:

```
server_variable_name1 original_variable_value1
variable_name_on_server2 original_variable_value2
…
```

For example:

```
Damper 45.5
```

4. The server sends a list of values of all subscribed variables

```
variable_name_client1 variable_value_server1
variable_name_client2 variable_value_server2
…
```

The initial connection process is complete.

5. Now if an outgoing variable changes on the client, the client sends it to the server:

```
server_variable_name client_variable_value
```

6. If the value of a subscribed variable has changed on the server, the server sends it to the client:

```
variable_name_client variable_value_server
```

Error handling.

If the server detects an error in a message from the client, it sends an error message to the client:

```
& Error Description
```

The client should treat a message whose first character is & as an error message.