2N® SmartCom PRO

Wireless Device Management over IP

User Manual

The 2N TELEKOMUNIKACE a.s. is a Czech manufacturer and supplier of telecommunications equipment.

The product family developed by 2N TELEKOMUNIKACE a.s. includes GSM gateways, private branch exchanges (PBX), and door and lift communicators. 2N TELEKOMUNIKACE a.s. has been ranked among the Czech top companies for years and represented a symbol of stability and prosperity on the telecommunications market for almost two decades. At present, we export our products into over 120 countries worldwide and have exclusive distributors on all continents.

$2N^®$ is a registered trademark of 2N TELEKOMUNIKACE a.s. Any product and/or other names mentioned herein are registered trademarks and/or trademarks or brands protected by law.

2N TELEKOMUNIKACE a.s. administers the FAQ database to help you quickly find information and to answer your questions about 2N products and services. On www.faq.2n.cz you can find information regarding products adjustment and instructions for optimum use and procedures „What to do if…".

2N TELEKOMUNIKACE a.s. hereby declares that the $2N^®$ product complies with all basic requirements and other relevant provisions of the 1999/5/EC directive. For the full wording of the Declaration of Conformity see the CD-ROM (if enclosed) or our website at www.2n.cz.

The 2N TELEKOMUNIKACE a.s. is the holder of the ISO 9001:2009 certificate. All development, production and distribution processes of the company are managed by this standard and guarantee a high quality, technical level and professional aspect of all our products.

Content:

1. Product Overview

In this section, we introduce the **2N**[®] **SmartCom PRO** product, outline its application options and highlight the advantages following from its use. The section also includes safety precautions.

Here is what you can find in this section:

- 1.1 Product Description
- 1.2 Changes
- 1.3 Terms and Symbols Used

1.1 Product Description

2N® SmartCom PRO and 2N® SmartCom PRO ETH GPRS/UMTS/ETH communication unit is a new product developed and manufactured to provide the maximum utility value, quality and reliability. We hope you will be fully satisfied with the 2N® SmartCom PRO for a long time. Use 2N® SmartCom PRo in accordance herewith for the purposes it was designed and manufactured for.

2N® SmartCom PRO is designed for Internet connection via the GPRS/UMTS/ETH network and data retransmission from peripherals connected via RS 232/485/M-Bus /MDB/WM-Bus/ZigBee via the IP protocol and GPRS/UMTS or Ethernet network to a data requesting server or device. Two relay outputs can be controlled and switched and parameters can be measured on two galvanically connected inputs.

Safety Precautions

Do not switch on 2N® SmartCom PRO in the vicinity of medical apparatuses to avoid interference. The minimum distance of the antenna and pacemakers should be 0.5m at least.

Do not switch 2N® SmartCom PRO aboard a plane.

Do not switch 2N® SmartCom PRO near petrol stations, chemical facilities or sites where explosives are used.

Any mobile telephone use prohibition based on RF energy radiation applies to 2N® SmartCom PRO too.

2N® SmartCom PRO may disturb the function of TV sets, radio sets and PCs.

Warning! 2N® SmartCom PRO contains components that can be swallowed by small children (SIM card, antenna, etc.).

Never exceed the voltage value specified on the adapter. Check the available voltage range before connecting **2N® SmartCom PRO** to a different power supply.

When **2N® SmartCom PRO** comes to the end of its operational life, dispose of it in accordance with applicable regulations.

2N® SmartCom PRO is equipped with an external antenna connector. The antenna has to be located indoors for safety reasons.

⚠ **Warning**

- Do not place the unit near heat sources (such as space heaters, hot air heaters, etc.).

- **2N® SmartCom PRO** only works reliably under the conditions specified in this User Manual. Any unauthorised interventions and/or changes in use and operation may result in malfunction or destruction of the product.

1.2 Changes

The manufacturer reserves the right to modify 2N® SmartCom PRO in order to improve its qualities.

Manual Version	Changes in Documentation
1.11.3	The User Manual corresponds to 2N® SmartCom PRO FW version **1.11.3** or user-optimized firmware version **1.12.x**.

> ⚠️ **Caution**
>
> - The manufacturer is committed to upgrading the firmware according to the clients' requirements. Refer to the 2N web sites **www.2n.cz** for the current 2N® **SmartCom PRO** firmware versions and User Manual updates.
>
> - Refer to the Maintenance section for 2N® **SmartCom PRO** firmware upgrade details.

1.3 Terms and Symbols Used

Symbols Used in Manual

The following symbols and pictograms are used in the manual:

> **⊘ Safety**
>
> - **Always abide** by this information to prevent persons from injury.

> **⊘ Warning**
>
> - **Always abide** by this information to prevent damage to the device.

> **⚠ Caution**
>
> - **Important information** for system functionality.

> **⊘ Tip**
>
> - **Useful information** for quick and efficient functionality.

> **ⓘ Note**
>
> - Routines or advice for efficient use of the device.

AT Command AT command stored in the hyper terminal.

Future Functions and Features

The grey-marked text in this document designates the functions and features that are under preparation or development at present.

2. Description and Installation

In this section we describe the **2N**® **SmartCom PRO** product and its installation.

Here is what you can find in this section:

- 2.1 Description
- 2.2 Before You Start
- 2.3 Mounting

2.1 Description

2N® SmartCom PRO consists of a PCB carrying a power supply and RS 232, RS 485 /M–Bus/MDB interfaces. An optional GSM/UMTS module provides continuous Internet connection via GPRS/UMTS. Two relays are available for output contact control.

Connectors for optional devices are located in the upper part of **2N® SmartCom PRO**. Refer to the figures below for description. There is a gel–lead–acid accumulator connector on the bottom panel. The board also includes a real time clock backup and optional Wireless M-Bus and ZigBee modules. An optional Ethernet connector is mounted on the bottom too. Refer to the figures below for details. The whole **2N® SmartCom PRO** system is enclosed in a solid aluminium case. Mount the case on a DIN rail for easier installation.

2N® SmartCom PRO Description (Bottom View)

1. **RS 232 connector**
   - 9-pin D-sub
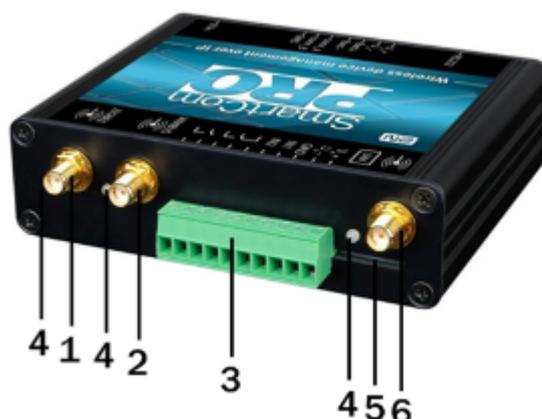2. **Power supply and battery terminal board**
   - $+ U_{in}$, $- U_{in}$ – input supply voltage terminals
   - $+$ Batt, $-$ Batt – gel–lead–acid accumulator terminals
3. **RS 485/M-Bus/RS232 connector (bus type according to the selected version)**
   - +,-,GND for RS 485
   - A,B,GND for M-Bus
4. **Ethernet interface (optional)**

- RJ-45

2N® SmartCom PRO Description (Top View)

1. **ZigBee antenna SMA connector (optional)**

2. **Wireless M-Bus antenna SMA connector (optional)**

3. **Input/output circuit terminal board (from left):**
   - RE1 – closing relay contacts
   - RE2 – switching relay contacts
   - U_{in}+, U_{in}- – auxiliary supply voltage contacts, used e.g. for connecting supply voltage to the relays (galvanically connected to the U_{in}+, U_{in}– contacts on the 4–pin power supply connector on the bottom side)
   - GND – ground used with the IN1 and IN2 input circuits
   - IN1, IN2 – input circuits (input type according to the jumper settings, refer to **Subs. 3.3**)

4. **Signalling LEDs**
   - from left: ZigBee, WM-Bus, GSM/UMTS or ETH as configured

5. **SIM holder**

6. **GSM/UMTS antenna SMA connector**

Installation Conditions

- Install the 2N$^®$ SmartCom PRO antenna with respect to the signal quality in the location. Where the signal is poor, use a higher gain antenna (not included in the delivery).

- Install 2N$^®$ SmartCom PRO off sensitive devices and human bodies to avoid EM interference.

- Refer to the Technical Parameters for the allowed range of working temperatures.

- 2N$^®$ SmartCom PRO may not be operated on sites exposed to direct sunshine or in the vicinity of heat sources.

- 2N$^®$ SmartCom PRO is designed for indoor use. Never expose it to rain, running water, condensation moisture, mist, etc.

- 2N$^®$ SmartCom PRO may not be exposed to aggressive gases, acid vapours, solvents and similar chemicals.

> ⊘ **Tip**
>
> - Order an IP65 Installation Box with a DIN rail, grounding terminals and bushings if you need to install 2N$^®$ SmartCom PRO in such adverse conditions.

> ⚠ **Caution**
>
> - Make sure that you are equipped with all necessary technical means, particularly a GPRS/UMTS supporting SIM card with PIN request disable or a PIN identical with that set in the 2N$^®$ SmartCom PRO configuration.

2.3 Mounting

External Antenna Connection

Screw the enclosed antenna into the SMA antenna connector.

Antenna Connection
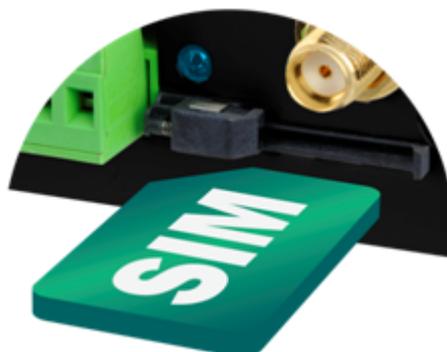
> ⚠️ **Caution**
>
> - Tighten the antenna connector **gently with your hand**; never use a tool!

> ℹ️ **Note**
>
> - The antenna should be located in one and the same building with **2N**® **SmartCom PRO** for safety reasons.

SIM Card Installation

Find the SIM card holder near the **2N**® **SmartCOM PRO** top.

SIM Card Installation

1. Insert the SIM card in the holder.
2. Push the card in until it clicks into position.

> ⚠️ **Caution**
>
> - Make sure that the GSM provider's SIM card is intended for the GSM network supported by your **2N**® **SmartCom PRO** version and that GPRS-based data transmission is activated.
> - Make sure that the SIM card does not request the PIN or that the PIN is identical with that set in the **2N**® **SmartCom PRO** configuration.
> - Remember to set the relevant SIM and provider's services (call forwarding, call barring, preferred networks, SMS centre, etc.) via your mobile phone before inserting the SIM card in **2N**® **SmartCom PRO**.

Power Supply

2N$^®$ SmartCom PRO is 5–50 V DC supplied. Make sure that the voltage value and polarity comply with the data specified at the 2N$^®$ SmartCom PRO power connector before connecting your 2N$^®$ SmartCom PRO to an adapter other than that included in the delivery. Connect the supply voltage to the 4-pin connector next to the RS 232 connector. You can also connect a gel–lead–acid accumulator as a power supply backup. The supply voltage is also present on the 10-pin input circuit terminal block on the opposite side of the device. These contacts can be used as auxiliar contacts, e. g., for connecting supply voltage to the relays.

> ⊙ **Warning**
>
> - Do not connect power supply before connecting an antenna to 2N$^®$ SmartCom PRO to avoid the GSM module damage.
> - Do not connect any incompliant power supply to avoid electric accident or system damage.

> ⊙ **Safety**
>
> - Do not connect the batteries before making sure that the charger has a correct voltage value or is off to avoid device damage and battery destruction!

3. Function and Use

In this section we provide the basic and extended functions of the 2N® SmartCom PRO product.

Here is what you can find in this section:

- 3.1 Putting in Operation
- 3.2 LED Indicator
- 3.3 Input and Output Circuits
- 3.4 Input Calibration
- 3.5 User Defined Functions - UDF
- 3.6 Wireless M-Bus
- 3.7 ZigBee interface
- 3.8 Autonomous Metering System (AMS)
- 3.9 Other Useful Functions

3.1 Putting in Operation

Operational state is 2N® SmartCom PRO's main function. In this state, periodic data readouts from input circuits are made. 2N® SmartCom PRO responds to queries from the superior data concentrator, which can be the 2N® SmartCom server, monitors events on the RS 232 / RS 485 / M-Bus interfaces and sends the acquired data to the server via GPRS or Ethernet in the case of the PRO ETH version.

Initialisation

Insert the PIN–disabled data SIM card in 2N® SmartCom PRO and connect the GSM antenna and power supply as instructed in Subs. 2.3., Mounting.

When the 2N® SmartCom PRO power supply is connected, the LED indicator on the upper side of the module flashes three times quickly and then starts flashing in longer intervals. Refer to Subs. 3.2. for LED status details. The GPRS module starts communicating with the provider in about 10 seconds trying to log in and obtain the IP address. The LED starts shining constantly to indicate a successful login.

> ⊘ **Tip**
>
> - 2N® SmartCom PRO is working if the LED indicator is shining, thus indicating a successful mobile provider connection and IP address assignment.

To identify the 2N® SmartCom PRO IP address assigned by the mobile provider or perform the basic configuration upon power up, get connected via the serial port (RS 232). The port is in the data mode upon power up, retransmitting all data from the input to the GPRS network. Use the +++ sequence preceded and followed by a 1–second delay to switch into the configuration mode. When your 2N® SmartCom PRO replies OK, start configuring using the AT commands as listed below.

Restart the serial port to return to the data mode. To do so, you can either send the ATO command or disconnect 2N® SmartCom PRO from the power supply.

> ⊘ **Tip**
>
> - **2N® SmartCom PRO** can also be configured via Terminal Config. Refer to **Sub. 4.1** .
> - If you cannot use a PC, make the basic settings using SMS messages. Refer to **Sub 4.2** for details.

When your **2N® SmartCom PRO** has been programmed as required, connect the device to the available serial ports RS 232 / RS 485 / M-Bus. Launch the hyperterminal and get connected to the **2N® SmartCom PRO** public IP address via port 10000, which, together with the serial port, can be used for configuring. Use port 10001 for retransmitting data to RS 232 and port 10002 for retransmitting data to RS 485 / M-Bus.

Basic Configuration by SCCFG Command

Enter the SCCFG command via the AT interface to make basic configuration of the GSM/UMTS module, access to the server and listening ports of 2N® SmartCom PRO. If you do not have access to a PC, you can control the terminal via SMS to a certain extent. Refer to Subs. 4.2.

```
at^sccfg?
```

The command displays the currently set interface values.

- ^SCCFG: "MODE","STANDALONE"
- ^SCCFG: "GPRS_APN","publicip.t-mobile.cz"
- ^SCCFG: "GPRS_USER",""
- ^SCCFG: "GPRS_PASS",""
- ^SCCFG: "LOCAL_IP","0.0.0.0"
- ^SCCFG: "SERVER_IP","90.182.112.70"
- ^SCCFG: "SERVER_PORT",1730
- ^SCCFG: "AUTH_PASS","12345"
- ^SCCFG: "SIM_PIN",""
- ^SCCFG: "AUTH_IP",0
- ^SCCFG: "ENCRYPT",0
- ^SCCFG: "ENCRYPT_KEY",3131323335363738393435363135363 4
- ^SCCFG: "LISTEN_PORTCMD",10000
- ^SCCFG: "LISTEN_PORT1",10001
- ^SCCFG: "LISTEN_PORT2",10002

```
at^sccfg=?
```

The command displays the interface setting options.

- *^SCCFG: "MODE","TCPCLIENT"|"STANDALONE"|"DATA_OFF"*
- *^SCCFG: "GPRS_APN","0-32"*
- *^SCCFG: "GPRS_USER","0-32"*
- *^SCCFG: "GPRS_PASS","0-32"*
- *^SCCFG: "LOCAL_IP"*
- *^SCCFG: "SERVER_IP","7-15"*
- *^SCCFG: "SERVER_PORT",0-65535*
- *^SCCFG: "AUTH_PASS","5-64"*
- *^SCCFG: "SIM_PIN","0-8"*
- *^SCCFG: "AUTH_IP",0-1*
- *^SCCFG: "ENCRYPT",0-1*
- *^SCCFG: "ENCRYPT_KEY","16"|<32 HEX>*
- *^SCCFG: "LISTEN_PORTCMD",1-65535*
- *^SCCFG: "LISTEN_PORT1",1-65535*
- *^SCCFG: "LISTEN_PORT2",1-65535*
- *^SCCFG: "SAVE"*
- *^SCCFG: "RESTART"*
- *^SCCFG: "SRESTART"*

```
at^sccfg=&quot;mode&quot;,1
```

- Select one of the following three 2N® SmartCom PRO working modes:

  - TCPCLIENT (1) – 2N® SmartCom PRO connects to the selected server for interface data resending.

  - STANDALONE (3) – 2N® SmartCom PRO works as a server to which clients get connected (transparent transmission between TCP and serial interfaces).

  - DATA_OFF - the 2N® SmartCom PRO GSM/UMTS module is not logged in to the data network but waits for an SMS.

> ⊘ **Tip**
>
> - Set the mode number or enter the mode name to select a mode. The AT command is as follows:
>
> ```
> at^sccfg="mode","tcpclient"
> ```

> ⚠ **Upozornění**
>
> - The ETH port is visible in the network in the DATA_OFF mode and PING can be used. But no outgoing connections are set up and all incoming connections are rejected.

```
at^sccfg=&quot;gprs_apn&quot;,&quot;internet.open&quot;
```

- This command sets the access APN to the mobile provider. O2 CZ configuration is used as an example.

```
at^sccfg=&quot;gprs_user&quot;,&quot;internet&quot; at^sccfg=&quot;
gprs_pass&quot;,&quot;hfsdj515d&quot;
```

- These two commands set the user name and password for provider authentication. Most providers use no authentication. If your provider requires authentication, ask for necessary access data.

```
at^sccfg=&quot;local_ip&quot;
```

- This command makes 2N® SmartCom PRO return the IP address assigned to it by the provider.

```
at^sccfg=&quot;server_ip&quot;,&quot;90.182.112.54&quot;
```

- The command sets the IPv4 address of the server to which 2N® SmartCom PRO connects in mode 1.

```
at^sccfg=&quot;server_port&quot;,1564
```

- The command sets the server port to which 2N® SmartCom PRO connects in mode 1.

```
at^sccfg=&quot;auth_pass&quot;,&quot;xJ32ppp_v1&quot;
```

- The command sets the authorisation password for the above mentioned server.

```
at^sccfg=&quot;sim_pin&quot;,&quot;1156&quot;
```

- The command sets the PIN to be entered into the SIM card upon power up.

```
at^sccfg=&quot;auth_ip&quot;,1
```

- The command enables/disables IP address access authorisation. Refer to S. 3 for details.

```
at^sccfg=&quot;encrypt&quot;,1
```

- Select one of the following **2N**® **SmartCom PRO** server communication encryption modes:

  - Mode 0 DISABLED – **2N**® **SmartCom PRO** shall not encrypt any data itself.

  - Mode 1 OPTIONAL – **2N**® **SmartCom PRO** shall encrypt data if the counterparty requires so.

  - Mode 2 FORCED – **2N**® **SmartCom PRO** shall encrypt all outgoing data.

```
at^sccfg=&quot;encrypt_key&quot;,&quot;
E2978FE2978FE2978FE2978FE2978F20&quot;
```

- Use the parameter to specify the encryption key. Set 32 valid (hexa) characters only, i.e. 16 ASCII characters altogether via the **2N**® **SmartCom Control Panel**.

```
at^sccfg=&quot;listenportcmd&quot;,10000
```

- Use the parameter to specify the listening port for IP access to configuration. The default value is 10000.

```
at^sccfg=&quot;listen_port1&quot;,10001
```

- Use the parameter to specify the listening port for IP access to port1. The default value is 10001.

```
at^sccfg=&quot;listen_port2&quot;,10002
```

- Use the parameter to specify the listening port for IP access to port2. The default value is 10002.

```
at^sccfg=&quot;save&quot;
```

- The command saves the changes.

```
at^sccfg=&quot;restart&quot;
```

- The command restarts 2N® SmartCom PRO .

```
at^sccfg=&quot;srestart&quot;
```

- The command saves the changes and restarts 2N® SmartCom PRO.

Ethernet Port Setting (2N ® SmartCom PRO ETH only)

These commands are only implemented in 2N® SmartCom PRO ETH, which includes an Ethernet port.

```
at^sceth=&quot;ip_method&quot;,&quot;DHCP&quot;
```

- The command selects the IP address setting mode:
  - DHCP – automatic IP address setting from the DHCP server.
  - FIXED – manual IP address setting.

```
at^sceth=&quot;local_mac&quot;
```

- The command writes out the MAC address currently used on the Ethernet port.

```
at^sceth=&quot;local_ip&quot;
```

- The command writes out the IPv4 address currently used on the Ethernet port.

```
at^sceth=&quot;fixed_ip&quot;,&quot;192.168.1.1&quot;
```

- The command is used for manual IP address setting.

```
at^sceth=&quot;mask&quot;,&quot;255.255.255.0&quot;
```

- The command is used for manual network mask setting.

```
at^sceth=&quot;gateway&quot;,&quot;192.168.1.254&quot;
```

- The command is used for manual default gateway setting.

```
at^sceth=&quot;dns1&quot;,&quot;0.0.0.0&quot;
```

- The command is used for manual primary DNS setting.

```
at^sceth="dns2","0.0.0.0"
```

- The command is used for manual secondary DNS setting.

```
at^sceth="save"
```

- The command saves the changes.

```
at^sceth="restart"
```

- The command restarts the Ethernet port.

```
at^sceth="srestart"
```

- The command saves the changes and restarts the Ethernet port.

```
at^sceth?
```

- The command displays the currently set interface values.
- ^SCETH: "IP_METHOD","FIXED"
- ^SCETH: "LOCAL_MAC","7C:1E:B3:00:92:E5"
- ^SCETH: "LOCAL_IP","192.168.1.1"
- ^SCETH: "FIXED_IP","192.168.1.1"
- ^SCETH: "MASK","255.255.255.0"
- ^SCETH: "GATEWAY","192.168.1.254"
- ^SCETH: "DNS1","192.168.1.1"
- ^SCETH: "DNS2","0.0.0.0"

```
at^sceth=?
```

- The command displays the interface setting options.

- ^SCETH: "IP_METHOD","FIXED"|"DHCP"

- ^SCETH: "LOCAL_MAC"

- ^SCETH: "LOCAL_IP"

- ^SCETH: "FIXED_IP","7–15"

- ^SCETH: "MASK","7–15"

- ^SCETH: "GATEWAY","7–15"

- ^SCETH: "DNS1","7–15"

- ^SCETH: "DNS2","7–15"

- ^SCETH: "SAVE"

- ^SCETH: "RESTART"

- ^SCETH: "SRESTART"

```
at^scchprim=&quot;GSM&quot;
```

- The command sets the primary channel for connections:
    - GSM – via GPRS/UMTS.
    - ETH – via Ethernet.

```
at^scchprim?
```

- The command writes out the current primary channel setting.

```
at^scchled1=&quot;GSM&quot;
```

- The command sets the LED indicating function:
    - GSM – LED indicates the GSM/UMTS connection state.
    - ETH – LED indicates the Ethernet port state.

```
at^scchled1?
```

- The command writes out the current LED indication state.

Gel-Lead-Acid Accumulator Charger

This function is only available in 2N[®] SmartCom PRO and 2N[®] SmartCom PRO ETH, which includes an intelligent charger for an external gel-lead-acid accumulator. Charging is automatically controlled and terminated when full charging is achieved. The device is supplied from the accumulator without loss in the case of power outage and the accumulator is automatically recharged upon power recovery. 6 or 12V accumulators can only be used. Make sure while charging that the supply voltage is 1 V at least higher than the maximum accumulator voltage. Use the commands below for configuration.

```
at^scchg=&quot;max_voltage&quot;,6800
```

- The command sets the maximum accumulator voltage in mV.

```
at^scchg=&quot;max_current&quot;,350
```

- The command sets the maximum supply current in mA.

```
at^scchg=&quot;state&quot;
```

- The command identifies the current charging state:
    - ACTIVE – charging activated, accumulator fully charged.
    - CHARGING – charging activated, accumulator is being charged.
    - STOPPED – charging stopped.

```
at^scchg=&quot;start&quot;
```

- The command activates the charging function.

```
at^scchg=&quot;stop&quot;
```

- The command deactivates the charging function.

```
at^scchg=&quot;save&quot;
```

- The command saves the changes.

```
at^scchg=&quot;srestart&quot;
```

- The command saves the changes and resets the charging parameters immediately.

```
at^scchg?
```

- The command displays the current charger parameters and charging state.

```
at^scchg=?
```

- The command displays the charger setting options.

⊘ **Warning**

- Remember to set the charger parameters before connecting the accumulator and starting the charging process. Make sure that the settings are in compliance with the catalogue values of the accumulator manufacturer to avoid accumulator damage, fire or even explosion.

- **Never connect an accumulator with a rated voltage higher than the 2N®**

 SmartCom PRO supply voltage. If this happened, 2N® SmartCom PRO would start using a higher voltage supply, i.e. the battery, discharge the accumulator and switch to a lower voltage supply. However, it would be unable to recharge the battery (see the condition below) and you would lose backup during power outage.

⚠ **Caution**

- Use a 6 or 12V gel-lead-acid accumulator only. Make sure while charging that the supply voltage is 1 V at least higher than the maximum accumulator voltage.

Real Time Clock

This function is only available in 2N® SmartCom PRO and 2N® SmartCom PRO ETH. The real. time clock is backed up with a super capacitor, which keeps the clock going for two days upon disconnection or power failure. Configure the clock via the AT interface or synchronisation from the 2N® SmartCom Server as follows.

```
at^scclk=&quot;RR/MM/DD,hh:mm:ss+ZZzz&quot;
```

- The command sets the real time clock. The ZZzz parameter defines the time zone: shift of ZZ hours and zz minutes against GMT.

```
at^scclk?
```

- The command displays the current time as follows:

^SCCLK: "12/04/19,16:00:25+0200"

(April 19, 2012, 16:00:25, 2 hour shift from GMT)

3.2 LED Indicator

Here the variable statuses of the wireless interface state indicating LEDs are described. The first LED from the right (when viewed from the front, see the figure below) indicates the GSM/UMTS module state. This LED can have two functions: the basic function is to signal terminal connection to the GPRS/UMTS network in the standalone mode and **2N**® **SmartCom Server** in the TCP client mode. For **2N**® **SmartCom PRO ETH**, however, use the scchled1 parameter to select whether the LED shall indicate the GSM or ETH interface state. The states are then identical.

Front Panel with LED Indicators

The LED states below correspond to both the modes.

1. **NO LIGHT** – POWER OFF – indicates that the 12V GPRS module power supply is disconnected or has a malfunction or that the fuse is defective.

2. **THREE QUICK FLASHES** – START or RESTART – indicates the device start /restart. This state must always be indicated upon power supply connection.

3. **SLOW FLASHING** – INITIALISATION – indicates that the module is being connected to the provider, IP network and **2N® SmartCom Server**. If this happens when the ETH activity is being monitored, the network is unavailable. The link is missing, the cable is disconnected or interrupted between the device and the nearest switch.

4. **VERY QUICK FLASHING** – ERROR - indicates an error state: wrong PIN, no SIM card, the count of remaining PIN entering attempts is lower than three.

5. **ILLUMINATED** – OK – **2N® SmartCom PRO** is connected to the provider, IP address is assigned, server connection is established. A quick flash indicates data transmission in this state.

The other two LEDs from the left indicate the extending module states. Their function is always the same regardless of the module type: WM-Bus of two different frequencies, ZigBee and, prospectively, KNX.

The LED states are as follows:

1. **NO LIGHT** — POWER OFF - no module is mounted.

2. **SLOW FLASHING** — INITIALISATION - the module starts and logs in (ZigBee). This state is visible upon power up only.

3. **ILLUMINATED** — OK - the module is ready, waiting for incoming data or command sending.

4. **QUICK FLASHING** — DATA - data transmission in progress. A short message makes the LED flash very shortly.

3.3 Input and Output Circuits

This subsection describes the input and output circuits of **2N**® **SmartCom** including their main use, control and location.

> ⊘ **Tip**
>
> - **2N**® **SmartCom PRO** is equipped with reliable terminals for easier and faster connection.
> - Refer to **S. 6** for a complete AT command list.

Input Circuits

2N® **SmartCom PRO** is equipped with two inputs, which are mutually galvanically connected with the GND reference point (have a common GROUND). Remember to keep the parameters specified in the Technical Parameters at the end of this manual.

The input circuits can work in three modes – voltage, current or logical levels – as configured.

The figures below show the location of the current/voltage/S0 switching jumpers. Loosen the screws and open the box lid to get access to the PCB jumpers. See the figure to the right for the mode setting options.
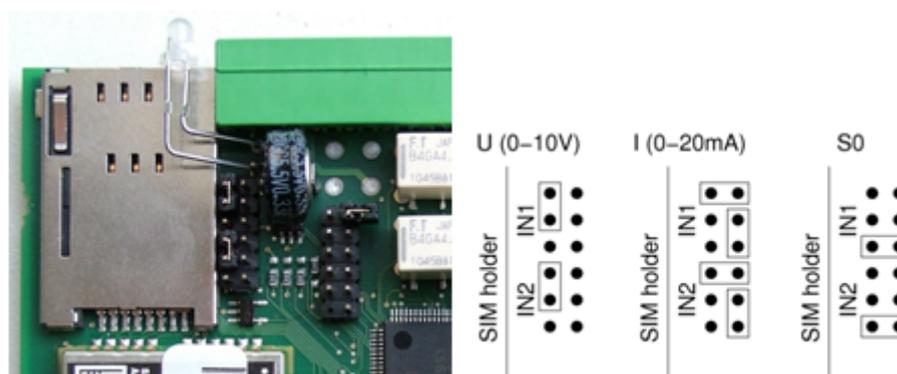
Jumper Settings

Voltage Measurement

Voltage can be measured in the range between 0 and 10 V DC. Make sure that the 2N®
SmartCom PRO jumpers are set as shown in the figure above before using this mode.
Use the AT commands listed below for reading and calibration.

Current Measurement

Electric current can be measured in the range between 4 and 20 mA. Make sure that
the 2N® SmartCom PRO jumpers are set as shown in the figure above before using
this mode. Use the AT commands listed below for reading and calibration. These
commands control the input circuits, set calibration and perform measurement. An
example is given for the ADC circuit 1 (scadc1). To read circuit 2, replace 1 with 2 in the
command (scadc2).

```
at^scadc1=&quot;get_value&quot;
```

- The command performs measurement on the selected interface and sends the
  calibrated value measured.

```
at^scadc1=&quot;adc_value&quot;
```

- The command performs measurement and returns the value from the A/D
  converter.

```
at^scadc1=&quot;calib_min&quot;
```

- Automatic interface calibration according to the currently measured value.

```
at^scadc1=&quot;calib_max&quot;
```

- Automatic interface calibration according to the currently measured value.

```
at^scadc1=&quot;threshold_low&quot;,200
```

- Manual calibration setting (minimum values). Setting options: 0 – 960.

```
at^scadc1=&quot;threshold_high&quot;,250
```

- Manual calibration setting (maximum values). Setting options: 0 – 960.

```
at^scadc1=&quot;save&quot;
```

- The command saves the changes.

Logical Level Monitoring

You can monitor the logical levels of voltage and current signals. Use other commands than those intended for classic measurements and define the threshold levels. Again, use the AT commands for reading values on input circuits.

```
at^scdin1=&quot;get_value&quot;
```

- The command detects the logical level on the selected interface and sends the value 1/0.

```
at^scdin1=&quot;calib_min&quot;
```

- Automatic interface calibration according to the currently measured value .

```
at^scdin1=&quot;calib_max&quot;
```

- Automatic interface calibration according to the currently measured value .

```
at^scdin1=&quot;threshold_low&quot;,400
```

- Manual calibration setting (minimum values). Setting options: 0 – 960.

```
at^scdin1=&quot;threshold_high&quot;,450
```

- Manual calibration setting (maximum values). Setting options: 0 – 960.

```
at^scdin1=&quot;save&quot;
```

- The command saves the changes.

P ulse Counting on S0 Inputs

Make sure that the jumpers are set as shown in the figure below for pulse counting on S0 inputs. The S0 input has thus a power supply of its own and can be connected directly to the switch or relay contacts. The counter is disabled by default and has to be enabled using the AT commands listed below. The commands are used for setting and reading of the S0 input pulse counter states.

```
at^scpulse1=&quot;state&quot;
```

- The command detects the current states of the input 1 counters.
  - COUNTING – counting in progress.
  - STOPPED – counting stopped.

```
at^scpulse1=&quot;get_value&quot;
```

-
  The command reads out the counter state.

```
at^scpulse1=&quot;set_value&quot;,100
```

- The command sets the counter pulse count (100 in this case) if you do not want to start from zero. Setting options: 0 – 4294967295.

```
at^scpulse1=&quot;stimer&quot;,60
```

- The command sets the EEPROM storing intervals for the counter in seconds. Setting options: 5 – 10000 s.

```
at^scpulse1=&quot;start&quot;
```

- The command starts counting on input 1.

```
at^scpulse1=&quot;stop&quot;
```

- The command stops counting on input 1.

```
at^scpulse1=&quot;clear&quot;
```

- The command clears the input 1 counter.

```
at^scpulse1=&quot;save&quot;
```

- Manual storing of the counter value into the EEPROM.

Output Circuits

2N® SmartCom PRO is equipped with two relay outputs for turning on/off electrical appliances (by an additional power switch or as signalling control inputs, depending on their power input). Remember to keep the parameters specified in the Technical parameters at the end of this manual.

You can control the relay outputs with the AT commands from the CMD mode of the RS 232 interface, via a remote GPRS connection to the SIM card IP address, or using an SMS message. Refer to **Subs. 4.2** for AT command details and **S. 5** for SMS configuration details .

> ⚠️ **Caution**
>
> - Attention! Any manual setting is valid until the next change or device power off. 2N® SmartCom PRO does not keep in mind the relay settings by default and restores the default status upon power up . S et the parameter determining the relay position upon power up to 2 to make your 2N® SmartCom PRO remember the last contact position before power off. Refer to the configuration subsections for more details.

Relay Outputs

Use the commands below to control the relay outputs. An example is given for relay 1. For relay 2, use the same commands but for screl2.

```
at^screl1?
```

- The commands detects the state of relay 1.

```
at^screl1=0
```

- The command opens relay 1.

```
at^screl1=0,0
```

- The command opens relay 1. The relay will be open upon the terminal restart.

```
at^screl1=0,1
```

- The command opens relay 1. The relay will be closed upon the terminal restart.

```
at^screl1=0,2
```

- The command opens relay 1. Upon terminal restart, the relay will be in the state in which it was before restart.

```
at^screl1=1
```

- The command closes relay 1.

```
at^screl1=1,0
```

- The command closes relay 1 . The relay will be open upon the terminal restart .

```
at^screl1=1,1
```

- The command closes relay 1 . The relay will be closed upon the terminal restart .

```
at^screl1=1,2
```

- The command closes relay 1 . Upon terminal restart, the relay will be in the state in which it was before restart .

RS 232 and RS 485/ M-Bus Interfaces

Find the interface connectors on the bottom side of 2N® SmartCom PRO. Refer to Subs. 2.1 . A 9-pin D-sub connector is used for RS-232 and a 3-pin WAGO terminal for RS 485 or M-Bus located on the bottom panel next to the power supply terminals. Follow the wiring instructions on the label. Use the commands below to set the RS 232 port parameters. Use the same commands to set the other ports but change the port address (scport2).

```
at^scport1=&quot;baudrate&quot;,9600
```

- The command sets the baud rate to 9600 bps.
  - The default baud rate value is 115200 bps.

```
at^scport1=&quot;baudrate&quot;,230400
```

- The command sets the baud rate to 230400 bps.
  - Set the baud rate in standard steps from 110 to 230400.

```
at^scport1=&quot;data_bits&quot;,8
```

- The command sets the data bit count to 8.
  - Setting options: 5–8.

```
at^scport1=&quot;stop_bits&quot;,2
```

- The command sets the stop bit count to 2.
  - Setting options: 1–2.

```
at^scport1=&quot;parity&quot;,2
```

- The command sets the parity security type:
  - 0 – none
  - 1 – even
  - 2 – odd
  - 3 – constant 1
  - 4 – constant 0

```
at^scport1=&quot;flowcontrol&quot;,1
```

- The command sets the flow control type:
  - 0 – no flow control.
  - 1 – hardware flow control.

```
at^scport1=&quot;save&quot;
```

- The command saves the changes.

```
at^scport1=&quot;restart&quot;
```

- The command restarts the interface.

```
at^scport1=&quot;srestart&quot;
```

- The command saves the changes and restarts the interface .

3.4 Input Calibration

A 10-bit A/D converter is connected to the IN1 and IN2 input terminals, which converts the measured value to a number ranging between 0 and 1023. Practically, the 0 - 1020 range is used; the remaining three steps are used for overvoltage detection.

The purpose of calibration is to compensate the inaccuracy of components and external factors and to ensure that the 960 value of the A/D converter matches exactly 10 V in each input. The value usually oscillates slightly around this point. Let us give an example: suppose you place two terminals next to each other under identical external conditions. One terminal will measure 966 and the other 957 for 10 V. Calibrate the input to eliminate such inconsistency.

> ⓘ **Note**
>
> - Keep in mind that an uncalibrated input is not non-functional. It is able to measure and read values less precisely, but sufficiently for rough voltage or current detection.

For calibration, use the terminals and AT commands described in **Sub. 4.2** , or the Terminal Config tool - refer to **Sub. 4.3** - for setting and application details. The following figure shows the calibration diagram and basic principle. The meanings of the curves are explained below.
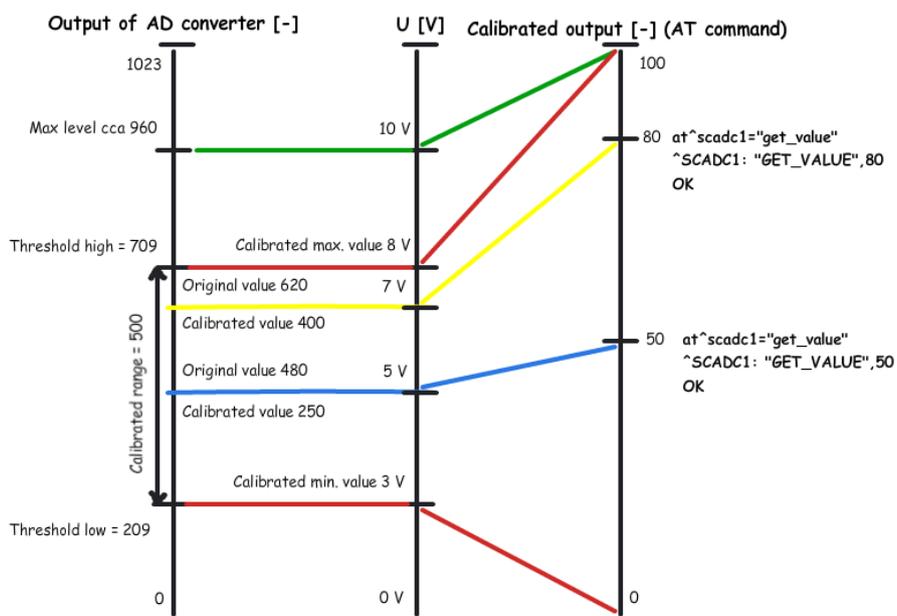
Figure: Calibration Diagram

– The green lines indicate that no calibration has been made yet. The A/D converter works in the whole range and deviations may occur from the actual value measured on the input. If you, in this situation, set Threshold low to 0V and Threshold high to 10V on the input, the terminal will be calibrated for 10V precisely. For 0 on the input, the AT command will return 0. For 10, we obtain 100. Thus, the calibrated input can be considered 0–100% of the calibrated range.

– The red lines mean that the input has been calibrated to the interval of 3 to 8V, i.e. to the range of 5V. This range now represents new measurement limits. If you apply 3V or less on the input, you are in the calibrated min area and the AT command will return 0. If you apply 8V or more, you are in the calibrated max area and the AT command will return 100. This means that the measured range width has narrowed by 50% and 0–100% is now mapped to the input range of 3–8V.

– The blue line is an example of measuring in the calibrated range of 0 to 10V. Apply precisely one half of the calibrated voltage, i.e. 5V, to the input. The AT command will return 50, i.e. one half of the 0–100 range again. In this case, 5V is the centre of the calibrated range. If, however, the range was shifted, the value would be shifted too as shown below.

– The yellow line shows the case that 7V is applied to the input. In case you measure 7V in the calibrated 10V range, the AT command will return 70 as 70 represents 70% of the calibrated range. But if the range is just 3 to 8V as in the red case, 7V will represent 80% of the calibrated range of 5V and the AT command will return 80.

> ⚠ **Caution**
>
> - If you use the calibrated input for reading logic values, keep in mind that the calibrated area is considered unstable. **Values lower than the set minimum will be 0 and those exceeding the maximum will be 1.** No value is guaranteed in an unstable area and a spurious state change may occur any time in the interval. Practically, the decisive level is near the middle of the set interval.

3.5 User Defined Functions - UDF

UDF provides a user–defined set of conditions for the terminal that are necessary for execution of the set actions. The conditions must be met to make it possible to send SMS/ TCP messages or set the relay position on the given terminal.

Up to 10 UDF conditions can be set for a terminal.

Each condition is evaluated continuously and, if **True**, the defined action will be executed. For re-execution, the condition must meet the **False** status or remain in the set status for a timer–defined period. Thus, SMS will not be sent at every iteration.

> ⚠ **Caution**
>
> - Follow tht the timer t[xx] in seconds, where xx is the waiting time. Refer to the Timer subsection for de general programming principles while creating the conditions to avoid nonsensical syntaxes and/or repetition of identical conditions. Such meaningless statuses cannot be processed effectively due to a limited memory and power capacities of the terminal.

Set the conditions using the AT commands in the terminal configuration. Each AT command must include the condition syntax (<parameter1>sign<parameter2>) followed by the separator ':', action type and action parameters.

A typical AT commands looks as follows:

```
AT^SCUDF="ADD","AIN1>=50:SMS P[+420123456789]m[alarm1] t[60]"
```

> ⚠ **Caution**
>
> - Make sure that each command includes the condition (syntax), separator (:) and action to be executed ("**<syntax>:<action>**") to avoid command setting error.

> ⚠ **Caution**
>
> - The maximum length of an AT command is 80 characters.

> ⓘ **Note**
>
> - Any error occurring during condition saving is signalled by the ERROR message and identified by a code if necessary. Refer to the **Error Codes** subsection below for the list of available messages.

Basic AT Commands

This subsection provides all the AT commands and parameters that can be used for setting of UDF conditions. The basic commands are described first and then their parts are explained. These are the basic commands for the 2N® SmartCom PRO UDF.

```
AT^SCUDF="ADD","<code>"
```

- The "ADD" command adds a condition to the list of conditions. The command section marked as the code is the condition itself. The terminal replies OK after saving.

> ⚠ **Caution**
>
> - When the OK message is received, the condition will be listed and available for work. Therefore, mind that an SMS or TCP message can be sent or the relay can be activated immediately.

```
AT^SCUDF="CLEAR"
```

- Enter the command to clear all conditions from the terminal memory.

```
AT^SCUDF="REMOVE",<ID>
```

- Enter the command to clear the condition with the selected ID.

```
AT^SCUDF?
```

- Enter the command to list all the conditions saved.

```
AT^SCUDF=?
```

- Enter the command to display the setting options.
- ^SCUDF: "ADD","0–100"
- ^SCUDF: "REMOVE",1–10
- ^SCUDF: "CLEAR"

Supported Syntaxes

The condition parameters can be entered in variable ways. For universality, each of the parameters can be a digit, i.e. an analogue input value or a binary input (0–1). Set a specified syntax to select one of the available setting options.

The UDF can work with the following parameters:

<paramX>	Description
AIN1	Analogue input 1
AIN2	Analogue input 2
DIN1	Digital input 1
DIN2	Digital input 2
PUL1	Pulse input 1
PUL2	Pulse input 2
CONx	Reference condition where x = condition <id>
CNTx	Internal counter, where x = counter <id> (1 - 5)
Constant	Value to be used for comparison with paramX
NFG	Network Fail GSM / UMTS

> ⚠️ **Caution**
>
> - The parameters that are listed as digits, i.e. **AIN** and **PUL**, which may be 0 – 100, or 0 – 4294967295, are converted to logic values as follows: if their value is **0**, their state will be **0**.
> - Make sure that **SC PING** is set to make the **NFG** syntax evaluate connection states correctly.

The following operators can be used for evaluation:

<operator>	Description
>	param1 is greater than param2
<	param1 is smaller than param2
==	param1 is equal to param2
>=	param1 is greater than or equal to param2
<=	param1 is smaller than or equal to param2
&&	Conjunction
||	Disjunction
!	Negation

Use the standard syntax, i.e. **<param1><operator><param2>**, to compare the measured value. However, you can just write **<param1>.** If you omit the operator and enter one parameter only, the parameter is automatically considered as logic even with **AIN** and **PUL**. When this parameter becomes effective, the set action will be executed.

Let us show you some syntax examples now.

AIN1>50 – the syntax will be marked true in case the value measured on AIN1 is higher than 50.

PUL2==2596 – the syntax will be marked true in case the value measured on the PUL2 counter is equal to the constant.

DIN1&&CON3 – the syntax will be marked true in case the DIN1 value is true and the reference condition is also true.

!AIN2 – the syntax will be marked true in case any value higher than 0 is measured on the input.

AIN1<=AIN2 – the syntax will be marked true in case the value measured on AIN1 is lower than the value measured on AIN2.

NFG - the syntax will be marked true (i.e. network failure) in case PING failed or the IP address was not obtained during the time set at at^scping="interval",<minutes>. When the IP address is obtained, the state will be cancelled and the condition will become FALSE. See below for the PING settings.

> ⚠️ **Caution**
>
> - If you use Negation, the parameter will be treated as a digital input! It cannot be expected, for example, that if 50 is measured on AIN1, -50 can be obtained after negation.
> - To match two inputs use the absolute values obtained from the converter (ADC 0 - 1020). Therefore, make sure that the two inputs have been calibrated properly.

Action Types

Insert an action after the separator for the selected syntax.

SAVE

Enter SAVE to save the condition result into the terminal memory for later use.

> ℹ️ **Note**
>
> - A typical condition saving AT command looks as follows:
>
> ```
> At^scudf="add","AIN1>50:SAVE"
> ```

REL1

The action controls the relay 1 state.

REL2

The action controls the relay 2 state.

> ⓘ **Note**
>
> - A typical relay switching AT command looks as follows:
>
> ```
> At^scudf="add","AIN1>50:REL1 r[1]"
> ```

SMS

Enter SMS to send the defined SMS to the defined telephone number.

> ⓘ **Note**
>
> - A typical SMS sending AT command looks as follows:
>
> ```
> At^scudf="add","AIN1>50:SMS p[+420123456789] m[text 1]"
> ```

> ⚠ **Caution**
>
> - Make sure that the telephone number and message parameters are completed to set this function successfully.

TCP

Enter TCP to send the defined TCP message to the defined IP address and port.

Use telnet for TCP sending.

> ⓘ **Note**
>
> - A typical TCP sending AT command looks as follows:
>
> ```
> At^scudf="add","AIN1>50:TCP i[62.134.22.87:8026] m[text 1]"
> ```

> ⚠️ **Caution**
>
> - Make sure that the telephone number and message parameters are completed to set this function successfully.

> ⓘ **Warning**
>
> - Make sure that the TCP receiving server exists to avoid process slow-down due to the terminal waiting for the reply from a non-existent server and not executing new actions.

CNT 1-5

Enter CNT 1-5 to control all of the 5 internal counters.

> ⓘ **Note**
>
> - The AT command for counter increment by 1 can be as follows, for example:
>
> ```
> At^scudf="add","AIN1>50:CNT1 d[1]"
> ```

Action Parameters

Use the parameters of the actions mentioned above to change the relay states and set the address for SMS/TCP sending. Enter the parameters in the format <mark>[value] and separate them with a space on both ends. There is no specified order for the parameters.

IP address

Set the IP address **i[xxx.xxx.xxx.xxx]** and default port 80 for the TCP messages. To change the port, enter **i[xxx.xxx.xxx.xxx:yyyyy]**, where **x** is the IP address of the remote server and **y** is the port to be used for TCP sending.

> ⚠️ **Caution**
>
> - Enter the IP address for the **TCP** action only as it is meaningless and signalled as error for the other actions.

Telephone number

Set the telephone number **p[xx]**, where **xx** is the telephone number in the national or international format. Hence, enter **p[123456789]** or **p[+420123456789].**

> ⚠️ **Caution**
>
> - The telephone number can be used for the **SMS** action only.
> - Enter telephone numbers consisting of 1 – 15 characters into the parameter.

Relay state

Set the relay contact position **r[1 or 0]**.

1 means that the relay is closed and **0** means that the relay is open.

> ⚠️ **Caution**
>
> - The **r** parameter can be used for the **REL** action only.

Timer

Set the timer **t[xx]** in seconds, where **xx** is the waiting time. Refer to the **Timer** subsection for details.

> ⚠️ **Caution**
>
> - The timer can be used for the **TCP** and **SMS** actions only.

Message

Set the message **m[text]**. The maximum message length is 40 characters. Longer messages will be rejected.

> ⓘ **Note**
>
> - The maximum of 40 characters is derived from the maximum length of the SMS to be sent. If the maximum **PUL** key values are used, the length is 160 characters.

> ⚠️ **Caution**
>
> - The message can be used for the **TCP** and **SMS** actions only.

You can also insert the following keys into the message parameter to show the value currently measured on the input circuits or the current logic value of the conditions in the SMS.

Key	Description
$AIN1	Analogue input 1 value
$AIN2	Analogue input 2 value
$DIN1	Digital input 1 value
$DIN2	Digital input 2 value
$PUL1	Pulse input 1 value
$PUL2	Pulse input 2 value
$CNT1	Counter 1 value
$CNTx	Counter x value (1-5)
$CON1	Condition 1 result value 1
$CONx	Condition x result value (1-10)

Let us show you an example for illustration.

Suppose you want to enter a command into the terminal, which, when the limit value 80 on input 1 is exceeded, will send an SMS to number 123456789 including the **ALARM value is:** (the currently measured value).

The AT command looks as follows:

```
AT^SCUDF="ADD","AIN1>80:SMS p[123456789] m[ALARM value is:$AIN1]"
```

The following SMS will be sent to the set number:

```
ALARM value is:87
```

> ⚠️ **Caution**
>
> - Should you enter an invalid or incomplete key, the key will not be included in the SMS sent.

> ⊘ **Tip**
>
> - Enter a special key $$ to display the $ character in the SMS as the single $ character introduces other keys.

Timers

There are three ways how to make use of condition timing. See below.

Timer not set

The action is executed when the condition is evaluated as true. For re-execution, the condition must be false and then true again.

Timer

Set the **t[60]** string, where **t** identifies a timer. Number **60** is the timer delay in seconds. The timer setting options are 5 – 4294967s.

The action is executed when the condition is evaluated as true. For re-execution, the condition must be false and then true again, and the set timeout must be over.

Continuous timer

Set the **t[c120]** string, where **t** identifies a timer, **c** indicates a continuous timer and number **120** is the timer delay in seconds. The timer setting options are 5 – 4294967s.

The action is executed when the condition is evaluated as true. The action is re-executed after the timeout if the condition is true. The condition need not be evaluated as false before re-execution.

Counters

There 5 virtual counters altogether in the terminal program for counting actions or storing current values for later computation. The following two parameters are used for control:

Counter incrementing

Enter string **d[1].** The 'd' parameter d efines that the counter value increases by the value included in the brackets (one step). You can enter 1, 5, 254, etc. and also negative values, which will be subtracted from the set value.

Counter setting

Enter string **s[4658].** The 's' parameter defines the counter setting value and number **4658** sets the counter value.

Error Codes

This table includes the error codes you can use for UDF setting. An incorrect command is signalled by ERROR <err_code>. The table helps you identify the type of error.

<err_code>	Name	Description
1	UDF_RESP_ERROR	General error
2	UDF_RESP_ERROR_NO_FREE_SLOT	No space for new record
3	UDF_RESP_ERROR_SYNTAX_ACTION_SEPARATOR	Separator ':' not found
4	UDF_RESP_ERROR_SYNTAX_LOGIC_MARK	Logic mark not found or wrong
5	UDF_RESP_ERROR_SYNTAX_PARAM1	Condition parameter 1 not found or wrong
6	UDF_RESP_ERROR_SYNTAX_PARAM2	Condition parameter 2 not found or wrong
7	UDF_RESP_ERROR_SYNTAX_INVALID_CONDITION	Condition not supported
8	UDF_RESP_ERROR_SYNTAX_ACTION	Action syntax error
9	UDF_RESP_ERROR_SYNTAX_ACTION_TYPE	Action type not supported

10	UDF_RESP_ERROR_PHONE_NUMBER_LEN	Telephone number length error
11	UDF_RESP_ERROR_PHONE_NUMBER	Telephone number not found or not supported
12	UDF_RESP_ERROR_IP_ADDRESS	IP address not found or not supported
13	UDF_RESP_ERROR_IP_PORT	IP port wrong
14	UDF_RESP_ERROR_RELAY_STATE	Relay state not found or not supported
15	UDF_RESP_ERROR_PARAM_PARSE	Action parameter translation error
16	UDF_RESP_ERROR_PARAM_MARK	Unknown action parameter type
17	UDF_RESP_ERROR_PARAM_MARK_TIMING_PERIOD	Timing period wrong or beyond limits
18	UDF_RESP_ERROR_PARAM_MESSAGE_LENGTH	Message length error
19	UDF_RESP_ERROR_ID_NOT_FOUND	Reference condition ID not found
20	UDF_RESP_ERROR_PARAM_TYPE_INVALID	Action parameter not supported for selected action type
21	UDF_RESP_ERROR_ACTION_IN_PROGRESS	Condition action in progress
22	UDF_RESP_ERROR_MESSAGE_MISSING	Message to be sent not found
23	UDF_RESP_ERROR_PARAM_VALUE	Value format or range is invalid

SC Ping

Enter the commands to set periodical pinging to the set address in defined intervals. Use SC Ping to ensure the correct function of NFG syntax in UDF or keep firewall connection if the terminal is in the STANDALONE mode.

```
at^scping=&quot;host&quot;,&quot;8.8.8.8&quot;
```

- Enter the command to set the Host IP address to which the ICMP request shall be sent.

```
at^scping=&quot;interval&quot;,5
```

- Enter the interval in minutes. 0 means that periodical pings are deactivated.

```
at^scping=&quot;test&quot;
```

- Enter the 'test' command to test configuration before saving. The currently changed value is always saved and thus no restart is needed during setting and functionality check.

```
at^scping=&quot;save&quot;
```

- Enter the command to save the changes.

```
at^scping=&quot;restart&quot;
```

- Enter the command to restart the function.

```
at^scping=&quot;srestart&quot;
```

- Enter the command to save the changes and reset the ping parameters immediately.

```
at^scping?
```

- Enter the command to display the current interface values.
    - SCPING: "HOST","8.8.8.8"
    - SCPING: "INTERVAL",0

```
at^scping=?
```

- Enter the command to display the interface setting options.

```
SCPING: "HOST","1-32"
SCPING: "INTERVAL",0-65534
SCPING: "TEST"
SCPING: "SAVE"
SCPING: "RESTART"
SCPING: "SRESTART"
```

3.6 Wireless M-Bus

Wireless M-Bus Interface

This function is only available in **2N$^®$ SmartCom PRO** and **2N$^®$ SmartCom PRO ETH**, which includes an optional Wireless M-Bus communication module. There are different WM-Bus interfaces according to the module mounted (radio frequency, e.g.). You can define up to 64 meters for which the device will collect readings. Having received a

WM-Bus message from a meter defined in the meter list, **2N$^®$ SmartCom PRO** will store it into a circular buffer in the internal memory. The stored data can only be retrieved and deleted starting from the oldest record. When the memory is full, the oldest data will be deleted automatically to make space to new records. Define the minimum reading time interval for each meter. Any meter reading coming before the end of this interval will not be stored.

The 868MHz WM-Bus module can work in four modes: T1, S1, C1 and T1+C. The 169MHz module works in the N mode and the 433MHz module in the T1 and C1 modes. They support encrypted messages too in all the modes. In the basic configuration, the terminal does not decrypt messages and stores them as they are received. Enter the correct encryption key and enable decryption to make the messages be decrypted by the module and stored in the decrypted format in the terminal memory. For security reasons, encryption keys are not stored in the memory, but written directly into the module. Therefore, if decryption does not work, check the device transmission and device settings in the terminal and re-enter the encryption key.

> ⊘ **Tip**
>
> - **2N$^®$ SmartCom PRO can be equipped with two WM-Bus modules. The other WM-Bus module replaces the ZigBee module.**
> - This optimisation is useful whenever you need the 868MHz and 169MHz modules at the same time or use the C1+T1 and S1 modes on one frequency.
> - Use the **AT^SCWMBUS2** command to configure the other module. A complete set of commands will then become available. These commands are only available if both the WM-Bus modules are mounted .

Wireless M–Bus Command List

These commands are implemented in the 2N® SmartCom PRO unit that is equipped with an optional Wireless M–Bus module.

```
at^scwmbus=&quot;dev_add&quot;,&quot;MAN&quot;,12345678,600,01,07
```

- This command adds a meter to the meter list. MAN is a three-letter abbreviation of the manufacturer followed by the meter serial number. The next parameter is the minimum interval in seconds between two readings of the meter to be added. If 0 is set, all messages received from the meter are stored. The following two parameters are also necessary for meter identification: Version and Device. Ask the device manufacturer for these values.

```
at^scwmbus=&quot;decrypt&quot;,1
```

- The command enables decryption of the messages received by the module. Setting options: 0 - 1.

```
at^scwmbus=&quot;key_add&quot;,1,A5B95C144134DE257AF2ED4F384C7EB7
```

- The command sets the encryption key for meter 1 in the 32 HEX format.

```
at^scwmbus=&quot;dev_remove&quot;,1
```

- The command removes a meter from the meter list; the parameter gives the meter position in the list (1 - 64).

```
at^scwmbus=&quot;dev_clear&quot;
```

- The command clears the meter list.

```
at^scwmbus=&quot;get_oldest&quot;
```

- The command writes out the oldest record from the Wireless M–Bus meter reading data zone. Example of reply:

```
^SCDATA: "WMBUS",<timestamp>,<id>,<rssi>,<len>,"<binary data in
hexadecimal format>"
```

Meaning of the parameters:

- timestamp – record receiving time (UNIX timestamp).
- id – record id in the circular buffer.
- rssi – signal strength at message reception in dBm.
- len – message length in bytes.

> ⊘ **Tip**
>
> - Earlier firmware versions may include different message formats: ^SCWMBUS: <timestamp>,<id>,<rssi>,<len>,"<binary data in hexadecimal format>".
> - This message contains different data. Three bytes of information on the Radiocrafts module are added to its end.

```
at^scwmbus=&quot;get_oldest&quot;&lt;,n&gt;
```

- The command provides bulk message reading. The optional n parameter defines how many of the last messages shall be read.

```
at^scwmbus=&quot;del_oldest&quot;
```

- The command deletes the oldest record from the Wireless M–Bus meter reading data zone to allow you to read the next record using the "get_oldest" function.

```
at^scwmbus=&quot;del_oldest&quot;&lt;,n&gt;
```

- The command provides bulk message deletion. The optional n parameter defines how many of the last messages shall be deleted.

```
at^scwmbus=&quot;del_all&quot;
```

- The command deletes all records from the Wireless M–Bus meter reading data zone.

```
at^scwmbus=&quot;mode&quot;,11
```

- The command sets the working mode for the device:
  - 1 - T1
  - 3 - S1
  - 8 - C1
  - 11 - T1+C
  - 17 - N1

> ⓘ **Note**
>
> - The 17 - N1 mode is displayed and can be selected only if the terminal is equipped with a WM-Bus module that supports this mode.

```
at^scwmbus=&quot;get_space&quot;
```

- The command displays free memory space for WM-Bus message storing.

```
at^scwmbus=&quot;mess_cnt&quot;
```

- The command displays the count of the WM-Bus messages received.

```
at^scwmbus=&quot;RF_channel&quot;,1
```

- The parameter defines the WM-Bus transmission channel. Setting options: 1 - 10.

ID	RC1180 channel	RC1701 channel
1	868.050 MHz	169.406250 MHz (Channel 1a)
2	868.150 MHz	169.418750 MHz (Channel 1b)
3	868.250 MHz	169.431250 MHz (Channel 2a)
4	868.350 MHz	169.443750 MHz (Channel 2b)
5	868.450 MHz	169.256250 MHz (Channel 3a)
6	868.550 MHz	169.458750 MHz (Channel 3b)
7	868.650 MHz	169.437500 MHz (Channel g)
8	868.750 MHz	169.412500 MHz (Channel 1)
9	868.850 MHz	169.437500 MHz (Channel 2)
10	868.950 MHz	169.462500 MHz (Channel 3)
11	869.050 MHz	
12	869.150 MHz	

```
at^scwmbus=&quot;preamble_len&quot;,0
```

- The parameter specifies the framework format: 0 - means 4 (short) / Frame format A. 1 - 70 (long) / Frame format A.

```
at^scwmbus=&quot;Install_mode&quot;,0
```

- The parameter selects the normal/filter off mode. Normal means that it is necessary to define the modules from which data can be received. All the other modules will be ignored. Filter off means that the module receives and stores all data captured, even those from the modules that are not defined. 0 - normal. 2 - filter off.

```
at^scwmbus=&quot;restart&quot;
```

- The command restarts the WM-Bus module.

```
at^scwmbus=&quot;sniff&quot;
```

- The command switches the WM-Bus module into the sniff mode to capture all messages from the devices in the vicinity and display them in the console. When the command is entered, the connect message is displayed and the terminal starts writing out all the messages. Click Enter to quit. Disconnect is displayed for confirmation, The displayed message looks as follows:

```
at^scwmbus="sniff"<br>
CONNECT<br>
^SCWMBUS: "AMB",54001477,01,37,-1,-92dBm
```

The message format and meanings of the elements are as follows: ^SCWMBUS: "MAN", 12345678,VER,DEV,ID,SIGNAL

- MAN - manufacturer.
- 12345678 - device serial number.
- VER - meter version given by the manufacturer.
- DEV - quantity to be measured (water / gas) as given by the manufacturer.
- ID - ID in the terminal meter list; if no ID is assigned, the meter is not configured in the terminal and -1 is displayed.
- SIGNAL - signal level at message reception.

⊘ **Tip**

- Click ENTER to stop capturing.

```
at^scwmbus=&quot;module_type&quot;
```

- The command displays information on the WM-Bus module. Example of reply: ^SCWMBUS: ″MODULE_TYPE″,″RC1180-MBUS3,2.00,3.14″.

at^scwmbus=″cf_discard″,0

- The module is switched into a mode in which it discards messages captured in the Compact Frame (CF) format and stores only messages received in the Full Frame format.
  - 0 - disabled, all messages are stored.
  - 1 - enabled, CF are discarded.

```
at^scwmbus?
```

- The command displays the currently set interface values.

```
at^scwmbus=?
```

- The command displays the Wireless M–Bus setting options.

3.7 ZigBee interface

The ZigBee functionality is only available in the 2N® SmartCom PRO version equipped with a ZigBee interface. The ZigBee parameters depend on the module mounted (radio frequency, e.g.). You can define up to 16 terminals within one network to communicate with 2N® SmartCom PRO, which works as a coordinator for this network. The ZigBee communication technology, based on the IEEE 802.15.4 – ZigBee standard, is one of the wireless PANs (*Personal Area Networks*). PANs also include a very popular IEEE 802.15.1 – Bluetooth technology, which is mostly applied in consumer electronics. However, there are a lot of industrial applications where Bluetooth is inapplicable. Therefore, a new wireless communication standard has been introduced suitable for industrial automation too. At present, over 60 companies including the world automation leaders are involved in ZigBee development and extension. ZigBee has been designed as a simple and flexible technology for extensive wireless networks that require no huge data transmissions. ZigBee features reliability, easy and undemanding implementation, low power requirements and, last but not least, cost efficiency. Based on multi-step ad-hoc routing, ZigBee provides long-distance communication without direct radio visibility of the devices involved. It is primarily applied in industry and sensor networks working at 2.4 GHz in licenceless zones (general permission). The bit rate is 20, 40, 250 kbps. Thanks to these merits, ZigBee is widely applied in many areas. For our purpose, its use in the available functions will be described. See below for the list of parameters including descriptions.

> ⓘ **Note**
>
> - Check the front sticker or back panel label or use the SCZB AT command (**AT^SCZB?**) to know whether your 2N® SmartCom PRO is equipped with a ZigBee module. The terminals that are not equipped with ZigBee reply **ERROR**.

At present, 2N® SmartCom PRO behaves as a Master, i.e. the main network coordinator to which devices get connected. A sufficient amount of clusters have been mounted to help you comfortably control your air conditioners, ZigBee keypads, sockets, moisture and other meters, street lighting and many other systems.

> **ⓘ Note**
>
> - Contact us at sales@2n.cz for more details or integration capacities.

Now let us briefly describe the ZigBee properties. Each device is identified by its IEEE address. One or more endpoints (EPs) are defined for each device and function clusters are defined for each EP. One cluster may occur just once in each EP. Let us give a simple example. Suppose we have a device that measures temperature and is equipped with 3 relays that switch other devices. In this case, the device will have 3 EPs with one ONOFF cluster each and there will be a DEVTEMP cluster in one of them. The simplified record in the device list would be as follows:

```
^SCZB: ID: 6, STATE: ONLINE, IEEE_ADDR: 00:12:4B:00:04:26:35:C1, SIG: 43
%, UPDATED: 307332129283893
EP2: ONOFF: 0, DEVTEMP: 25,00,
EP3: ONOFF: 1,
EP4: ONOFF: 0,
```

2N® SmartCom PRO constitutes a wireless PAN to which devices log in. The meters are listed in a table. The count of the meters (n) is determined by the firmware version. There can be up to 32 devices depending on the customer's requirements. As the first step is to properly configure a wireless network, the first set of commands relates to network setup and device adding. The next step is description of clusters and related functions. Refer to the ZigBee alliance web sites for detailed specifications. Now let us describe the PAN configuration AT commands.

Network Configuration

```
AT^SCZB=&quot;ADD&quot;,&lt;16 HEX&gt;[,&lt;dev_id&gt;]
```

- The command adds a device to the list. <16 HEX> means a sixteen-character IEEE address. Ask your meter manufacturer for these data.
    - <dev_id> - this optional parameter specifies the meter list position. If it is not used, a new device is added to the first free list position.

> ⊘ **Tip**
>
> - The IEEE ZigBee addresses have the following format: **00:12:4B:00:04:26: 35:C1**. The device adding AT command is as follows:
>
> ```
> AT^SCZB="ADD",00124B00042635C1,6
> ```
>
> - The command adds a device with the above mentioned address to position 6 of the list.
> - The count of the list positions depends on the FW version and is customer adjustable.

```
AT^SCZB?
```

- The command reads out the meter list. The reply can be as follows:

```
AT^SCZB?<br> ^SCZB: ID: 1, STATE: CONFIGURED, IEEE_ADDR: 00:12:4B:00:05:
18:E7:42, SIG: 0%, CAP: UNKNOWN<br> ^SCZB: ID: 2, STATE: CONFIGURED,
IEEE_ADDR: 00:12:4B:00:04:26:35:C1, SIG: 0%, CAP: UNKNOWN<br> ^SCZB:
ID: 3, STATE: SLOT_FREE, IEEE_ADDR: 00:00:00:00:00:00:00:00, SIG: 0%,
CAP: UNKNOWN<br> ^SCZB: ID: 4, STATE: SLOT_FREE, IEEE_ADDR: 00:00:00:00:
00:00:00:00, SIG: 0%, CAP: UNKNOWN<br> ^SCZB: ID: 5, STATE: SLOT_FREE,
IEEE_ADDR: 00:00:00:00:00:00:00:00, SIG: 0%, CAP: UNKNOWN<br> ^SCZB:
ID: 6, STATE: CONFIGURED, IEEE_ADDR: 00:12:4B:00:04:26:35:C1, SIG: 0%,
CAP: UNKNOWN<br> ^SCZB: ID: 7, STATE: SLOT_FREE, IEEE_ADDR: 00:00:00:00:
00:00:00:00, SIG: 0%, CAP: UNKNOWN<br> ^SCZB: ID: 8, STATE: SLOT_FREE,
IEEE_ADDR: 00:00:00:00:00:00:00:00, SIG: 0%, CAP: UNKNOWN<br> ^SCZB:
ID: 9, STATE: SLOT_FREE, IEEE_ADDR: 00:00:00:00:00:00:00:00, SIG: 0%,
CAP: UNKNOWN<br> ^SCZB: ID: 10, STATE: SLOT_FREE, IEEE_ADDR: 00:00:00:00
:00:00:00:00, SIG: 0%, CAP: UNKNOWN<br> OK
```

```
AT^SCZB="DEL",<dev_id>
```

- The command removes a device from the list. The parameter setting depends on the count of records in the list. One record is only removed.

```
AT^SCZB="CLEAR"
```

- The command removes all devices from the list.

```
AT^SCZB="PANID"[,<4 HEX>]
```

- The command sets the network identifier to the value specified in the parameter.
  - <4 HEX> - gives the new network name. The setting options are 4 hexadecimal characters (O - F).

> ⊘ **Tip**
>
> - In case the optional parameter is not entered, the command only reads out the currently set value. This holds true for all the commands!

> ⓘ **Note**
>
> - If there is another network within reach with the same name and channel, the module will choose the more available channel or automatically change the network name to the set value +1. Therefore, you are advised to check the setting by reading the current value.

```
AT^SCZB="PANID_READ"
```

- The command reads out the current value of the PANID parameter.

```
AT^SCZB=&quot;CHANNEL&quot;[,&lt;8 HEX&gt;]
```

The command sets the used channel to the value set by the parameter.

- <8 HEX> - gives the new network name. The setting options are 8 hexadecimal characters (0 - F). The values of the last 8 positions are entered without 0x.
- It is a channel bit mask. The default value is 0x00000800 - channel 11.

Channel	Mask	Channel	Mask
NONE	0x00000000	Channel 18	0x00040000
ALL_CHANNELS	0x07FFF800	Channel 19	0x00080000
Channel 11	0x00000800	Channel 20	0x00100000
Channel 12	0x00001000	Channel 21	0x00200000
Channel 13	0x00002000	Channel 22	0x00400000
Channel 14	0x00004000	Channel 23	0x00800000
Channel 15	0x00008000	Channel 24	0x01000000
Channel 16	0x00010000	Channel 25	0x02000000
Channel 17	0x00020000	Channel 26	0x04000000

- In case the optional parameter is not entered, the command only reads out the currently set value. It does not return a mask but a string with the selected channel number.

```
AT^SCZB=&quot;CHANNEL_READ&quot;
```

- **The command only reads out the currently set value. It does not return a mask but a string with the selected channel number.**

```
AT^SCZB="PRECFGKEY"[,<32 HEX>]
```

- The command sets the pre-configured key for encrypted communication.
    - <32 HEX> - encryption string including 16 ASCII characters in hexadecimal format: 32 characters 0 - F.

```
AT^SCZB="PRECFGKEY_ENABLE"[,<0-1>]
```

- The command enables/disables sending of the encryption key.
    - If configured for both the communicating devices, the key need not be sent at communication start.

```
AT^SCZB="TCLK"[,<32 HEX>]
```

- The command sets the encryption key for communication key negotiation and transmission. Used for key replacement encryption.
    - <32 HEX> - encryption string including 16 ASCII characters in hexadecimal format: 32 characters 0 - F.

> ⊘ **Tip**
>
> - Make sure that the key is configured identically on both the communication ends before communication starts. The key is set to the default factory value in all ZigBee devices, which is identical for all the devices. Keep the default setting to ensure compatibility. Change the key to increase security. The default key is as follows:
>   - ASCII - ZigBeeAlliance09
>   - HEX - 5A6967426565416C6C69616E63653039

```
AT^SCZB="IEEE_ADDR"
```

- The command reads out and displays the ZigBee interface IEEE address.

```
AT^SCZB="MODULE_TYPE"
```

- The command reads out and displays information on the module.

```
AT^SCZB="RESTART"
```

- The command restarts the interface and creates a new PAN. If the main parameters are not changed, no new PAN is created.

```
AT^SCZB="POLL",<dev_id>
```

- The command reads out and updates the device list data for the defined dev_id (device list position).
  - One device is only updated. All available information including cluster values in endpoints are read out.

```
AT^SCZB="POLL_TIME",<time>
```

- The command reads out and updates the device list data automatically in predefined intervals. The timer is kept for each device separately.
  - <time> - time is set in seconds. Setting options: 0 - 10000, where 0 means disabled.

```
AT^SCZB="PERMIT_JOIN",<dev_id>,<time>
```

- The command permits a device to let new devices join the network.
  - <dev_id> - define the device(s) to be permitted to accept new devices. Setting 0 you enable permit_join for **2N® SmartCom PRO** directly.
  - <time> - time is set in seconds. Setting options: 0 - 255.
    - 0 - off
    - 1 - 254 on for a defined period of time
    - 255 - constantly on

> **⊘ Tip**
>
> - To add devices via Next Hop, make sure that permit join is enabled for **2N® SmartCom PRO** and the point that is the closest to the device to be added.

```
AT^SCZB=&quot;PERMIT_JOIN_ALL&quot;,&lt;time&gt;
```

- The command permits all the devices in the list to let new devices join the network.
  - <time> - time is set in seconds. Setting options: 0 - 255.
    - 0 - off
    - 1 - 254 on for a defined period of time
    - 255 - constantly on

- ## Description of Function Commands

Remember to enter the dev_id of the device to which the command is to be sent. Also enter the end_point (EP) to identify the cluster in the given device. The default EP is always 2, but can have any value theoretically: 2, 3 and 4 are the most frequent values. These two parameters are the requisites of the commands below and so their meanings will not be described in every command.

Every sent command is followed by a reply with one of the following 4 states:

- <DATA> <OK> - data parameter - displays description of the set/measured values and the command is terminated with an OK sequence: everything is OK.
- <ERROR> - an invalid parameter was sent or an error occurred while processing.
- <TIMEOUT> - the command was sent to the device but the device failed to reply within the timeout.
- <ZCLERROR:[code]> - an error was detected in the ZigBee protocol. A numerical error code is displayed. See the ZigBee datasheet for details.

> ⓘ **Note**
>
> - The device replies to the command if the cluster with the given parameter is implemented. Not all of the devices support all the clusters. The devices only use the clusters they need for their function.

```
AT^SCZB=&quot;ANLGVAL&quot;,&lt;dev_id&gt;,&lt;EP&gt;
```

- The command reads out the value of the analogue input in a device on the selected end_point.

```
AT^SCZB="BASIC",<dev_id>,<EP>
```

- The command reads out all the values included in the BASIC cluster.
  - The following information can be included, for example: (BASIC_ZCL_VER, BASIC_APP_VER, BASIC_STACK_VER, BASIC_HW_VER, BASIC_MAN_NAME, BASIC_MODEL_ID, BASIC_PWR_SRC, BASIC_LOC_DESC, ONOFF, POWCFG_MAINS_VOLTAGE, POWCFG_BATTERY_VOLTAGE).

```
AT^SCZB="DEVTEMP",<dev_id>,<EP>
```

- The command reads out the current device temperature.

```
AT^SCZB="FAN",<dev_id>,<EP>
```

- The command returns the current state of the fan.

```
AT^SCZB="FAN_MODE",<dev_id>,<EP>,<value>
```

- The command sets the fan mode to the defined value.

```
AT^SCZB="HMDT",<dev_id>,<EP>
```

- The command returns the moisture value in % with the accuracy of 2 decimal places.

```
AT^SCZB="LC",<dev_id>,<EP>
```

- The command returns the Level Control value (position of the aerodynamic air flow compensator, "flap")

```
AT^SCZB="LC_MOVE_TO_LEVEL",<dev_id>,<EP>,<level>[,<transition_time>]
```

- The command sets the Level Control value (flap position). The optional transition_time parameter in tenths of a second defines when the position is to be set. If time is not set, the change will be made immediately.

```
AT^SCZB="METER",<dev_id>,<EP>
```

- The command writes out the immediate consumption values. The output includes two numbers: the first specifies the immediate power output and the other the total output withdrawn (the parameter is adjusted to the needs of electric meters and so it displays the [W, Wh] units. In general, however, any meter can be used and the values then specify energy [J] or flow rate [m3/h]).

```
AT^SCZB="MSTVAL",<dev_id>,<EP>
```

- The command reads out the Multi State Value. Used mostly for error codes.

```
AT^SCZB="ONOFF",<dev_id>,<EP>
```

- The command returns the 1/0 value depending on the immediate setting.

```
AT^SCZB="ONOFF_SET",<dev_id>,<EP>,<state>
```

- The command sets the switch to the value specified in the state parameter. Setting options: 1 - on, 0 - off.

```
AT^SCZB="TEMP",<dev_id>,<EP>
```

- The command returns a temperature value in °C with the accuracy of two decimal places.

```
AT^SCZB="THERM",<dev_id>,<EP>
```

- The command reads out and displays the currently set thermostat mode.

```
AT^SCZB="THERM_MODE",<dev_id>,<EP>,<value>
```

- The command sets the thermostat mode to the defined value: 2 hexadecimal characters. Refer to the table below.

Value	Description
00	Off
01	Auto
03	Cool
04	Heat
05	Emergency heating
06	Precooling
07	Fan only
02, 08, fe	Reserved

```
AT^SCZB="THERM_TEMP_HEAT",<dev_id>,<EP>,<value>
```

- The command sets the heating temperature in °C with the accuracy of two decimal places.

```
AT^SCZB="THERM_TEMP_COOL",<dev_id>,<EP>,<value>
```

- The command sets the cooling temperature in °C with the accuracy of two decimal places.

```
AT^SCZB="TIME",<dev_id>,<EP>
```

- The command returns the currently set time value and time status.

```
AT^SCZB="TIME_SET",<dev_id>,<EP>,<utc_time>
```

- The command sets the device time to the value specified by the utc_time parameter in a 32-bit format.

```
AT^SCZB="GROUP_ADD",<dev_id>,<EP>,<group_id>
[,<group_name>]
```

- The command adds the device with the selected dev_id to the group defined in group_id. The optional group_name parameter helps you assign a name of the maximum length of 15 characters to a new group.

```
AT^SCZB="GROUP_MEMBERSHIP",<dev_id>,<EP>,<group_id>
```

- The command returns the YES string if the device defined in dev_id is assigned to the group defined in group_id. If not, the NO string is returned.

```
AT^SCZB="GROUP_REMOVE",<dev_id>,<EP>,<group_id>
```

- The command removes the device specified in dev_id from the group defined in group_id.

```
AT^SCZB="GROUP_REMOVE_ALL",<dev_id>,<EP>
```

- The command removes the device specified in dev_id from all the groups.

```
AT^SCZB="SCENE_RECALL_ALL",<group_id>,<scene_id>
```

- The command starts the scenario defined in scene_id in all the devices in the group defined in group_id.
  - Make sure that this scenario has been implemented in the devices. The command can only start it.

3.8 Autonomous Metering System (AMS)

The AMS solution provides an automatic remote reading of not only such appliances as electric meters, gas meters or water meters, but also variable sensors and detectors. Data are read, remotely collected and transmitted to central software. This provides the building manager or power supplier with fresh consumption data for further processing (service billing, e.g.). IEC 62056-21 or M-Bus via the RS 232/RS 485 /M-Bus interface is used for meter communication. Data can thus be read out

periodically and stored in the 2N® SmartCom PRO internal memory for sorting and filtering. Integration of other communication protocols can be solved according to the customer's needs.

The AMS is configured via the AT interfaces described below. Log in via your terminal

to an AMS-supporting 2N® SmartCom PRO and follow the instructions below. Add up to 64 independent meters to the meter list. The data space size is determined by the HW version. All records are written into a circular buffer, which means that whenever the memory is full, the oldest record will be deleted and replaced with a new one.

> ⓘ **Note**
>
> - The standard baud rate is 300 bps. 2N® SmartCom PRO can start communication at a higher rate if necessary, which is then decreased to the standard. To achieve the typical increased initial rate of 2400 bps, set a higher rate on port RS 485 in the 2N® SmartCom PRO configuration. When the AMS communication stops at 300 bps, the preset higher rate is reset automatically.
>
> - 2N® SmartCom PRO supports the A, C and E modes according to IEC 62056-21.

AMS Device Configuration

The commands below are used for meter configuring.

```
AT^SCAMS=?
```

The command writes out the list of AMS module control commands, their parameters and parameter formats.

Reply:

```
^SCAMS: "ENABLE",0|1
^SCAMS: "RETRY",0-9
^SCAMS: "DEV_ADD",1-1000,"32"|1-1000,"32","3"
^SCAMS: "DEV_REMOVE",1-32
^SCAMS: "DEV_CLEAR"
^SCAMS: "DEV_INTERVAL","ALL"|1-32,"3"
^SCAMS: "DIRECT_READ",1-1000,"32"|1-1000,"32",110-230400,0|1
OK
```

```
AT^SCAMS?
```

The command writes out the contents of the device list to be served by the AMS module.Table row format: *^SCAMS: <index>> <port>,<protocol>,<address>,<interval>*

- *<index> - device index in the device list - number*
- *<port> - number of the physical port to which the device is connected - number*
    - *2 - RS232/RS485/MBUS communication port (depending on HW version)*
    - *128 - internal A/D input module communication port (analogue/digital input module)*
- *<protocol> - name of the communication protocol used for device communication - text string*
    - *"IEC62056-21" - IEC 62056-21 protocol*
    - *"MBUS" - M-Bus protocol*
    - *"AIN_DIN" - A/D input protocol (analogue/digital input module data reading protocol)*
- *<address> - device address - text string*
    - *device address in the format corresponding to the device address format for the given protocol*
- *<interval> - data reading time interval - time interval symbol text string*
    - *minutes, hours or days; setting options:*
    - *"1M", "2M", "3M", "4M", "5M", "6M", "10M", "12M", "15M", "20M", "30M", "1H", "2H", "3H", "4H", "6H", "8H", "12H", "1D"*

Example of a reply if the list is empty:

```
^SCAMS: NO DEVICE
OK
```

```
Example of a reply if the list includes 4 devices:
```

```
^SCAMS: 1> 2,"IEC62056-21","869551","1M"
^SCAMS: 2> 2,"IEC62056-21","869552","15M"
^SCAMS: 3> 128,"AIN_DIN","1/ADC/CAL","1M"
^SCAMS: 4> 128,"AIN_DIN","2/PULSE/ABS","1H"
OK
```

```
AT^SCAMS="ENABLE",&lt;zapnuto&gt;
```

The AMS module command writes out the current state (on/off) of the AMS module (without additional details) or sets the on/off state for the AMS module (according to the parameter below).

- *<on> - if included, the parameter specifies the required new AMS on/off state and the command sets the new state; if the parameter is not included, the command writes out the currently set state - logical value*
  - *0 - AMS off request*
  - *1 - AMS on request*

Reply format (AMS set to on):

```
^SCAMS: "ENABLE",1
OK
```

Reply format (AMS state setting command) - action succeeded:

```
OK
```

Reply format (AMS state setting command) - action failed

```
ERROR
```

```
AT^SCAMS="RETRY"[,&lt;retry_count&gt;]
```

The AMS module command writes out the current setting of the count of the data reading retries in case reading failed (without any additional parameters) or sets the count of the device data reading retries (according to the parameter below).

- *<retry_count> - if included, the parameter specifies the required device data reading attempts in case reading failed with an error, which allows for retrying data reading; if the parameter is not included, the command writes out the currently set retry count value.*
  - *number, range: 0 to 9*

Reply format (currently set value identifying command):

```
^SCAMS: "RETRY",<current_retry_count>
OK
```

Example (find the currently set device data reading attempts):

```
AT^SCAMS="RETRY"
```

Example (set the count of the device data reading retries; when the first data reading attempt fails, up to 2 more attempts will follow): AT^SCAMS="RETRY",2

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

```
AT^SCAMS="DEV_ADD",&lt;port&gt;,&lt;protocol&gt;,&lt;address&gt;,&lt;
interval&gt;
```

The AMS module command adds a device to the device list .

- *<port> - number of the physical port to which the device is connected - number*
  - *2 - RS232/RS485/M-Bus communication port (depending on the HW version)*
  - *128 - internal A/D input module communication port (analogue/digital input module)*
- *<protocol> - name/number of the communication protocol used for device communication*
  - *numerical format (number) or symbolic name (text string)*
    - *for number*
      - *1 - denotes the IEC 62056-21 protocol*
      - *2 - denotes the M-Bus protocol*
      - *128 - denotes the A/D input reading rotocol (for reading data from the analogue/digital input module)*
    - *for text string*
      - *"IEC62056-21" - denotes the IEC 62056-21 protocol*
      - *"MBUS" - denotes the M-Bus protocol*
    - *"AIN_DIN" - denotes the A/D input reading protocol (for reading data from the analogue/digital input module)*
- *<address> - device address - text string*
  - *for IEC 62056-21 the address consists of a text string <address> [@ <default_baudrate>][% <switches>], where:*
    - *<address> - device address (identical with the original device address)*
    - *< default_baudrate > - an optional parameter defining the default baud rate in Bauds*
    - *<switches> - an optional parameter - switch array for data reading process control. Now an 'R' switch is only implemented to prevent a change of the default baud rate*
  - *for M-Bus the address consists of a text string <address>@<baudrate>*
    - *<address> - M-Bus device (meter) address*
      - *primary address - <address> is a number ranging from 0 to 250*
      - *secondary address - <address> has the format *<address_8>, where <address_8> consists of 8 characters of the secondary address, where each character is:*
        - *a digit ranging from 0 to 9*

- a wildcard 'F' character (substituting any numerical value on the given position in the range between 0 and 9)
- *<baudrate>* - communication rate in Bauds; if the parameter is not included, the default value is used. Setting options: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400. Default value: 2400.
- *for A/D input protocol the address consists of a text string* <input>/<parameter>[/<option>]
  - *<input>* - number of the input for which data reading is to be made, setting options:
    - *1 - input 1*
    - *2 - input 2*
  - *<parameter>* - text string specifying the data to be read; setting options:
    - *ADC - analogue signal measurement on an analogue input*
    - *DIN - logical value reading on analogue input; result values: 0 to 1*
    - *PULSE - digital input pulse counter reading*
  - *<option>* - text string specifying the parameter options; setting options:
    - *for ADC:*
      - *CAL - the data reading result is a calibrated value ranging from 0 to 100*
      - *ABS - the data reading result is a value obtained directly from the A/D converter ranging from 0 to 1023*
    - *for DIN - <option> is undefined and thus unavailable for this parameter*
    - *for PULSE:*
      - *ABS - the data reading result is the current pulse count read in the input pulse counter ranging from 0 to (2^64 - 1)*
      - *REL - the data reading result is the pulse count read after the last data reading (or pulse counter deletion/resetting) ranging from 0 to (2^64 - 1)*
- *<interval>* - data reading interval - time interval symbol text string
  - *count of minutes, hours or days; setting options:*
  - *"1M", "2M", "3M", "4M", "5M", "6M", "10M", "12M", "15M", "20M", "30M", "1H", "2H", "3H", "4H", "6H", "8H", "12H", "1D"*

— Examples of Use —

As configuration is not easy, let us give a few examples to help you set the required function(s) successfully. Example 1 - suppose you want to add a device to the device list, which is connected to physical port 2, communicates via IEC 62056-21, the device address is 869551 and the data reading interval is 3 minutes. The command is as follows:

```
AT^SCAMS="DEV_ADD",2,"IEC62056-21","869551","3M"
```

Example 2 - same as above, but the protocol is given in a numerical format:

```
AT^SCAMS="DEV_ADD",2,1,"869551","3M"
```

Example 3 - suppose you want to add a device to the device list, which is connected to physical port 2, communicates via M-Bus, the device address is 128, the baud rate is 2400 Bd and the data reading interval is 1 hour:

```
AT^SCAMS="DEV_ADD",2,"MBUS","128@2400","1H"
```

Example 4 - same as above, but the baud rate is not specified and so the default baud rate of 2400 Bd is used:

```
AT^SCAMS="DEV_ADD",2,"MBUS","128","1H"
```

Example 5 - same as above, but the protocol is given in a numerical format:

```
AT^SCAMS="DEV_ADD",2,2,"128","1H"
```

Example 6 - suppose you want to add a device to the device list, which is connected to physical port 2, communicates via M-Bus, the secondary device address is 01579411, the baud rate is 2400 Bd and the data reading interval is 1 hour:

```
AT^SCAMS="DEV_ADD",2,"MBUS","*01579411@2400","1H"
```

Example 7 - s uppose you want to add a device to the device list, which is connected to physical port 2, communicates via M-Bus, the secondary device address with two wildcard positions is 015794FF, the baud rate is 2400 Bd and the data reading interval is 1 hour:

```
AT^SCAMS="DEV_ADD",2,"MBUS","*015794FF@2400","1H"
```

Example 8 - s uppose you want to add a device to the device list, which is connected to the A/D input port, and the interval of calibrated analogue value reading on analogue input 1 is 1 hour:

```
AT^SCAMS="DEV_ADD",128,"AIN_DIN","1/ADC/CAL","1H"
```

Example 9 - s uppose you want to add a device to the device list, which is connected to the A/D input port, and the interval of logical value reading on analogue input 2 is 2 hours:

```
AT^SCAMS="DEV_ADD",128,"AIN_DIN","2/DIN","2H"
```

Example 10 - same as above, but the protocol is given in a numerical format

```
AT^SCAMS="DEV_ADD",128,128,"2/DIN","2H"
```

Example 11 - s uppose you want to add a device to the device list, which is connected to the A/D input port, and the interval of pulse counter reading on digital input 1 is 2 hours:

```
AT^SCAMS="DEV_ADD",128,"AIN_DIN","1/PULSE/ABS","2H"
```

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

— End of Examples —

```
AT^SCAMS="DEV_REMOVE",&lt;index&gt;
```

The AMS module command removes a device from the device list.

- *<index> - index of the device (in the device list) to be removed*

Example - suppose you want to remove the index 4 device from the device list:

```
AT^SCAMS="DEV_REMOVE",4
```

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

```
AT^SCAMS="DEV_CLEAR"
```

The AMS module command removes all the devices from the device list.

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

```
AT^SCAMS="DEV_INTERVAL",&lt;index&gt;,&lt;interval&gt;
```

The AMS module command changes the data reading time interval for a device included in the device list.

- *<index> - index of the device (in the device list) for which time is to be changed*
  - *numerical format (number) or symbolic name (text string)*
    - *for number: index of the device (in the device list) for which time is to be changed*
    - *for text string: "ALL" - change time for all the devices included in the device list*
- *<interval> - time interval symbol text string (minutes/hours/days), setting options:*
  - *"1M", "2M", "3M", "4M", "5M", "6M", "10M", "12M", "15M", "20M", "30M", "1H", "2H", "3H", "4H", "6H", "8H", "12H", "1D"*

Example 1 - suppose you want to change the data reading interval to 1 hour for the index 3 device:

```
AT^SCAMS="DEV_INTERVAL",3,"1H"
```

Example 2 - suppose you want to change the data reading interval to 30 minutes for all the devices listed:

```
AT^SCAMS="DEV_INTERVAL","ALL","30M"
```

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

```
AT^SCAMS="DIRECT_READ",&lt;port&gt;,&lt;protocol&gt;,&lt;address&gt;,
{&lt;extra_parameter1&gt;[,&lt;extra_parameter2&gt;]}
```

The AMS module command performs immediate data reading from a device and writes out the data.

- *<port> - number of the physical port to which the device is connected - number*
  - *2 - RS232/RS485/MBUS communication port (depending on HW version)*
  - *128 - internal A/D input module communication port (analogue/digital input module)*
- *<protocol> - name of the communication protocol used for device communication*
  - *for number:*
    - *1 - denotes the IEC 62056-21 protocol*
    - *2 - denotes the M-Bus protocol*
    - *128 - denotes the A/D input protocol (for reading data from the analogue/digital input module)*
  - *for text string:*
    - *"IEC62056-21" - denotes the IEC 62056-21 protocol*
    - *"MBUS" - denotes the M-Bus protocol*
    - *"AIN_DIN" - denotes the A/D input protocol (for reading data from the analogue/digital input module)*
- *<address> - device address - text string*
  - *for IEC 62056-21 - the address consists of a text string <address> [@ <default_baudrate>][% <switches>], where:*
    - *<address> - device address (identical with the original device address)*
    - *< default_baudrate > - an optional parameter defining the default baud rate in Bauds*
    - *< switches > - an optional parameter - switch array for data reading process control. Now an 'R' switch is only implemented to prevent a change of the default baud rate*
  - *for M-Bus- the address consists of a text string <address>[@<baudrate>]*
    - *<address> - M-Bus device (meter) address*
      - *primary address - <address> is a number ranging from 0 to 250*
      - *secondary address - <address> has the format *<address_8>, where <address_8> consists of 8 characters of the secondary address, where each character is:*
        - *a digit ranging from 0 to 9*
        - *a wildcard 'F' character (substituting any numerical value on the given position in the range between 0 and 9)*
    - *<baudrate> - communication rate in Bauds; if the parameter is not included, the default value is used. Setting options: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400. Default value: 2400 .*

- *for A/D input protocol - t he address consists of the text string <input> /<parameter>[/<option>]*
  - *<input> - number of the input for which data reading is to be made, setting options:*
    - *1 - input 1*
    - *2 - input 2*
  - *<parameter> - text string specifying the data to be read; setting options:*
    - *ADC - analogue signal measurement on an analogue input*
    - *DIN - logical value reading on analogue input; result values: 0 to 1*
    - *PULSE - digital input pulse counter reading*
  - *<option> - text string specifying the parameter options; setting options :*
    - *for ADC:*
      - *CAL - the data reading result is a calibrated value ranging from 0 to 100*
      - *ABS - he data reading result is a value obtained directly from the A/D converter ranging from 0 to 1023*
    - *for DIN - <option> is undefined and thus unavailable for this parameter*
    - *for PULSE:*
      - *ABS - the data reading result is the current pulse count read in the input pulse counter ranging from 0 to (2^64 - 1)*
      - *REL - the data reading result is the pulse count read after the last data reading (or pulse counter deletion/resetting) ranging from 0 to (2^64 - 1)*
- *< extra_parameter1 > - optional protocol parameter*
  - *specifies the protocol default baud rate*
  - *IEC 62056-21 - specifies the required default baud rate of the protocol; if the parameter is not included, the default value is used*
  - *M-Bus - no meaning (ignored)*
  - *A/D inputs - no meaning (ignored)*
- *< extra_parameter2 > - optional protocol parameter*
  - *specifies whether to deny or accept a higher baud rate offer for the "Protocol mode C"*
  - *IEC 62056-21 - logical value (if the parameter is not included, 0 is used)*
    - *0 - the protocol accepts the higher baud rate offer from the device for the "Protocol mode C"*
    - *1 - the protocol denies the higher baud rate offer from the device for the "Protocol mode C" and communication goes on at the default rate*

- *M-Bus - no meaning (ignored)*
- *A/D inputs - no meaning (ignored)*

Reply format:

```
^SCAMS: <error_code>,<device_identification>,<data>
OK
```

- *<error_code> - error code returned as a result of the data reading process (0 means no error) - number*
  - 32-bit signless decimal number
- *<device_identification> - device identification - text string<data> - raw data read from a device - text string*
  - 32-bit signless decimal number

— Examples of Immediately Readings —

Example 1 - suppose you want to read immediately and write out data for a device connected to port 2, communicating via IEC 62056-21 and having the address 947833. The command is as follows:

```
AT^SCAMS="DIRECT_READ",2,1,"947833"
```

Example 2 - same as above, but 1200 Bd is the default baud rate:

```
AT^SCAMS="DIRECT_READ",2,"IEC62056-21","947833",1200
```

Example 3 - same as above, but the higher baud rate offer for the "Protocol mode C" from the device is denied:

```
AT^SCAMS="DIRECT_READ",2,1,"947833",1200,1
```

Example 4 - suppose you want to read immediately and write out data for a device connected to port 2, communicating via M-Bus, having the primary address 128 and communicating at 2400 Bd:

```
AT^SCAMS="DIRECT_READ",2,"MBUS","128@2400"
```

Example 5 - same as above, but the baud rate is not specified and thus the default baud rate of 2400 Bd is used:

```
AT^SCAMS="DIRECT_READ",2,"MBUS","128"
```

Example 6 - same as above, but the protocol is given in a numerical format :

```
AT^SCAMS="DIRECT_READ",2,2,"128"
```

Example 7 - suppose you want to read immediately and write out data for a device connected to port 2, communicating via M-Bus, having the secondary address 01579411 and communicating at 2400 Bd:

```
AT^SCAMS="DIRECT_READ",2,"MBUS","*01579411@2400"
```

Example 8 - suppose you want to read immediately and write out data for a device connected to port 2, communicating via M-Bus, having the secondary address with two wildcard positions 015794FF and communicating at 2400 Bd :

```
AT^SCAMS="DIRECT_READ",2,"MBUS","*015794FF@2400"
```

Example 9 - suppose you want to read the immediate signal value for analogue input 1 via the A/D input port and calibrate and write out the result :

```
AT^SCAMS="DIRECT_READ",128,"AIN_DIN","1/ADC/CAL"
```

Example 10 - suppose you want to read the immediate signal value for analogue input 2 via the A/D input port, convert the result into logical interpretation of the signal level and write out the result :

```
AT^SCAMS="DIRECT_READ",128,"AIN_DIN","2/DIN"
```

Example 11 - suppose you want to read the immediate pulse counter value for digital input 1 via the A/D input port and write out the result :

```
AT^SCAMS="DIRECT_READ",128,"AIN_DIN","1/PULSE/ABS"
```

— End of Examples of Immediate Readings —

Automatic Meter Detection

The command is used for automatic detection of a device on a bus.

> ⚠️ **Caution**
>
> - When you enter this command, the SCDATA (record memory) and meter list will be deleted!

```
AT^SCAMSADD=,&lt;port&gt;,&lt;protocol&gt;,&lt;baudrate&gt;,&lt;
interval&gt;,&lt;count_of_expected_devices&gt;
```

The command detects a device on the bus, adds it to AMS device list, deletes data in the data space and writes out the result including the addresses of the meters detected.

- *<port> - number of the physical port to which the device is connected - number*
    - *2 - RS232/RS485/M-Bus communication port (depending on the HW version)*
    - *128 - internal A/D input module communication port (analogue/digital input module)*
- *<protocol> - name/number of the communication protocol used for device communication*
    - *for number:*
        - *1 - denotes the IEC 62056-21 protocol*
        - *2 - denotes the M-Bus protocol*
        - *128 - denotes the A/D input reading protocol (for reading data from the analogue/digital input module)*
    - *for text string:*
        - *"IEC62056-21" - denotes the IEC 62056-21 protocol*
        - *"MBUS" - denotes the M-Bus protocol*
        - *"AIN_DIN" - denotes the A/D input protocol (for reading data from the analogue/digital input module)*
- *<baudrate> - communication rate in Bauds; if the parameter is not included, the default value is used*
    - *setting options: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400; default value: 2400.*

- *<interval> - time interval symbol text string (minutes/hours/days); setting options:*
  - *"1M", "2M", "3M", "4M", "5M", "6M", "10M", "12M", "15M", "20M", "30M", "1H", "2H", "3H", "4H", "6H", "8H", "12H", "1D"*
- *<count_of_expected_devices> - number specifying the count of newly added devices*
  - *count of devices expected to be on the bus. If the count is met, automatic detection will be terminated.*

Reply format:

```
^SCAMSADD: <error_code> , <count_of_devices> , <primary_address1>
; <secondary_address1> ; <primary_address2> ; <secondary_address2> ;
..................... "OK
```

Examples of commands (3 meter detection on M-Bus, 2400 Bd 2 hour AMS reading interval) - one command written in different ways:

```
AT^SCAMSADD=2,2,"default","2h",3
```

```
AT^SCAMSADD=2,2,"2400","2h",3
```

```
AT^SCAMSADD=2,mbus,"default","2h",3
```

```
AT^SCAMSADD=2,mbus,"2400","2h",3
```

Examples of replies:

No error (detection of 3 meters): With error (a different meter count was detected: 2 meters in this case):

```
^SCAMSADD: 65286,2,"1,00660421;19,08780251"OK
```

With error (secondary address collision of the M-Bus slave devices):

```
^SCAMSADD: 131094,0,""OK
```

Reading of AMS Records

AT interface for AMS-type data record reading.

```
AT^SCDATA=?
```

The command writes out the list of commands for data record reading module control.

Reply:

```
^SCDATA: "MESS_CNT"^SCDATA: "GET_SPACE"^SCDATA: "GET_OLDEST"^SCDATA: "GE
T_FILTERED","0-150"^SCDATA: "DEL_OLDEST"^SCDATA: "DEL_ALL"OK
```

```
AT^SCDATA="MESS_CNT"
```

The data record reading module command returns the count of data records stored.

Reply format:

```
^SCDATA: "MESS_CNT",<count>OK
```

Example of reply (the data record memory contains 44 records):

```
^SCDATA: "MESS_CNT",44OK
```

The data record reading module command returns the memory size available for data record storing (in bytes).

Reply format:

```
^SCDATA: "GET_SPACE",<count>OK
```

Example of reply (the available free memory space is 7,008,089 B)

```
^SCDATA: "GET_SPACE",7008089OK
```

```
AT^SCDATA="GET_OLDEST",&lt;count&gt;
```

The data record reading module command returns the listing of the oldest data records .

- *<count> - specifies the count of the oldest records to be listed - number*

> ✅ **Tip**
>
> - If you enter AT^SCDATA="GET_OLDEST" without specifying the count of records, the last record will only be read.

Reply format for a known data format and meaning of the parameters in the row:

```
AT^SCDATA: <record_type>,<time_stamp>,<sequence_number>,<protocol>,
<index>,<error_code>,<device_identification>,<data>OK
```

- *<record_type> - data record type - text string*
  - *"AMS" - AMS (Autonomous Metering System) record*
  - *"WMBUS" - Wireless M-Bus record*
  - *"WMBUS2" - Wireless M-Bus record captured by the other module (if available)*
- *<time_stamp> - data storing timestamp - number*
  - *32-bit Unix timestamp*
- *<sequence_number> - record sequence number (incrementing with every new record) - number*
- *<protocol> - name of the communication protocol used for device communication - text string*
  - *"IEC62056-21", "MBUS", "AIN_DIN" and "Unknown" are available at present*
- *<index> - device index in the device list at the time of data recording - number*
- *<error_code> - error code returned as a result of the device data reading process (0 means no error) - number*
  - *32-bit decimal signless number*
- *<device_identification> - device identification - text string*
  - *for IEC 62056-21 - text identification string (as received from the device)*
  - *for M-Bus - empty text string (identification string does not exist)*
  - *for A/D input protocol - empty text string (identification string does not exist)*
- *<data> - raw data read from the device - text string*

- *for IEC 62056-21 - received data text string (as received from the device)*
- *for M-Bus - data framework text string received from the device in the ASCII HEX format (data byte is represented by two consecutive ASCII characters specifying the data byte value in the hexadecimal format)*
- *for A/D input protocol - result value text string in the decimal format*

Reply format:

```
^SCDATA:"WMBUS",1452142010,36617,-14,70,"44A20577140054013772771400544A20
50137E70000002F2F046D052707210167460DFD11106563616B696E756D6F6B656C65544
E320E138545030000002F2F2F2F2F2F2F2F"OK
```

Reply format for unknown data format:

```
AT^SCDATA: <record_type>,<time_stamp>,<sequence_number>,"Unsupported
record format"OK
```

Filtering of AMS Records

```
AT^SCDATA="GET_FILTERED",&lt;filter&gt;
```

The data record reading module command returns the listing of records according to the preset record filter.

- <filter> - specifies the filtering requirements for data record listing and is applied to all data records in the record memory.

Reply format for a known data format and meaning of the parameters in the row:

```
AT^SCDATA: <record_type>,<time_stamp>,<sequence_number>,<protocol>,
<index>,<error_code>,<device_identification>,<data>
OK
```

- *<record_type> - data record type - text string*
    - *"AMS" - AMS (Autonomous Metering System) record*
    - *"WMBUS" - Wireless M-Bus record*
    - *"WMBUS2" - Wireless M-Bus record captured by the other module (if available)*
- *<time_stamp> - data storing timestamp - number*
    - *32-bit Unix timestamp*

- *<sequence_number> - record sequence number (incrementing with every new record) - number*
- *<protocol> - name of the communication protocol used for device communication - text string*
  - *"IEC62056-21", "MBUS", "AIN_DIN" and "Unknown" are available at present*
- *< index> - device index in the device list at the time of data recording - number*
- *<error_code> - error code returned as a result of the device data reading process (0 means no error) - number*
  - *32-bit decimal signless number*
- *<device_identification> - device identification - text string*
  - *for IEC 62056-21 - text identification string (as received from the device)*
  - *for M-Bus - empty text string (identification string does not exist)*
  - *for A/D input protocol - empty text string (identification string does not exist)*
- *<data> - raw data read from the device - text string*
  - *for IEC 62056-21 - received data text string (as received from the device)*
  - *for M-Bus - data framework text string received from the device in the ASCII HEX format (data byte is represented by two consecutive ASCII characters specifying the data byte value in the hexadecimal format)*
  - *for A/D input protocol - result value text string in the decimal format*

Filter Settings

The general format for **<filter>** is "<condition>;<condition>;........;<condition>", where <condition> specifies the filtering condition. The maximum count of conditions in the <filter> array is 10. The data record listing is sent only if it meets all the <filter> conditions.

<condition> - the general format for <condition> is <parameter><session><value>

<parameter> - text string defining the condition parameter. The supported strings are as follows:

- *RECORD_TYPE* - data record type
- *TIME* - data record timestamp
- *TABLE_ID* - device index in the device list
- *ERROR_CODE* - data record error code

<session> - text string defining the required session between the condition parameter and condition value. The supported strings are as follows:

- == - is equal to
- != - is not equal to
- > - is greater than
- >= - is greater than or equal to
- < - is smaller than
- <= - is smaller than or equal to

<value> - numerical value/name text string

Examples of Use

The command writes out all the Wireless M-Bus data records contained in the data record memory.

```
AT^SCDATA="GET_FILTERED","RECORD_TYPE==WMBUS"
```

The command writes out all the AMS data records whose timestamps fall into the interval ranging from 388665085 to 1388668685 inclusive.

```
AT^SCDATA="GET_FILTERED","RECORD_TYPE==AMS;TIME>=1388665085;
TIME<=1388668685"
```

The command writes out all the AMS data records whose timestamps are greater than 1388665085 and whose device index in the AMS device list is 1.

```
AT^SCDATA="GET_FILTERED","RECORD_TYPE==AMS;TIME>1388665085;TABLE_ID==1"
```

The command writes out all the data records whose error code is non-zero .

```
AT^SCDATA="GET_FILTERED","ERROR_CODE!=0"
```

The command writes out all the data records whose error code is 0, whose record type is AMS and timestamp is greater than 1388668685.

```
AT^SCDATA="GET_FILTERED","ERROR_CODE==0;RECORD_TYPE==AMS;
TIME>1388668685"
```

Record Deleting

```
AT^SCDATA="DEL_OLDEST",&lt;count&gt;
```

The data record reading module command deletes the oldest data record .

- <count> - specifies the count of the oldest records to be deleted.

Example: remove the five oldest data records from the memory.

```
AT^SCDATA="DEL_OLDEST",5
```

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

```
AT^SCDATA="DEL_ALL"
```

The data record reading module command deletes all the data records .

Reply format - action succeeded:

```
OK
```

Reply format - action failed:

```
ERROR
```

AMS Error Codes in Data Reading Lists

General AMS Core Errors

0	No error (no error has occurred)
1	Error in the input of parametric reading data to protocol module
2	Error in the output of reading data from protocol module
3	Required protocol cannot be connected to the specified physical port
4	Data reading from device was not started

IEC 62056-21 Support Plug-In Module Errors	
65536	Non-specific error (module internal error)
65537	Data reading interrupted and terminated by user
65538	Wrong device address string format
65539	Non-supported baud rate character in device reply
65540	Physical port baud rate was not set
65541	Identification message timeout
65542	AMS-unsupported protocol mode is used in AMS
65543	Acknowledgement/option select message timeout
65544	Physical port baud rate was not set
65545	Two consecutive message character timeout
65546	Two consecutive message timeout

IEC 62056-21 Support Plug-In Module Errors	
65793	Invalid character in Manufacturer's identification in Identification message received
65794	Invalid character in Baud rate identification) Identification message received
65795	Invalid character in Identification in Identification message received
65796	Invalid character in CR and LF arrays in Identification message received
66049	Invalid character in Data block in Data message received
66050	Insufficient size of AMS internal received data buffer for the amount of data in Data block in Data message received
66051	Invalid character in CR and LF arrays in Data message received
66052	Invalid character in ETX in Data message received
66053	Data integrity error in Data message, BCC value does not match the calculated block control character

M-Bus Support Plug-In Module Errors

131072	Non-specific error (module internal error)
131073	Data reading interrupted and terminated by user
131074	Wrong device address (and baud rate) string format
131075	Physical port baud rate was not set
131076	"SND_NKE" communication framework was not sent to device within defined timeout
131077	"SND_NKE" communication framework was not sent to device within defined timeout
131078	Reply reception timeout from device to "SND_NKE" communication framework
131079	

M-Bus Support Plug-In Module Errors

	Wrong communication framework type in reply from device to "SND_NKE" communication framework
131080	"REQ_UD2" communication framework was not sent to device within defined timeout
131081	"REQ_UD2" communication framework was not sent to device within defined timeout
131082	Reply reception timeout from device to"REQ_UD2" communication framework
131083	Wrong communication framework type in reply from device to "REQ_UD2" communication framework
131084	Wrong reply type from device to "REQ_UD2" communication framework
131085	Wrong device address in reply from device to "REQ_UD2" communication framework
131329	Wrong first "start" character of reply framework from device
131330	Wrong second "start" character of reply framework from device
131331	Wrong "stop" character of reply framework from device
131332	Wrong checksum of reply framework from device
131333	Integrity error of data defining length of reply framework from device
131334	Invalid value of data defining length of reply framework from device

A/D Input Support Plug-In Module Errors

8388608	Non-specific error (module internal error)
8388609	Data reading interrupted and terminated by user
8388610	Wrong device address string format

A/D Input Support Plug-In Module Errors

8388611	Request communication framework sending timeout to A/D input module
8388612	Request communication framework sending timeout to A/D input module
8388613	Reply framework receiving timeout from A/D input module
8388865	Wrong reply framework type from A/D input module
8388866	Wrong reply framework sequence number from A/D input module
8388867	Insufficient size of AMS internal data buffer for the amount of A/D input module reply framework data
8388868	Wrong checksum of reply framework from A/D input module
8388869	Wrong synchronisation array of reply framework end from A/D input module

ONF Customer Module Errors (Pseudo-AMS Error Codes)

65280	Non-specific error (module internal error)
65281	User-defined expected count of M-Bus slave devices is greater than the maximum device count to be served by AMS
65282	AMS searching start error for M-Bus slave devices
65283	AMS searching result reading error for M-Bus slave devices
65284	AMS device list deleting error
65285	AMS searching result parsing error for M-Bus slave devices
65286	Count of slave devices found on M-Bus differs from user-defined expected count of M-Bus slave devices
65287	M-Bus slave device adding error to AMS device list

ONF Customer Module Errors (Pseudo-AMS Error Codes)

| 65288 | User output SMS count exceeds the maximum allowed SMS count |

```
AT^SCAMS="DIRECT_READ",128,"AIN_DIN","2/DIN"
```

3.9 Other Useful Functions

2N ® SmartCom PRO as Router

2N® SmartCom PRO can be used for access to the Internet. This function is only available in the versions equipped with an Ethernet connector. Set the main route to the Internet via GSM and connect the device to the network or machine to be connected to the Internet. Set 2N® SmartCom PRO as the default gateway and any available public DNS (the server mediated by Google at 8.8.8.8., e.g.) as the DNS server.

> ⊘ **Tip**
>
> - The function is only available in the versions equipped with an ETH connector and GSM or UMTS module.

> ⓘ **Note**
>
> - Make sure that a data connection supporting SIM card is inserted.
>
> - Set the default gateway routing manually. 2N® SmartCom PRO does not support automatic changes of default gateways (VRRP).

> ⚠ **Caution**
>
> - Check whether fees are to be paid for data transmissions.

2N® SmartCom PRO is used for low data flows as the GPRS technology does not provide a very high Internet connection rate. However, it can be a useful backup for power outage.

IP Address Access Authorisation

These commands helps you define 5 IP addresses authorised for connection to your terminal. All incoming calls from undefined addresses will be ignored.

```
at^sccfg=&quot;auth_ip&quot;,1
```

- The command enables/disables IP address access authorisation.

```
at^scipauth=&quot;add&quot;,&quot;54.182.153.21&quot;
```

- The command adds an IP address to the end of the list. When the maximum count of IP addresses is reached, the command will be rejected.

```
at^scipauth=&quot;remove&quot;,1-5
```

- The command deletes an IP address from the list. Select 1 to 5 to define the address position.

```
at^scipauth=&quot;clear&quot;
```

- The command deletes all the addresses listed.

```
at^scipauth?
```

- The command writes out the IP address list.

```
SCIPAUTH: 1> "90.182.112.54"<br> SCIPAUTH: 2> "90.182.112.5"
```

```
at^scipauth=?
```

- The command displays the interface setting options.

```
SCIPAUTH: "ADD","15"<br> SCIPAUTH: "REMOVE",1-5<br> SCIPAUTH: "CLEAR"
```

Configuration Port Control

Communication via port RS-232 is used for configuration. Switch the port to the command mode to start configuring and back to the transparent mode when you have finished to make the port resend the right data. Use the commands below to control the port, set the default modes, reject incoming connections and use authorisation.

```
+++
```

- The command switches the port into the command mode.

```
ato
```

- The command switches the port into the transparent mode.

```
ata
```

- The command switches the port into the transparent mode whenever an incoming TCP connection is ringing.

```
ath
```

- The command terminates (rejects) an incoming TCP ringing.

```
ats0?
```

- The command displays the automatic answering setting state for incoming calls.
    - 0 – means that incoming calls are ignored and answered manually via the ata command.
    - Digit – count of rings after which the incoming connection is answered automatically.

```
ats0=5
```

- The command sets the count of rings before automatic answer to 5. After the 5th incoming ring, the port is switched to the transparent mode automatically.

```
at^sc232at=1
```

- The command sets the AT mode upon power up.
    - 0 - off - the port is in the DATA mode upon power up.
    - 1 - on - the port is in the CMD mode upon power up.

```
at^sc232at?
```

- The command displays the automatic AT mode switching setting state.

```
at^sc232at=?
```

- The command displays the automatic AT mode switching setting options.

```
at^sc232lock=0-1
```

- The command sets RS 232 authorisation, which means that a password is required for connection via this port: 0 = enabled, 1 = disabled.

```
at^sc232lock?
```

- The command displays the current RS 232 lock setting.

```
at^sc232lock=?
```

- The command displays the RS 232 lock setting options.

4. Configuration

In this section we describe the 2N® SmartCom PRO configuration using AT commands entered into the command line.

Here is what you can find in this section:

- 4.1 Configuration by Terminal
- 4.2 Configuration by SMS
- 4.3 Configuration Program

4.1 Configuration by Terminal

The basic **2N**® **SmartCom PRO** configuration and firmware download take place during the manufacturing process. In general, no additional configuration is necessary except for GPRS connection to the Internet on the GSM module and the port baud rates, which may be different depending on the providers and types of peripherals.

You can configure **2N**® **SmartCom PRO** in one of the following two ways.

- Via the RS 232 serial interface; or
- Via the IP interface.

Configuration via RS 232

The RS 232 port is always in the data mode upon power up, which means that it retransmits all captured events to GPRS port 10001. Therefore, to configure **2N**® **SmartCom PRO** via the serial port, switch the port into the command mode first and then get connected to it using your PC serial port. The baud rate is 115,200 bps by default. Now enter the following sequence of characters, preceded and followed by a 1–second delay, into the terminal:

```
+++
```

When **2N**® **SmartCom PRO** answers OK, you can start configuring the device using the AT commands listed in **Sub. 6**.

Restart the RS–232 port when you have finished configuring to return into the data mode. Send the following command:

```
at^scport1="restart"
```

Configuration via IP

If you know the 2N® SmartCom PRO public IP address, you can configure the device remotely too using the IP protocol. If not, connect to 2N® SmartCom PRO using RS 232 as described above and send one of the following commands (for GPRS /Ethernet):

```
at^sccfg="local_ip"<br>
at^sceth="local_ip"
```

2N® SmartCom PRO will send you the IP address as negotiated with the provider or assigned by the DHCP server in the Ethernet case. If a fixed public IP address is activated on the SIM card, the SIM will always use this IP address. If a dynamic public IP address is used, check the IP address whenever your restart the device.

> ⚠️ **Caution**
>
> - **Make sure that the IP address assigned to the SIM card is public.** It is because you will not be able to establish remote connection to 2N® SmartCom PRO in the STANDALONE mode if an IP address from the provider's internal range is used. The 2N® SmartCom PRO address need not be public if you use the TCPCLIENT server connection.

If you know the public IP address, launch the Putty and get connected via port 10000, which, together with the serial port, can also be used for configuring. Port 10000 is used for configuration, input data reading and output control in the IP domain. It is in the command mode upon power up and thus need neither be switched nor restarted after configuration.

> ⓘ **Note**
>
> - This port requires the safety password if configured so.

> ⚠️ **Caution**
>
> - If your **2N® SmartCom PRO** fails to respond to incoming TCP connections, check the status of the port to which you are connecting. If the port is in the command mode, notification on incoming connection is sent to the line (RING is displayed). Hence, enter the 'ata' command for answer or set automatic answering by parameter ats0. Refer to the **General Commands** subsection in the **List of supported AT commands** below.

4.2 Configuration by SMS

The SMS messages described in this section are used for basic settings of the terminal.

You have to know the telephone number of the SIM card inserted in your **2N®** **SmartCom PRO** to ensure a reliable function.

> ⚠ **Caution**
>
> - Every SMS sent to **2N® SmartCom PRO** must meet all of the conditions mentioned below. Otherwise, the SMS will be ignored, no changes will be made and an error message will be sent to the SMS-sending number.

Function Desription

You can configure **2N® SmartCom PRO** by sending short text messages to the telephone number of the SIM card inserted in the terminal. Doing this, respect the below listed parameters to avoid an SMS processing error.

> ⊙ **Every configuration SMS must meet the following conditions:**
>
> - Name of device – 'SC'
> - Correctly formatted command (INFO, CNF, etc.)
> - Authorisation password
> - Correctly defined parameters (refer to the command parameters)
> - Upper/Lower Case must be respected
> - Parameters must be separated with a space
> - 7-bit SMS format (no diacritic symbols)
> - Maximum length of 160 characters (which is one SMS, concatenated SMS are not supported)

> **ⓘ Note**
>
> - **The maximum SMS length is 160 characters.** This is a GSM standard, which allows 1120 bits, i.e. 140 bytes, per SMS. Since 7-bit encoding is used by default, 20 characters are saved per 140 bytes and so the resultant length is 160 characters. The GSM 03.38 character set is applied. UCS-2 16-bit encoding is used for the characters that are not included in the ASCII standard (characters with diacritics). In that case, the SMS contains only 70 characters. **Not supporting the UCS-2 character set, 2N® SmartCom cannot process such SMS correctly and marks them as errors.**
>
> - Modern GSM devices support SMS concatenating where several standard SMS are merged into a long one. Information on each fragment of the long SMS is written into the User Data Header (UDH) and so no fragment may be longer than 153 chars in 7-bit encoding. **Not supporting the UDH encoding system, 2N® SmartCom cannot process such SMS correctly and marks them as errors.**

INFO SMS

Send the following message to identify the provider, IMSI, IMEI or serial number of your 2N® SmartCom PRO :

```
SC INFO PWD=&lt;password&gt;
```

> ⚠️ **Caution**
>
> - Remember to enter the correct password into the SMS to avoid SMS ignoring as a SPAM.

The reply to such SMS can, for example, look as follows:

```
SIG=-71dBm OPER=T-Mobile CZ IMSI=230015001155344 IMEI=355915030750962
SN=50-0369-0080 LIP=89.24.0.141
FW=1.2.0.2.4 STAT=CONNECTED
```

> ⚠️ **Caution**
>
> - The reply format depends on the current terminal mode.

The reply for the TCP and UDP client modes is as follows:

```
SIG=&lt;signal_level&gt; OPER=&lt;operator&gt; IMSI=&lt;
sim_card_imsi&gt;IMEI=&lt;module_imei&gt;
SN=&lt;terminal_serial_number&gt; LIP=&lt;local_ip&gt;FW=&lt;
terminal_fw_version&gt;
STAT=&lt;current_terminal_status&gt;
```

The reply for the STANDALONE mode is as follows:

```
SIG=&lt;signal_level&gt; OPER=&lt;operator&gt; IMSI=&lt;
sim_card_imsi&gt;&lt;br&gt;IMEI=&lt;module_imei&gt;
SN=&lt;terminal_serial_number&gt; LIP=&lt;local_ip&gt;FW=&lt;
terminal_fw_version&gt;
STAT=LISTENING
```

The meanings of the parameters in the reply SMS are as follows:

- **<password>** – authentication password; the password has to be set up first (AUTH_PASS configuration parameter).
- **<signal_level>** – current signal level in the '–XXXdBm' format.
- **<operator>** – provider's name in the long format (name, no number).
- **<sim_card_imsi>** – IMSI of the SIM card inserted.
- **<module_imei>** – IMEI of the GSM module used.
- **<terminal_serial-number>** – serial number in the 'xx–xxxx–xxxx' format.
- **<local_ip>** – local address of the terminal (assigned by the provider).
- **<terminal_fw_version>** – terminal firmware version.
- **<current_terminal_status>** – current terminal status (in the Client mode only).

    - **CONNECTING** – terminal connection to 2N® SmartCom Server in progress.

    - **CONNECTED** – terminal connection to 2N® SmartCom Server completed.

    - **LISTENING** - terminal is in the STANDALONE mode.
    - **DATA-OFF** - terminal is in the DATA_OFF mode.

Basic Settings

The following configuration SMS sets the basic parameters necessary for a proper function of the terminal, including connection to the GSM provider and 2N® SmartCom Server, 2N® SmartCom PRO mode switching and RS 232 authorisation enable.

Use the following command to know the current parameters settings:

```
SC CNF PWD=&lt;password&gt;
```

The reply to such SMS can, for example, look as follows:

```
MODE=1 232LCK=0 APN=publicip.t-mobile.cz DUSR= DPWD= IPS=90.182.112.51
PORTS=1620
```

> ⊘ **Tip**
>
> - You can send all the parameters at once or enter only those you want to change in the configuration. Hence, you need not always send all optional parameters but have to meet the basic conditions mentioned in the Function Description above.

Now let us describe the contents of a configuration SMS and all optional parameters.

```
SC CNF PWD=&lt;password&gt; [MODE=&lt;mode&gt;] [232LCK=&lt;
rs232_lock&gt;] [APN=&lt;gprs_apn&gt;] [DUSR=&lt;gprs_data_user&gt;]
[DPWD=&lt;gprs_data_password&gt;] [IPS=&lt;server_ip_address&gt;]
[PORTS=&lt;server_port&gt;] [CHPRIM=&lt;primary_channel&gt;] [NPWD=&lt;
new_password&gt;]
```

The parameters in square brackets are optional and can be omitted. The password is the only mandatory parameter! The meanings of the CNF command parameters are as follows:

- **PWD=<password>** – authentication password. The password has to be set up first (AUTH_PASS configuration parameter). The password is the only **mandatory** parameter in this configuration SMS!

- **MODE=<mode>** – this parameter changes the mode and has the following numerical options:
  - **1** – TCP Client
  - **2** – UDP Client
  - **3** – STANDALONE
- **232LCK=<rs232_lock>** – this parameter enforces authorisation on the RS232 port. If enabled, you have to authorise yourself by the password to get connected via this port. 0 = disabled, 1 = enabled.
- **APN=<gprs_apn>** – provider's Access Point Name.
- **DUSR=<gprs_data_user>** – user name for provider connection.
- **DPWD=<gprs_data_password>** – password for provider connection.

- **IPS=<server_ip_address>** – 2N® SmartCom Server IP address (used in the Client mode).
- **PORTS=<server_port>** – listening server port for terminal connection (used in the Client mode).
- **CHPRIM =<primary_channel>** – preferred data channel entered as a text:
  - **ETH**
  - **GSM**
- **NPWD=<new_password>** – new authorisation password.

> ✓ **Tip**
>
> - The new authorisation password should contain 5 characters at least to prevent communication problems with other applications.

> ✓ **Tip**
>
> - If the SMS has been created and processed correctly, 2N® **SmartCom PRO** will send a confirmation SMS to the number from which the original SMS was sent. The confirmation SMS contains all the recently set parameters and is used for verification of the set values.

> ⚠️ **Caution**
>
> - Some GSM providers do not require user name and password for Internet connection setup. Therefore, contact your provider to make sure.

Example of configuration SMS:

Suppose you want to change the default password (into password 1) and set APN to 'internet.open'. Also, set the server IP address to 90.89.126.3 and the listening port to 1658. This is the most frequent way of setting.

Thus, send the following configuration SMS:

```
SC CNF PWD=APNpassword=internet.open IPS=90.89.126.3 PORTS=1658
NPWD=password1
```

The following reply will come from the terminal after processing:

```
MODE=1 232LCK=0 APN=internet.open DUSR= DPWD= IPS=90.89.126.3 PORTS=1658
```

Error Buffer

An error buffer is implemented in each terminal to store data on terminal operation errors. To display the buffer contents use either the terminal (an AT command) or an SMS. The SMS displays the last 10 errors starting with the most recent one.

```
SC ERR PWD=&lt;password&gt; [CLR]
```

- **CLR** – clear the error buffer.

> ⊘ **Tip**
>
> - This command includes one optional parameter – **CLR.** If you add this string after the password in the command SMS, you will receive an SMS showing the current buffer contents and all errors will then be deleted from the terminal memory.

The following reply will come from the terminal after processing:

```
1:8,10 2:3,10 1:1,0 7:8,10 6:8,7 5:7,40 4:11,848
```

The following reply will come if there is no error in the error buffer:

```
NO ERROR
```

> ⓘ **Note**
>
> - Refer to SC_errors.doc for the meanings of the error codes!

Relay Control

Every terminal is equipped with two relay outputs, which can be SMS controlled. The configuration SMS must include the password and one of the four programmable parameters at least.

The programmable parameters include:

- R1 or R2 – define the relay state:
  - 1 – relay is closed (on)
  - 0 – relay is open (off)
- S1 or S2 – define the relay state after the terminal restart:
  - 1 – relay is closed after restart
  - 0 – relay is open after restart
  - 2 – relay retains the pre-restart state

> ⊘ **Tip**
>
> - The relay gets off (contacts get released) after power off and, normally, the terminal does not remember the relay states. Therefore, the S1 and S2 parameters have been introduced, defining to which state the relay should be switched after power up.

Example:

Suppose you want relay 1 to be switched into the open state now and into the closed state after restart, and relay 2 to close and retain so after restart. Hence, send the following configuration SMS:

```
SC REL PWD=Password R1=0 R2=1 S1=1 S2=2
```

The command will be executed and the terminal will send the following reply:

```
R1=0 R2=1 S1=1 S2=2
```

IP Access

You can also set IP address access authorisation using SMS. By enabling the IP Access you will restrict the access to the terminal configuration to selected IP addresses. You can select up to 5 different IP addresses.

The configuration SMS is as follows:

```
SC IPA PWD=&lt;Password&gt; [AUTH=&lt;ip_auth&gt;] [ADD=&lt;
ip_addr&gt;] [DEL=&lt;ip_id&gt;] _[REP=&lt;ip_addr_1&gt;,&lt;
ip_addr_2&gt;,...&lt;ip_addr_5&gt;]
```

Now let us describe the contents of the configuration SMS and all optional parameters.

The parameters in square brackets are optional and can be omitted. The password is the only mandatory parameter! The meanings of the IPA command parameters are as follows:

- **PWD=<password>** – authentication password. The password has to be set up first (AUTH_PASS configuration parameter). The password is the only **mandatory** parameter in this configuration SMS!
- **AUTH=<ip_auth>** – this parameter changes the mode and has the following numerical options:
    - **0** – disabled
    - **1** – enabled
- **ADD=<ip_addr>** – add a selected IP address to the list end.

> ⚠ **Caution**
>
> - Enter the IP address in the numerical format only ranging from 0.0.0.0 to 255.255.255.255.

- **DEL=<ip_id>** – delete the IP address on the selected list position.

> ⚠ **Caution**
>
> - Select **0** in the ip_id parameter to delete all the IP addresses from the list. The SMS will be as follows: **SC IPA PWD=password DEL=0**.

- **REP=<ip_addr_1>, <ip_addr_2>,...<ip_addr_5>** – replace the existing address list with a new one.

Example:

Suppose you want to know whether the IP access function is enabled. Send the following SMS:

```
SC IPA PWD=&lt;password&gt;
```

The reply can, for example, look as follows:

```
AUTH=1 1:90.156.29.195 2:109.54.2.234
```

Now disable the IP access and delete all IP addresses from the list using the following SMS:

```
SC IPA PWD= AUTHpassword=0 DEL=0
```

The following reply will come from the terminal:

```
AUTH=0 NO IP
```

> **Tip**
>
> - **NO IP** means that there is no IP address in the address list.

AMS Settings

Use the SMS below to detect the M-Bus meters, assign them to the AMS list and read the current meter list.

```
SC AMS PWD=&lt;password&gt;
```

The command checks the meters installed.

- **PWD=<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!

The following reply comes from the terminal after processing:

```
&lt;session&gt;:&lt;message_id&gt;&gt; QTY=n address#1:protocol:port:
interval address#2:protocol:port:interval,..., address#N:protocol:port:
interval
```

Meanings of the SMS parameters:

- <session> - communication thread id: 1 - 100
- <message_id> - message id in the selected communication thread
- QTY - count of devices in the meter list
- address - primary_address
- *address - *secondary_address
- see below for more parameters

```
SC AMSADD PWD=&lt;password&gt; QTY=&lt;n&gt; PORT=&lt;port&gt;
PROTOCOL=&lt;protocol&gt; INT=&lt;interval&gt;
```

The command performs installation of meters.

> ⚠️ **Caution**
>
> - When you enter this command, the SCDATA (record memory) and meter list will be deleted!

Configuration SMS parameters:

- **PWD=<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!
- **QTY=<n>** – count of devices detected - number
    - This parameter gives the count of the devices that should be connected to the bus and thus detected. This count must show a match, otherwise an error will be reported (65286).
- **PORT=<port>** – number of the physical port to which the device is connected - number
    - 2 - RS232/RS485/M-BUS communication port (depending on HW version)
    - 128 - internal A/D input module communication port (analogue/digital input module)
- **PROTOCOL=<protocol>** – name/number of the communication protocol used for device communication
    - for number:
        - 1 - denotes the IEC 62056-21 protocol
        - 2 - denotes the M-Bus protocol
        - 128 - A/D input protocol (analogue/digital input module data reading protocol)
    - for text string:
        - "IEC62056-21" - denotes the IEC 62056-21 protocol
        - "MBUS" - denotes the M-Bus protocol
        - "AIN_DIN" - A/D input protocol (analogue/digital input module data reading protocol)
- **INT=<interval>** – text string of the time interval symbol (minutes/hours/days); setting options:
    - "1M", "2M", "3M", "4M", "5M", "6M", "10M", "12M", "15M", "20M", "30M", "1H", "2H", "3H", "4H", "6H", "8H", "12H", "1D"

The terminal sends the following reply:

```
&lt;session&gt;:&lt;message_id&gt;&gt; AMSADD started
```

Message parameters:

- <session> - communication thread id: 1 - 100
- <message_id> - message id in communication thread 1

The following reply comes from the terminal after processing:

```
&lt;session&gt;:&lt;id_zpravy&gt;&gt; QTY=n ERR=error_code address#1,
address#2,..., address#N
```

Message parameters:

- <session> - communication thread id: 1 - 100
- <message_id> - message id in communication thread 2
- QTY - count of devices detected
- ERR - error code (refer to ONF Customer Module Errors in Subs. 3.8 for details)
- address - primary_address
- *address - *secondary_address

Restart

Use SMS to perform terminal system restart or GSM module hard restart. After this SMS is sent and received, you will get a reply and the command will be executed. System restart means complete terminal restart like in the event of power supply disconnection. To reset the GSM module only without affecting other peripheral functions, use the GSM hard reset command.

The system restarting SMS is as follows:

```
SC SRST PWD=&lt;password&gt;
```

- **PWD=<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!

The terminal sends the following reply:

```
System Reset Initialized
```

The GSM hard reset SMS is as follows:

```
SC GSMHRST PWD=&lt;password&gt;
```

- **PWD=\<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!

The terminal sends the following reply :

```
GSM Hard Reset Initialized
```

Factory Reset

Use SMS to initiate the terminal factory reset. After this SMS is sent, you will get a reply and the command will be executed..

The SMS is as follows:

```
SC FRES PWD=&lt;password&gt;
```

- **PWD=\<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!

The terminal sends the following reply :

```
&lt;session&gt;:&lt;message_id&gt;&gt; Factory Reset Initialized
```

Message parameters:

- \<session> - communication thread id: 1 - 100
- \<message_id> - message id in the selected communication thread

Error Replies

Having noticed an error in the configuration SMS or command processing, 2N$^®$ SmartCom PRO will send you an error reply. **If the terminal generates an error reply, it means that the command was not executed!**

> ⊘ **Tip**
>
> - **2N$^®$ SmartCom PRO** sends error replies until all the configuration SMS parameters are OK and the command is executed.

The following errors can occur:

- **ERR Unknown command** – an unknown command has been entered. Four commands are allowed (INFO, CNF, ERR, REL, IPA).
- **ERR Password missing** – the configuration SMS fails to include the password, i.e. the PWD='password' string.
- **ERR Password incorrect** – the password is wrong.
- **ERR Too many parameters** – the configuration SMS contains more than 160 characters or was sent as a 16-bit message (with diacritic symbols and/or substandard characters).
- **ERR Unknown parameter** – an unknown parameter has been entered. Refer to the commands for all allowed parameters.
- **ERR Parameter value incorrect** – the parameter value is incorrect.

4.3 Configuration Program

The Terminal Config application is used for basic configuration of the terminal. This subsection provides details on its environment and basic functions. Launch the application from the directory into which you installed it.

> ⓘ **Note**
>
> - The 2N® SmartCom Terminal Configuration Program installation requires 20 MB of free disk space at least.

Details on Application
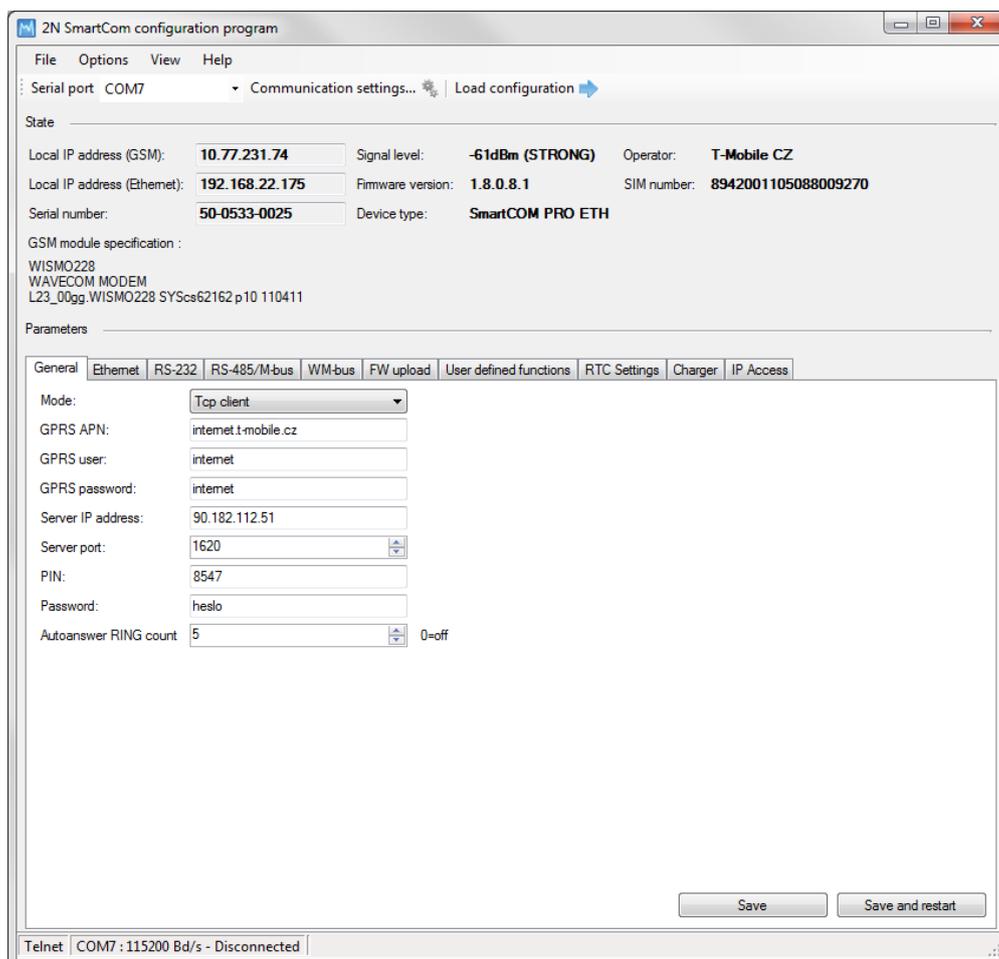
Figure: Terminal Config Window

The figure above shows the **2N® SmartCom** Terminal Configuration Program (hereinafter referred to as Terminal Config) window. The main toolbar helps you select the **Serial port** for connection to **2N® SmartCom PRO** (hereinafter referred to as terminal). This list includes all available serial ports installed in the PC on which the application is running.
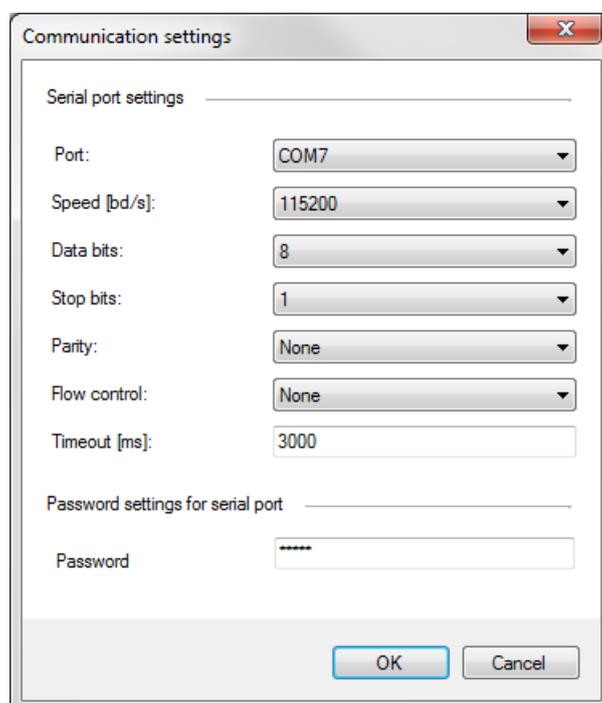
Figure: Communication Settings

Having selected a port, set the communication parameters to adjust communication rates and signal transmission parameters for the serial port. The **2N® SmartCom PRO** factory values are set here by default. Refer to the **Communication Settings** figure above.

No password is requested by default for the RS 232 port and so nothing has to be filled in. Passwords may be different for different terminals. If a password is requested, enter the password into the dialogue window displayed during the connecting process. The last main toolbar button helps you **Load configuration** from the terminal connected - the application gets connected and reads out all relevant data.

The **State** section displays the most important information on the given device.

The IP address, serial number, firmware version, logged-in provider and signal level are displayed. The SIM ICCID (Integrated Circuit Card ID) is displayed, which is not the SIM telephone number, i.e. MSISDN, but a unique SIM serial number. There are no two identical ICCIDs at one moment. These 19-digit numbers start with 8942 for the Czech Republic.

Moreover, the section provides information on the GSM module: type, manufacturer and firmware version.

The **Parameters** section helps you view and set parameters for the **2N$^®$ SmartCom PRO** terminals.

Use the **File** menu to read configuration, import/export configuration or reset the connected terminal. Via the **Options** menu you can adjust communication parameters and/or change the default language (CZ or EN). The **View** section helps you disable the tabs that are not frequently used and occupy too much space. Open the View menu and specify the tab(s) to be seen.
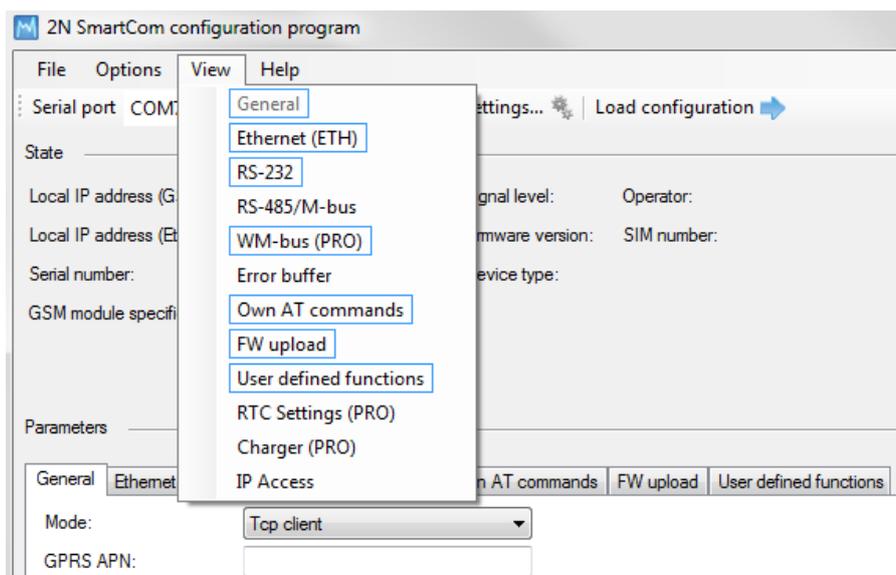
View menu Go to **Help** to know details on the Terminal Config manufacturer and version.

Use of Applicatione

> ✅ **Tip**
>
> Not all tabs are accessible to all device models.
>
> - **General, RS 232, RS 485 / M-Bus, Error buffer, Own AT commands, FW upload, User defined functions** and **IP access** are accessible to all terminals.
> - **WM-Bus, RTC settings** and **Charger** are available to the **PRO** and **PRO ETH** terminals only.
> - **Ethernet** is accessible to the **PRO ETH** terminals only.

Having read configuration, go to the **General** tab of the **Parameters** section.

Mode:	Tcp client
GPRS APN:	internet.t-mobile.cz
GPRS user:	internet
GPRS password:	internet
Server IP address:	90.182.112.54
Server port:	1620
PIN:	8547
Password:	heslo
Autoanswer RING count	5 0=off

Figure: General Menu

Here you can set all parameters comfortably as you can with the AT commands described in **S. 4** , i.e. GSM commands plus automatic answer parameters. Push **Save** to save the values. Push **Save and restart** to apply the changes immediately.

The Ethernet menu is only available to terminals equipped with an Ethernet port. Set the primary data flow channel, LED signalling functions, obtaining IP address from the DHCP server or static obtaining and DNS server use.

Ethernet menu

Use the RS 232 and RS 485 / M-Bus tabs to set communication parameters for each port. Set the password request enable and AT mode upon power up for RS 232. Use WM-Bus to set the operating and decryption modes for the module.

Click on the required item in the **List of configured devices** to select a device and push the buttons to the right to remove the item or add the decryption key.

> ⚠ **Caution**
>
> - The keys are added to the WM-Bus module directly for security reasons and are not displayed in the configuration. Therefore, be careful while configuring the keys.
> - A key adding error may lead to a decryption error.

Furthermore, you can **Add new device** to the list. Enter the device code and serial number. Complete the timeout, e.g., in case the selected device sends multiple identical pieces of information in a short time interval. The remaining parameters are optional and need not be specified.

> ⓘ **Warning**
>
> - Contact the device manufacturer for precise identification details.
> - Enter all the identification parameters of the new device to enable encryption in the device.

Display received messages in the **Get message** section. The messages are read from the database starting with the oldest one. The Message count parameter displays the count of received messages in the terminal and the amount of free space. When there is no free space, the oldest record will be deleted to make space for a new message. Push the **Export messages** button to export all the messages stored in the terminal into a PC file in order to make space for new messages.

The **Error buffer** helps you display the error codes captured during the operation. If no error has occurred since the last clearing date, ^SCERR: NO ERROR will be displayed. If an error occured, the error code will be displayed. Click on the Error code table button to display the message decryption window.
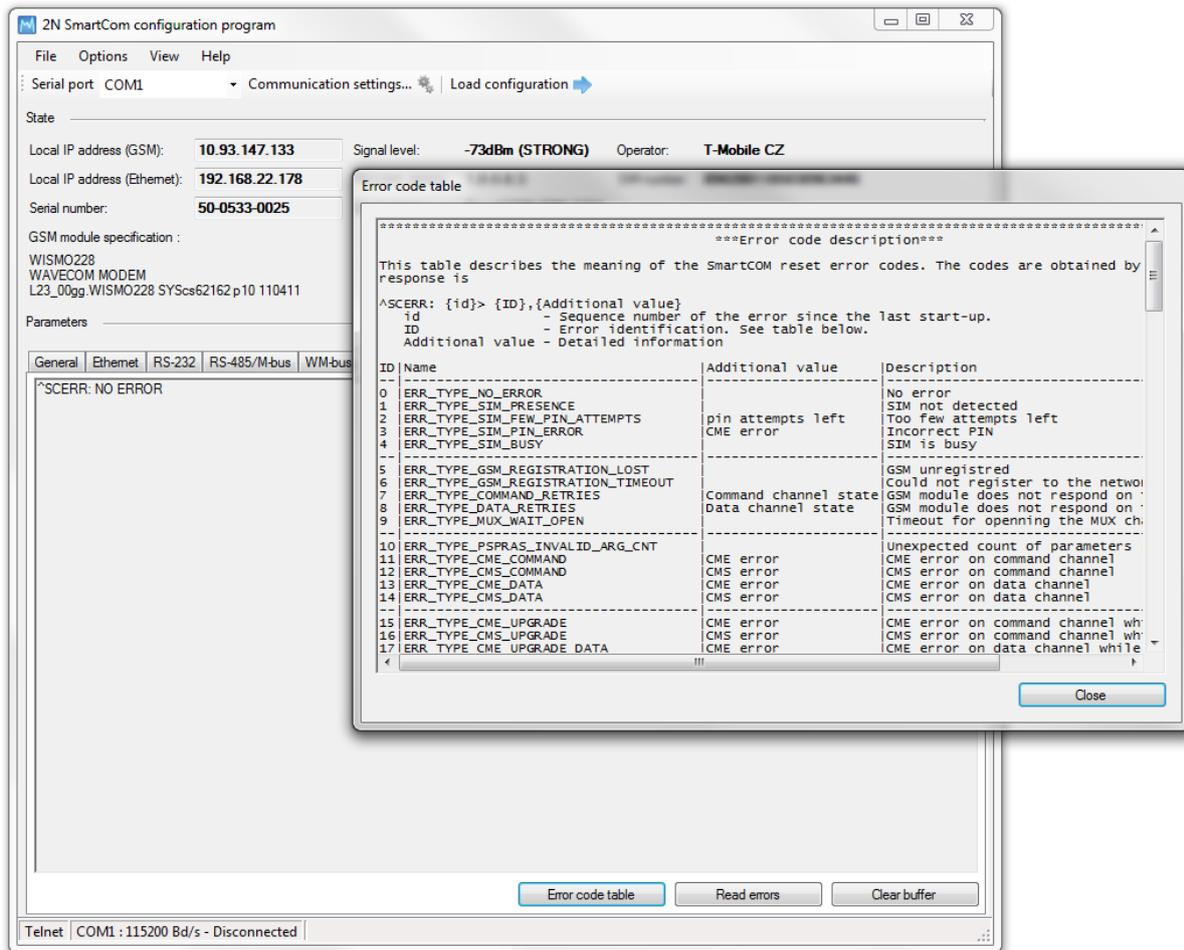
Figure: Error buffer

Use the **Own AT commands** tab to enter commands of your own for parameter setting. The tab behaves like a terminal. Enter a string into the AT command field corresponding to the supported commands listed in **Subs. 4.2** . Click on **Send** to send the command to the connected **2N® SmartCom PRO** unit and get the reply in the field below the command. A list of 20 last-applied commands is displayed to the right.

Figure: User Defined AT commands

The last tab helps you upgrade firmware via the terminal COM port. Select the path to the current firmware file and press **Start** to initiate the upgrading process, which is signalled by the progress bar below. When the process is completed, the terminal will get restarted.

Figure: FW Upload

> ⚠ **Caution**
>
> - Make sure that the terminal FW supports FW upgrade via the COM port to make the process successful. The minimum terminal FW version must be 1.4.0.

The **User defined functions** tab helps you set the UDF graphically and more easily in the application than using an AT command in the terminal. Just select the required functions and set their parameters and the application itself will generate the relevant AT command to be sent to the terminal, or to be stored for later use. Refer to **Console** for the current terminal communication listing and to **Current settings** for the list of configured commands in the terminal.

User Defined Functions Menu

Refer to **S. 3.5** for more details on the UDF setting options.

The **RTC settings** tab displays the current date, time and time zone data. You can also synchronise the device time with your PC time, or set the device time manually. Click on **Save** to confirm the settings.

Current date on device: **11.2.2013**   Current time on device: **12:20:35**   Current time zone on device: **UTC+02:00**

○ Time synchronization with your PC   **18.2.2013   16:06:27   UTC+01:00**

○ Manual time setting   Date: Time zone:

RTC Settings Menu

Charger is another setting option. It displays the back-up battery charging state for the **PRO** terminals. You can enable/disable the function and set the maximum charging voltage and current.

Charger Menu

IP access is the last accessible tab. You can define from which IP addresses it will be possible to access the terminal. Attempts from non-listed IP addresses will be ignored. Press **ON** to enable the function and **Add** to add an IPv4 address to the list. Click on an item for selection and push **Remove** to delete an address from the list.

IP Access Menu

5. SW Applications

In this section, administration and use of the **2N**® **SmartCom PRO** product using the **2N**® **SmartCom Server** and **2N**® **SmartCom Control Panel** are described.

Here is what you can find in this section:

- 5.1 SC Server
- 5.2 Control Panel
- 5.3 Control Panel Use
- 5.4 AT API

5.1 SC Server

2N® SmartCom Server is an application necessary for the 2N® SmartCom Control Panel operation, providing communication between the 2N® SmartCom Control Panel and the terminal.

> ⚠ **Caution**
>
> - The 2N® SmartCom Server must be installed on a PC accessible from a public Internet. Or, a NAT rule has to be defined to provide access to such PC. Otherwise, the terminals cannot be connected to the 2N® SmartCom Server. You can select the ports to be forwarded to your LAN via the NAT during installation.

Installation

Once launched, the installation program will scan your PC. If it finds another 2N® SmartCom Server version, it will ask you whether to uninstall. If you select No, the wizard will be terminated. If you select Yes, the current 2N® SmartCom Server version will be removed. Then you will be asked whether or not to remove the configuration files. If you select No, the current configuration will be retained and the application will only be updated.

Now the 2N® SmartCom Server Setup Wizard is launched. Follow the wizard instructions. Select the directory for the 2N® SmartCom Server to be installed. By default, C:\Program Files\2N TELEKOMUNIKACE\2N SmartCOM\SC Server is used.

Also select the port for the application to communicate with the terminals and other 2N® SmartCom Control Panel applications. Make sure that the port is accessible from the Internet (see above) and properly configured in all terminals and applications. Http accessibility is necessary as the selected port provides a listing of LOG files and can be used for firmware downloading to terminals. Firmware is downloaded whenever upgrade is initiated by the 2N® SmartCom Control Panel or the address is entered into the terminal.

Here you can also set the port for the AT API; refer to **Sub. 5.4** for details. Now the wizard is ready to install the **2N**® **SmartCom Server**. It displays the list of settings. Check the set parameters carefully as the values cannot be changed without reinstallation when the installation has been completed! If you find an error, push the **Back** button to return to the respective window. If all the settings are OK, push **Install** to continue. The wizard now installs the **2N**® **SmartCom Server**. The **2N**® **SmartCom Server** service is launched automatically after installation. If the automatic launch fails for any reason, launch the service manually. Now the **2N**® **SmartCom Server** is ready for use.

ⓘ **Note**

- The **2N**® **SmartCom Server** requires **500 MB** of free disk space at least for installation.

- Use the **2N**® **SmartCom Control Panel** application to make changes in the **2N**® **SmartCom Server** settings.

Licensing

All the **2N**® **SmartCom Server** functions are subject to licence. For your first installation of the application onto your PC, you will receive the trial licence for 800 h operation and connection of 2 users and 2 terminals.

> ⚠️ **Caution**
>
> - The clock counter is increased by 1 by every **2N**® **SmartCom Server** restart.

The **2N**® **SmartCom Licence Tool** is installed together with the server. Having launched the application, you can see the version, ID and details on service settings for the server installed. See Figure.
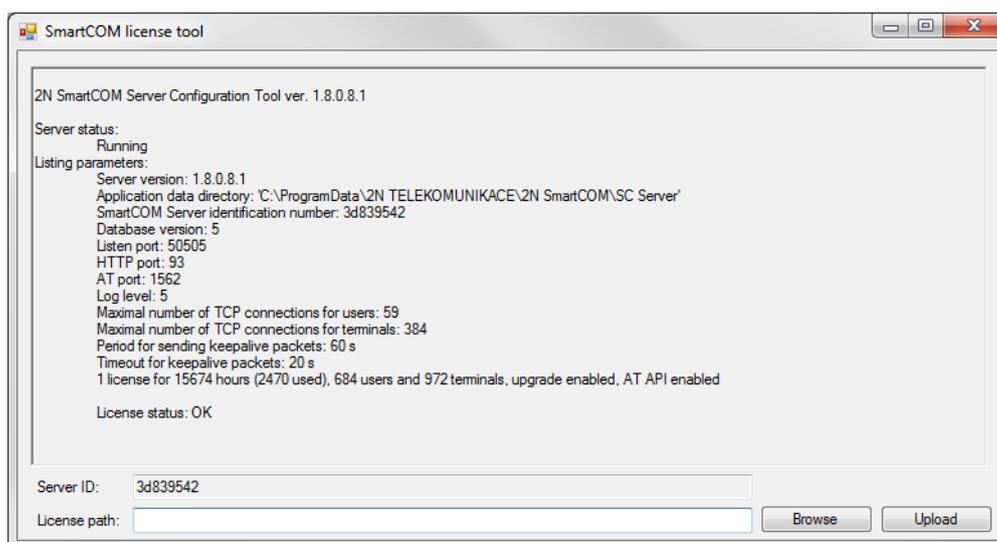
Figure: Licence Tool

A unique ID is generated for your server during installation. Send this code including your licence requirements to **sales@2n.cz** for an extended licence.

Having received the licence file, launch the Licence Tool application and enter the path to this file into the **License path** parameter. Push **Upload** to add the licence to the server directory and restart the server to update the licence data. The licence status will change into OK. Now, the last row of the **Listing parameters** includes all the licensed server properties: count of users, count of terminals and AT API and terminal upgrade enable.

Management

SC_config is an administrator program launched in the 2N® SmartCom Server control row. This application can help you configure all the server parameters: start/stop the server or export/import the server database.

> ⚠ **Caution**
>
> - Use this program to export and import databases safely.
> - You are not advised to copy the sc_config.db database files between servers to avoid data loss.

> ⊘ **Tip**
>
> - The application is launched in the command row. The common path to the program is identical for all WINDOWS XP, Vista and 7 versions:
> - c:\Program Files\2N TELEKOMUNIKACE\2N SmartCom\SC Server\sc_config.exe

Configuration Program

If sc_config.exe is launched without any parameter, the program will list all available information on the server. A typical listing looks as follows:

sc_config.exe

Database Export and Import

You can export the database under normal operation. When the program is launched, a copy of the current database is generated for export purposes so that the database cannot be changed during export. Launch sc_config with the –cExportDatabase parameter to initiate exporting.

The command is as follows:

C:\Program Files\2N TELEKOMUNIKACE\2N SmartCom\SC Server\sc_config.exe –cExportDatabase

The program will create a new file called, e.g., db–20120221095921–export.xml. The number in the filename includes the current database export date and time in the YYYYMMDDhhmmss format.

The database is saved into the user data folder and so can be different for different systems.

> ⓘ **Note**
>
> - A typical storage for WINDOWS XP is C:\Documents and Settings\All Users\Application Data\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\db–20120221095921–export.xml.
>
> - A typical storage for WINDOWS Vista and 7 is C:\ProgramData\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\db–20120221095921–export.xml.

You may import the database with the server switched off only. If the server is running, the database cannot be imported and the program signals an error. Stop the server using the following command:

C:\Program Files\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\sc_config.exe –cStopServer

To import the database, enter the –cImportDatabase parameter and set the absolute path to the file.

C:\Program Files\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\sc_config.exe –cImportDatabase "C:\ProgramData\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\db–20120221095921–export.xml"

When you enter this command, all the database tables will be imported one after another. A successful import of each table is always confirmed after completion.

> ⚠ **Caution**
>
> - The path to the file with the exported database must always be given in brackets as if no path is found, the current database will be deleted and replaced with a new blank one.

> ⓘ **Note**
>
> - If the database contains a high number of users and terminals, the import of tables will take a rather long time, typically a few minutes.

Restart the server upon successful import. Enter the following command:

C:\Program Files\2N TELEKOMUNIKACE\2N SmartCOM\SC Server\sc_config.exe – cStartServer

Setting Options

The sc_config program offers further server parameter settings. See below for the long list.

> ⊘ **Tip**
>
> - These parameters are set as program parameters, i.e. program name< space>program parameter.

General commands

```
-l
```

- Enter this command to list all the parameters.

```
--help or -h
```

- Enter this command to print out this help with the list of commands.

```
--version or -V
```

- Enter this command to print out the program version.

```
-d &lt;data directory&gt;
```

- Enter this command to s et data directory of the server.

Parameter get commands

```
-gListenPort
```

- Enter this command to print out the listen port (port for the 2N® SmartCom protocol).

```
-gHttpPort
```

- Enter this command to print out the HTTP port.

```
-gAtPort
```

- Enter this command to print out the AT API port.

```
-gTerminalPV1url
```

- URL for update of terminals with the 2N® SmartCom protocol version 1 (the oldest ones).

```
-gEncryptionKey
```

- Print packet encryption key.

```
-gEncryptionMode
```

- Print packet encryption mode (0: disabled, 1: optional, 2: forced).

Parameter set commands

```
-sListenPort &lt;listen_port&gt;
```

- Enter this command to set the listen port (port for the 2N® SmartCom protocol).

```
-sHttpPort &lt;http_port&gt;
```

- Enter this command to set the HTTP port.

```
-sAtPort &lt;at_port&gt;
```

- Enter this command to set the port for AT API.

```
-sLogLevel &lt;log_level&gt;
```

- Enter this command to set the level for logs. (1 – 5)

```
-sMaxUser &lt;max_user_tcp_conn&gt;
```

- Enter this command to set the maximum number of users connected to the server at the same time. The maximum value for the parameter is given by the licence.

```
-sMaxTerm &lt;max_terminal_tcp_conn&gt;
```

- Enter this command to set the maximum number of terminals connected to the server at the same time. The maximum value for the parameter is given by the licence.

```
-sKeepAliveTimeout &lt;keepalive_timeout&gt;
```

- Enter this command to set the timeout for the client (user/terminal) to answer the periodical KeepAlive packet in order not to be disconnected. Select a higher value if a high number of terminals (100+) are used to avoid disconnection.

```
-sKeepAlivePeriod &lt;keepalive_period&gt;
```

- Enter this command to set the interval between the KeepAlive packets. This number should be higher than the KEEP_ALIVE_TIMEOUT.

```
-sResetAdmin
```

- Enter this command to reset the Admin user password to '2n'.

```
-sTerminalPV1url
```

- Sets URL for update of terminals with the 2N**®** SmartCom protocol version 1 (the oldest ones). Use the '-reset' parameter to reset the URL default value (' http://star.2n.cz/~fejfar/SC_1-3-1-3-2.bin ').

```
-sEncryptionKey
```

- Set packet encryption key. 32 hexa characters.

```
-sEncryptionMode
```

- Set packet encryption mode (0: disabled, 1: optional, 2: forced).

Server commands

```
-cStartServer
```

- Enter the command to start the 2N® SmartCom Server service.

```
-cStopServer
```

- Enter the command to stop the 2N® SmartCom Server service.

```
-cRestartServer
```

- Enter the command to restart the 2N® SmartCom Server service.

```
-cExportDatabase
```

- Enter the command to export the 2N® SmartCom Server database to an XML file.

```
-cImportDatabase &lt;XML_file&gt;
```

- Enter the command to import the 2N® SmartCom Server database from an XML file. If the 2N® SmartCom Server is not stopped, the operation will fail.

5.2 Control Panel

The **2N® SmartCom Control Panel** provides a comfortable administration of terminals and users/user groups and assigns users relevant communication and administration rights. Also, it can create virtual TCPs and serial ports and connect terminals to them.

> ⚠ **Caution**
>
> - The **2N® SmartCom Control Panel** uses the English language version only.

Installation and Sign-In

Once launched, the installation program will scan your PC. If it finds another **2N® SmartCom Control Panel**, it will ask you whether to uninstall or modify it. If you select Uninstall, the current **2N® SmartCom Control Panel** version will be uninstalled.

Now the **2N® SmartCom** Control Panel Setup Wizard is launched. Follow the Wizard instructions. Select the directory for the **2N® SmartCom Control Panel** to be installed. By default, **C:\Program Files\2N TELEKOMUNIKACE\2N SmartCOM** is used. Also, select whether the application shall be installed for the currently signed-in user or all PC users.

Now the wizard is ready to install the **2N® SmartCom Control Panel**. It automatically adds an item to the **Start** menu and the application icon to the desktop.

> ⓘ **Note**
>
> - The **2N® SmartCom Control Panel** requires **150 MB** of free disk space at least for installation.

Now the **2N® SmartCom Control Panel** is ready for use. Click on the shortcut icon on the desk top (see Fig. 7.1) or select the **Start** menu option to launch the application.

Upon the first launch, you will be asked to enter the username and password and then the server and server listening port. After you do so, use the **Sign in** item.

Figure: 2N® SmartCom Control Panel Icon

> ⚠️ **Caution**
>
> - **Important!** A default user with the Administrator login was created during the **2N® SmartCom Server** installation to make it possible to connect to the server and configure all the other users. The default user login data are as follows:
> - **Username:Admin**
> - **Password: 2n**

If you use one and the same PC for the **2N® SmartCom Control Panel** and the **2N® SmartCom Server**, you can enter the local loop address 127.0.0.1 or the local host text as the server address. To log in to another server, enter the local or public IP address. Or, use the DNS names if you use DNS. Enter the listening port into the **Port** item as selected during your **2N® SmartCom Server** installation. Contact you network administrator for correct settings.

Figure: Sign-In Window

The encryption settings are in the right-hand bottom corner. Click on Change encryption key to display the encryption setting window. If the server requires so, enable packet encryption and set the key. Enter the key either in the ASCII format, where HEX is completed automatically with the translated characters, or enter the HEXA key. The key length is 16 ASCII / 32 HEXA characters. Press OK to sign in with applied encryption. A message will notify a key error if any or encryption options if supported by the server.
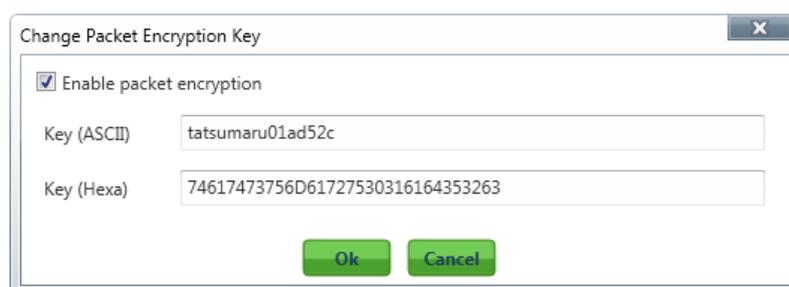
Encryption Settings

Introduction to Application

You will be shown the application layout and main controls in this subsection. The application can be viewed in two ways: the **User mode** and the **Admin mode**. The User may see the menus necessary for its work with terminals only. The Administrator also views the user/user group managing and server setting menus.

The **Status** line displays the following information: From the left: application version, currently signed-in user, count of configured terminals, count of on-line terminals, current server address, server listening port and application status. Find the **Sign-out** button in the bottom part.
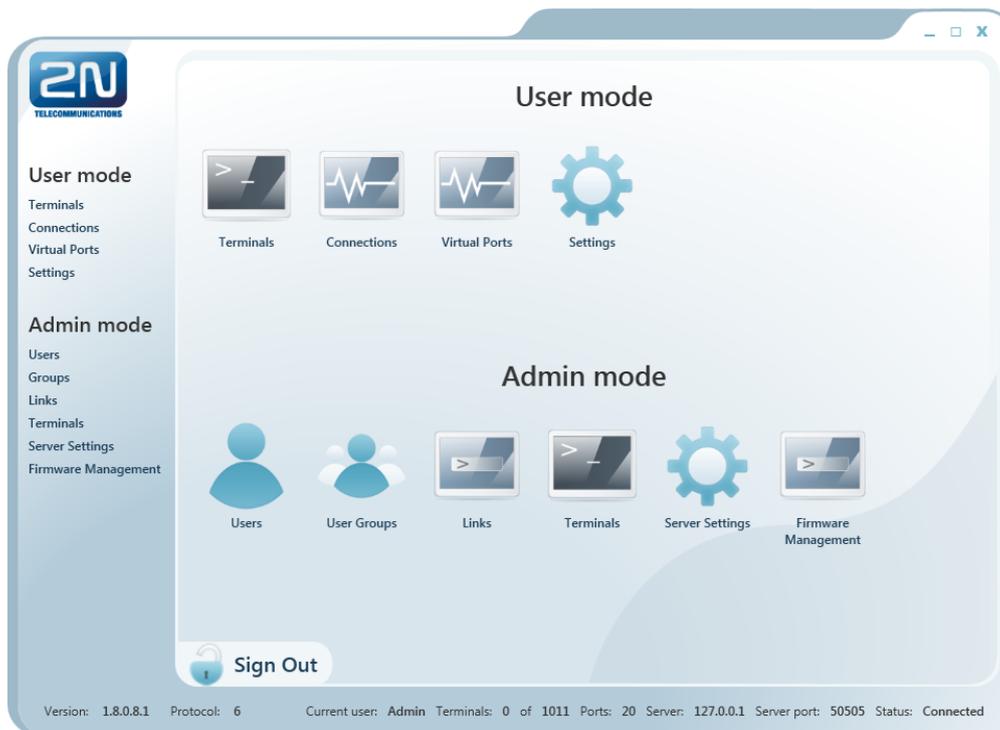
Figure: Application Window

Basic Controls

Home	Push the **Home** button to return to the main menu.
Refresh	Click on **Refresh** to update the status of objects in the selected menu, e.g. the terminal status in the **Terminals** or **Connections** menu.
Start **Stop**	Push these buttons to **Start/Stop** communication with the terminal.
	Push the **Add user** button to add a user..

Basic Controls

	Push the **Add group** button to add a user group.
	Push the general **Add** button to add an object to a group and create new virtual ports and terminals.
	Click on **Edit** to modify the parameters already configured.
	Click on **Remove** to remove an object from configuration.
Change Password	The **Change password** button changes the user password and may be used by the Administrator only.
Create	Push the **Create** button to create new objects.
Cancel	Click on the **Cancel** button to clear the currently made changes.
Save	Click on **Save** to save the currently made changes and send them to the 2N® SmartCom Server configuration.

Basic Controls

Install com0com x86	Use this button to install the virtual serial port driver.
Sign Out	Push the **Sign out** button to log out the currently signed-in user.
Search Filter	Use **Search** to search the relevant table columns.
Switch on **Switch off**	Use **Switch on**/**Switch off** to control the relay contacts.
(i)	Use the button to display details on the terminal.
Update	Click on **Update** to upgrade firmware of selected terminals.
Default	Click on **Default** to reset the factory default values for all parameters.

5.3 Control Panel Use

The main purpose of the 2N® SmartCom Control Panel is to virtualised ports and facilitate connection to terminals. The application also provides comfortable administration and user management including assignment of terminal administration rights, thus allowing users to manage a high number of terminals from one place as well as assign small terminal groups to authorised users.

User Mode

Having singed in as the User, you get access to four items of the main menu only: **Terminals**, **Connections**, **Virtual Ports** and **Settings**. They are all you need for standard work with the terminals. You have been assigned to one of the groups by the Administrator and so you are allowed to view the terminals assigned to this group only. See below for details on the menus.

Terminals

Figure: User Mode Terminals Menu

The **Terminals** menu displays the terminals assigned to the group you have the right to view.

A green dot before the terminal means that the terminal is on-line (connected to the server) and you can get connected to it. A red dot means that the terminal is off-line.

Click on a selected terminal on the right to display the **Terminal Details** window including the name, serial number, hardware revision number and count of available relays and digital and analogue inputs. Notes are also displayed on the terminal if available.

If you are a user, you can select a virtual port for terminal connection. The **Virtual ports** can fully be administered by the user; refer to the **Virtual Ports** subsection below. If you select an unoccupied port, the port will be marked 'in use' signalling to the other users that the port is engaged in the configuration.

If you have the appropriate user right, you can also set the communication parameters for the selected port including the transmission rate and others even if the terminal is connected and used for data transmission. Select the parameters using the combo boxes and the changes will be effective the moment they are saved.

Click on the **Info** button in on-line terminals to display details on the connected terminal, relay control and values measured on analogue and digital inputs.

Figure: Terminals Menu Terminal Details

The **Terminal Details** include the following data: GSM module IMEI (International Mobile Equipment Identity), name and number of the GSM provider at which your **2N®** **SmartCom PRO** is registered, signal intensity in dBm, ICCID (Integrated Circuit Card ID) and IMSI (International Mobile Subscriber Identity). The In roaming parameter indicates whether the SIM card is used in the roaming mode. Moreover, the **2N®** **SmartCom PRO** terminal serial number and current firmware as well as information (firmware version, manufacturer and type) on the GSM module used are displayed.

You can switch the relay contacts in the **Relays** section. The **Digital Inputs** displays the value currently measured at the input and transformed to a logic value (0 or 1) as calibrated. The **Analogue Inputs** displays the values measured at analogue inputs. The Calibrated and Absolute parameters include the percentage against the calibrated range and the absolute value against the A/D converter range respectively.

Refer to **Subs. 3.3** and **3.4**, Input Circuits, for more details on input circuit function and calibration.

Terminal Time Synchronisation

The time setting supporting **PRO** and **PRO ETH** terminals also display the terminal time. Click on **Synchronize** to set the same time value for the terminal as there is in the server.

Connections

Figure: Connections Menu

The **Connections** menu contains a table showing the terminals that are assigned a virtual port. The table includes the terminal name, serial number, IP address and physical port to be connected to the selected virtual port. The **Status** field defines the current of the terminal:

- **Not Connected** - the terminal is on-line connected to the server and ready for virtual port connection.
- **Connecting** - the terminal - virtual port connection is in progress.
- **Connected** - the terminal has been successfully interconnected with the virtual port.
- **Disconnecting** - disconnection from the virtual port is in progress.
- **Terminal Disconnected** - the terminal is off-line, i.e. has been disconnected from the SC server.
- **Disconnect by other user** - the terminal has been disconnected by another user. This means that the other user has interconnected its virtual port with this terminal.
- **Error** - an error occurrred during connection or disconnection.

The **Incoming** and **Outgoing Data** fields display the amount of data transmitted in the given direction and are for information only. Check off **Permanent connection** to enable automatic terminal connection to the virtual port upon the application start.

Press **Start** to interconnect the terminal with the virtual port. If one and the same port is used by multiple terminals, the other users will not be able to press **Start** until the first connection has been finished. Refer to the note below. Push **Stop** to terminate communication.

> ⓘ **Note**
>
> - Although there may be multiple terminals assigned to a virtual port, one terminal can only be assigned to the port within one connection.

In case another user is connected to the terminal, the **Start** button has an orange background colour. You can push the button to disconnect the original user and connect the terminal with your virtual port.

Use the cross to remove the selected row from the table, thus cancelling interconnection between the virtual port of the application and the physical port of the terminal easily.

Virtual Ports
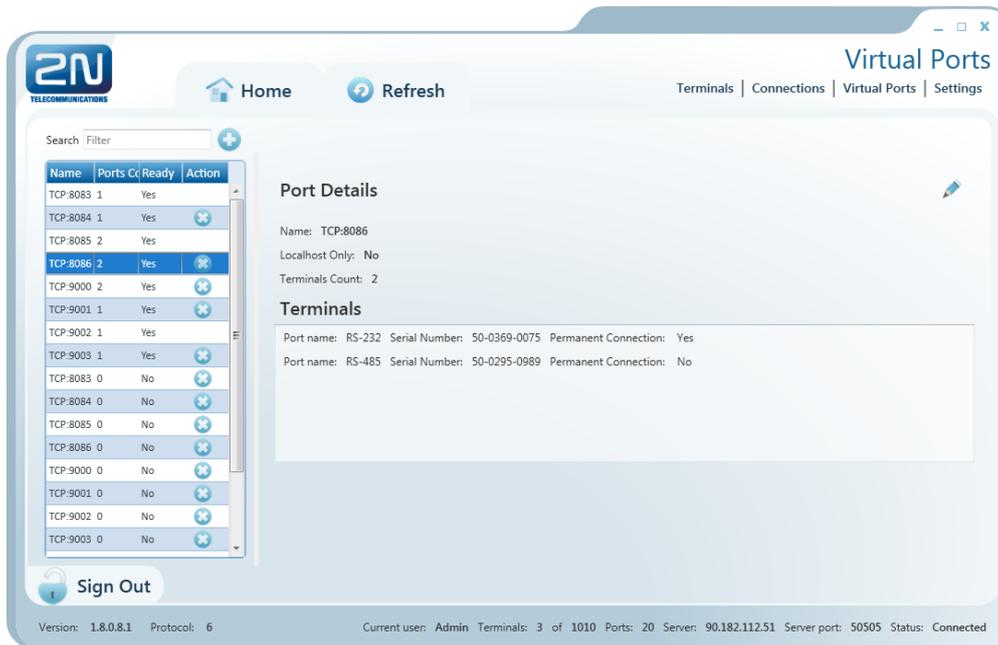
Figure: Virtual Ports Menu

Administer virtual ports using the **Virtual Ports** menu. Every user can use this menu as these settings only relate to the PC with which the user is working. This means that where more PCs are used, the user must create the virtual ports for each of them. This, however, applies to serial ports only; virtual ports are transferable to other PCs. This option, however, can be disabled in the configuration.

Create TCP Port

To create a virtual TCP port, select **Add** and enter the port to be created. Also, tick off **Allow local host only** to enable the port use for the current PC only. Finally, click on **Create**.
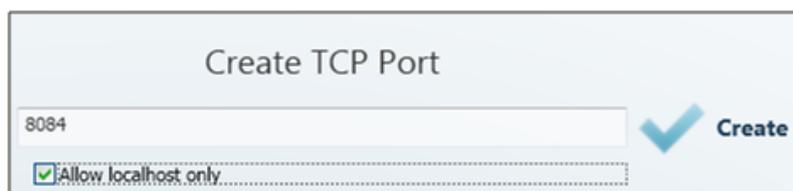
Figure: Create TCP Port

Create COM Port

A virtual COM port is created similarly as instructed above. Just select a port and click on **Create** to make the com0com driver install the selected port into the PC.

Figure: Create COM Port

Having installed a COM port into the PC, activate the COM port. Although the port has been added to the **Virtual ports** table, the **Ready** item shows NO. Therefore, click on this table row to view the port details and the **Refresh** button. Push the button to change the port status from Ready – NO to Ready – YES.
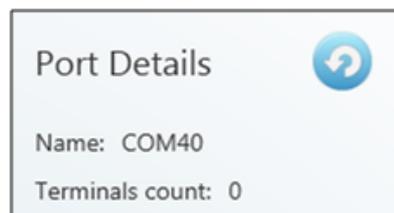
Figure: Refresh Button

Settings
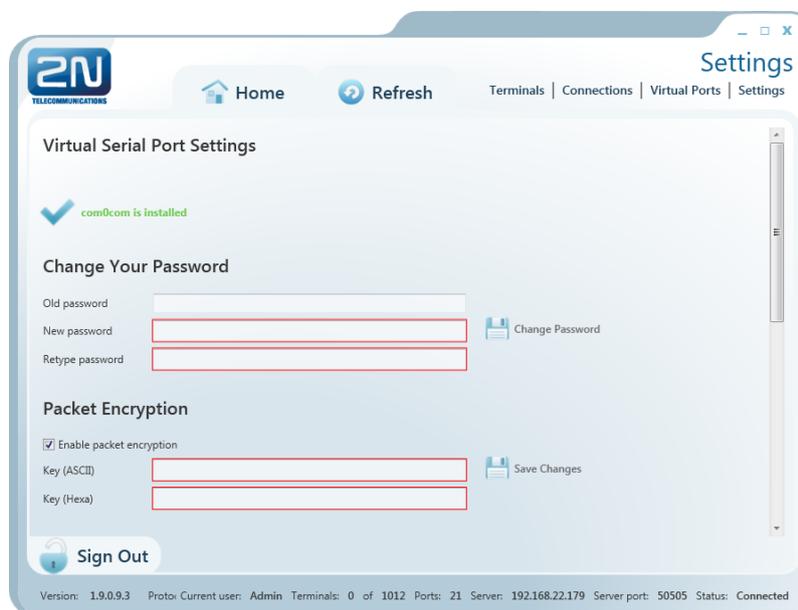
Figure: Settings Men

You can change the user password in the **Settings** menu. Enter the old password and then the new one. The minimum password length and correctness is checked by the parameters. Save the setting using the **Change password** button.

The menu also displays the installation states of the virtual serial port drivers. If no drivers have been installed, you cannot use or add serial ports in the **Virtual ports** menu. If the com0com driver is not installed, the 'com0com is not installed' message gets displayed as shown in Figure. You can install drivers for 32 and 64-bit systems too.

If the installation process is successful, the 'com0com is installed' will be displayed.

> ⚠️ **Caution**
>
> - You need the Admin rights for the PC used for com0com installation. If you do not have such rights currently, the installation will offer you an Admin login option.

> ⓘ **Note**
>
> - The installation requires **0.5 MB** of free disk space at least.

The menu also includes encryption key settings for server communication, like in the sign-in window. Tick the checkbox to enable packet encryption. Enter the selected key into the Key field and save the setting. The setting will not be applied until you sign out. Moreover, you can set the configuration export/import options. Select the Virtual ports, Connections and/or Permanent connections in the Export Settings item to transfer your configuration to another PC. Choose the export destination and press Export to execute the operation.

Setting Export / Import

To import your configuration, sign in as the user to which you want to import the settings. Make sure that this user is identical with the user from which export was made to create a correct configuration file. Now you can load the new configuration onto any account. Select the file to be imported and press Import to import the data. The change will be applied when the user signs out and in.

Moreover, you can set **Automatic sign-in** in this menu. If you enable this function, the application will automatically log in to the server within a preset time interval. If you interrupt autologin in this interval, you can change the login data and sign in as another user, or to another server.
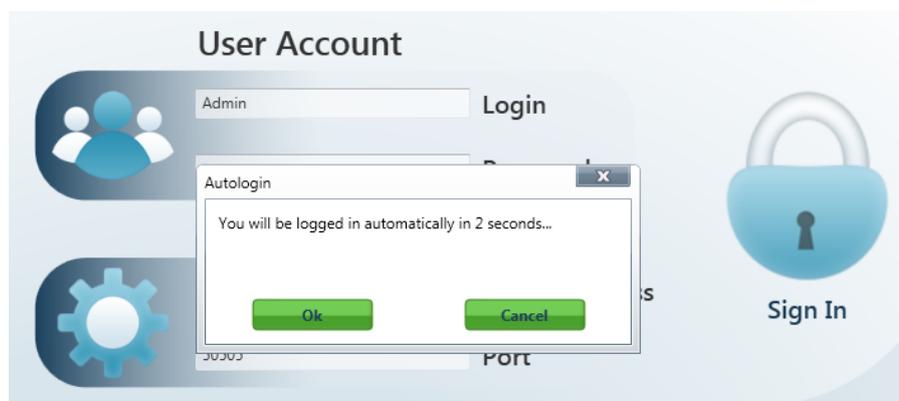
Autologin

> ⚠️ **Caution**
>
> - By interrupting automatic login you also disable the **Allow automatic login** parameter. Thus, you will have to re-enable it to use autologin again.

The application **Logging** checkbox helps you debug the application or just record server communication.

The files are automatically split after reaching 10MB.

Admin Mode

The Admin user has access to all User mode menus plus the user and terminal administering and server setting menus. The **Admin mode** menu includes the **Users**, **Groups**, **Links**, **Terminals** , **Server Settings** and **Firmware Management** menus. See the subsections below for details.

Users
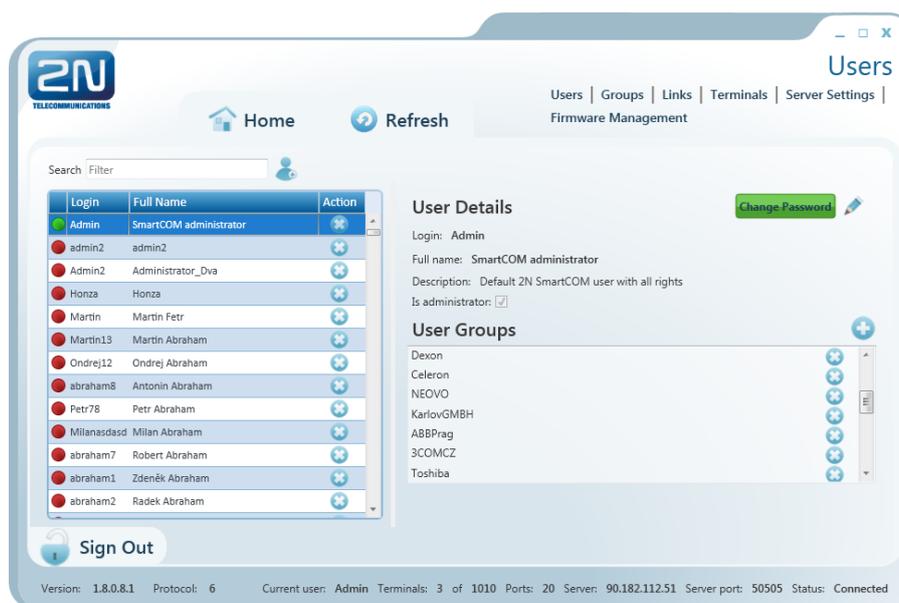
Figure: Users Menu

The **Users** menu is used for user administration. The list of users is on left. A green dot before the user name means that the user is on-line (connected to the 2N$^{®}$ SmartCom Server). A red dot means that the user is off-line (logged out).

Click on the user to display the **User Details** including the full name, login, notes and Admin flag if available. By clicking on **Change password**, the Administrator can change any user password without knowing the original one. This is mainly useful for those users who forget their passwords. Click on **Edit** to change the user details.

The **Groups** menu contains the user groups to which the user is assigned. Click on **Add** and select an item from the list to create a new group. Push **Remove** to remove a group from the list.

Add user

Click on the **Add user** button next to the filter setting. You will be asked to complete the login, full user name and password to ensure a proper function. The parameters are limited and you will be checked during entering. No name or login may be

duplicated in the configuration. You will be notified of any duplication in the configuration. You can also add a note and/or the Admin mode to the user by ticking of the **Is Administrator** option. Click on **Create** to create a new user.

Groups
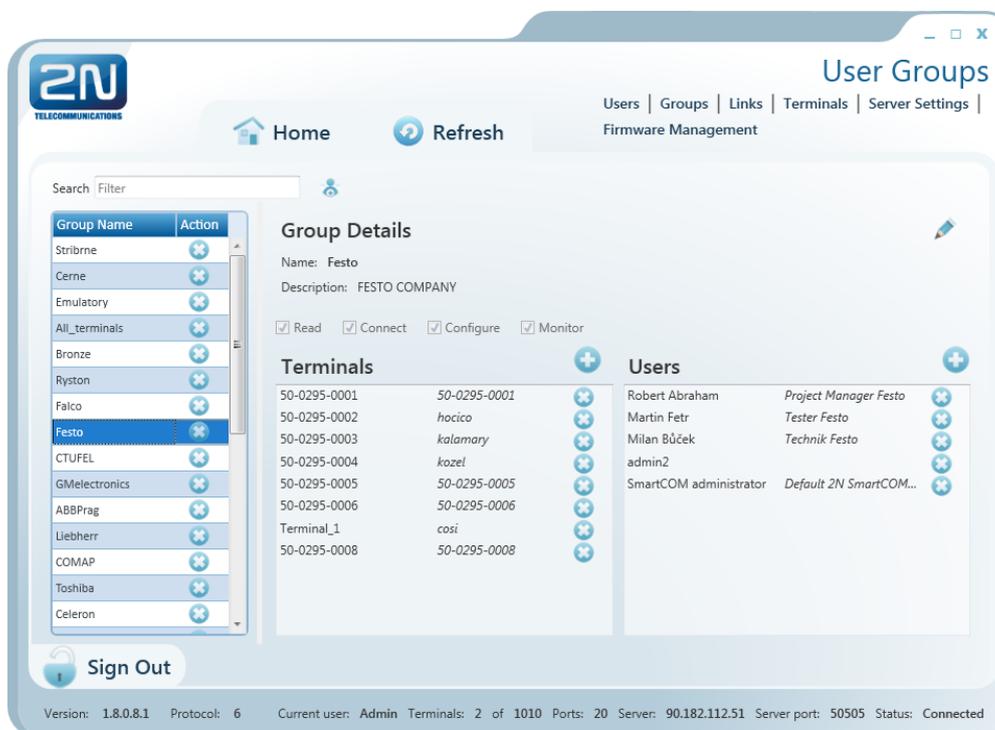
Figure: User Groups Menu

The **Groups** menu helps you group users and terminals. As it is unnecessary for all users to view and manage all terminals, you can create small user groups for terminal data reading and administering.

Click on a group to display the **Group Details** including the full group name, notes if any and rights assigned to the group. Click on **Edit** to change these parameters.

There four group right categories:

- **Read** – users with the **Read** right may view the terminals and monitor their states without being able to configure then or connect to them.

- **Connect** – users with the **Connect** right may view and also connect to the terminals via the **Connections** menu.

- **Configure** – users with the **Configure** right have the advantages of the two user groups above and, moreover, may configure the communication parameters of the terminal ports.

- **Monitor** – not implemented yet. Prepared for future use.

Use the **Terminals** menu to add/remove the terminal to be viewed by the users assigned to this group. Click on **Add** and select an item from the list to add a terminal. Click on **Remove** to remove a terminal from the list.

The **Users** menu includes a list of users assigned to this group. Click on **Add** and select an item from the list to add a user. Click on **Remove** to remove a user from the list.

Add Group

Click on the **Add group** button next to the filter setting. You will be asked to complete the group name and rights to ensure a proper function. The parameters are limited and you will be checked during entering. No name may be duplicated in the configuration. You will be notified of any duplication in the configuration. You can also add a note to the group. Click on **Create** to create a new group.

Links

Obrázek: Links Menu

The **Links** menu displays the currently connected terminals and users. The Administrator can supervise and disconnect them if necessary. The table includes the serial number, IP address and port of the selected terminal, the user currently working with the terminal, the user IP address and active port. Find the **Stop** button in the **Action** field.

> **⊘ Tip**
>
> - The **Stop** button is used for unconditional disconnection of a terminal from the virtual ports regardless of the data transmission state.

Terminals

Figure: Admin Mode Terminals

The **Terminals** menu in the Admin mode is similar to that in the User mode but is extended by group assignment, terminal adding and terminal editing options.

A green dot before the terminal means that the terminal is on-line (connected to the server) and you can get connected to it. A red dot means that the terminal is off-line.

Click on a terminal in the right-hand part of the screen to display the **Terminal Details** including the terminal name, serial number, hardware version as well as count of available relays and digital and analogue inputs. Notes can be added if available. Click on **Edit** to change the terminal details.

Specify in the **Terminal Groups** in which group the selected terminal shall be visible (i. e. which group shall be allowed to use this terminal). Click on **Add** to add a group. Click on **Remove** to remove a group from the list.

You can also set communication for the RS-232 and RS-485/M-BUS ports even if the terminal is connected and used for data transmission. Select the parameters using the combo boxes and the changes will be effective when saved.

Create Terminal

Click on **Add terminal** next to the filter to create a new terminal.

Figure: Create Terminal

To add a terminal, enter the terminal name, serial number and password. The parameters are limited and you will be checked during entering. No name or serial number may be duplicated in the configuration. You will be notified of any duplication in the configuration. Click on **Create** to create a new terminal.

Server Settings

Figure: Server Settings Menu

The **Server Settings** menu contains the 2N® SmartCom Server settings. The menu has two parts.

The first menu part affects the server function directly and any change may to lead to restart and subsequent disabling of work of other users. Set the listening port and http port and enable the http daemon here. The user AT API settings are available here too. Set the port for API connection and communication. There is a button for you to enable/disable the API. Futhermore, set packet encryption for the server. You can enter ASCII characters, which will automatically be translated into Hexa for later server settings. The key length is 32 hexa / 16 ASCII characters. Or, you can enter hexa characters. The ASCII field is thus not applied. See the figure above. Set the encryption mode in the Packet encryption mode parameter. There are three options.

- DISABLED – 2N® SmartCom Server does not encrypt any data.

- OPTIONAL – 2N® SmartCom Server encrypts data if the counterparty requires so.

- FORCED – 2N® SmartCom Server encrypts all incoming and outgoing data.

> ⚠️ **Caution**
>
> - **Important!** A change of these parameters will affect the server function directly. Remember the changed values for later use. For example, if you change the **Listen Port**, you will have to enter a new port during login. If you forget it, you will have to reinstall the whole server!
>
> - **The web interface connection is only possible from the IP addresses from which the terminal or Control Panel were connected.** Any other connecting attempts will be rejected to avoid server violation.

The other part does not affect the server function. You can modify the **Log level** setting, which is a digital value specifying how detailed the LOG file should be. The setting options are 1–5. The default value is 3.

The **Max user** and **Terminal TCP connections** items are automatically set to the highest possible value according to the licence. Enter the required count of connections if necessary. If you exceed the licence limit, the application will automatically limit your selection to the upper limit allowed in the licence.

Also, set the **Keep Alive** packets. The default value is an optimum solution and need not be changed. Should problems occur with the NAT use on application-server-terminal route, lower the time values. The **Application data directory** parameter shows the path to the directory where the 2N® SmartCom Server saves the LOG files and is for information only.

> ⓘ **Note**
>
> - The Application data directory shows the location of the server LOG files. Hence, it is only in the PC on which the **2N® SmartCom** Server is installed. You will not find this directory on the PC from which you launch the **2N® SmartCom Control Panel** and log in to the **2N® SmartCom Server** remotely!

Default Terminal Firmware URL is the address that can be used for firmware upgrade for the terminals. This firmware corresponds to and is fully compatible with the version of your **2N® SmartCom Server** and **2N® SmartCom Control Panel**.

> ⚠ **Caution**
>
> - Remember to keep the firmware address format. The currently valid address is available in the **Server settings** menu. If you enter this link into the terminal, the terminal will replace the sequence of characters '$(SERVER_IP)' with the address of the currently connected server.

The non-editable column to the right includes information on the server to which you are connected such as serial number, licensed hours, maximum counts of users and terminals, upgrade enable/disable and AT API functions. It also shows the remaining server operation hours and the current server time.

Firmware Management
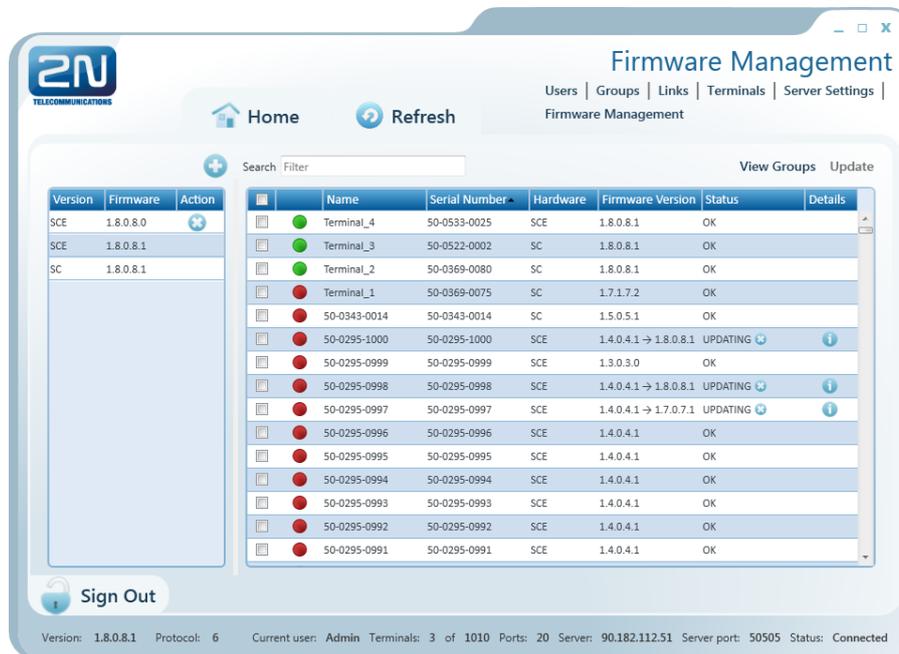
Figure: Firmware Management Menu

The **Firmware Management** menu is used for updating the 2N® SmartCom PRO terminal firmware via the 2N® SmartCom Server. The user selects a firmware version and ticks off the terminal or group of terminals to be updated. This enables the **Update** button. Push the button to execute automatic update for all the selected terminals.

> ⓘ **Note**
>
> - The terminals that are off-line during the update will be updated as soon as they log in to the 2N® SmartCom Server.

There is a list of available FW versions in the left section of the window. Push the **Add** button to add a new firmware version to the list.

> ⚠️ **Caution**
>
> - Just one default firmware version identical with the **2N**® **SmartCom Control Panel** FW version is available after a new server installation. Add more firmware versions to the database if necessary.
>
> - If you just upgrade the **2N**® **SmartCom Server** keeping the configuration instead of installing a new server, the firmware files will be retained. The firmware list will be identical with the original one plus a new default firmware version.
>
> - If you select a SC or SCE terminal for upgrade, the firmware versions matching these HW types will only be listed to avoid false FW upload.

The right-hand window section displays a table including the list of all configured terminals or terminal groups (View Groups/View Terminals).

View Terminals

The table includes six columns. The checkbox helps select a terminal to be updated. The second column indicates whether the terminal is on-line or off-line. The terminal name, serial number and current firmware versions are available here too. An actual or scheduled upgrade is indicated by 'original FW new FW'. The Status field defines the state of the terminal updating process as follows:

- **OK** – update completed, terminal ready for work.
- **UPDATING** – update in progress.
- **ERROR** – update error.
- **UNKNOWN** – terminal not connected yet (FW version unknown).

View Groups

The table includes only names of terminal groups and offers checkboxes for selection. When you select a group and push **Update**, all the terminals assigned to the selected group will be updated. Go back to **View Terminals** to display the update states of all the terminals.

> **⚠ Caution**
>
> - Remember to check off the firmware version and one terminal/terminal group at least to enable the **Update** button. **Otherwise, Update will not be available.**
> - Tick off the checkbox in the table header to select all table items, i.e. all available terminals and/or terminal groups.
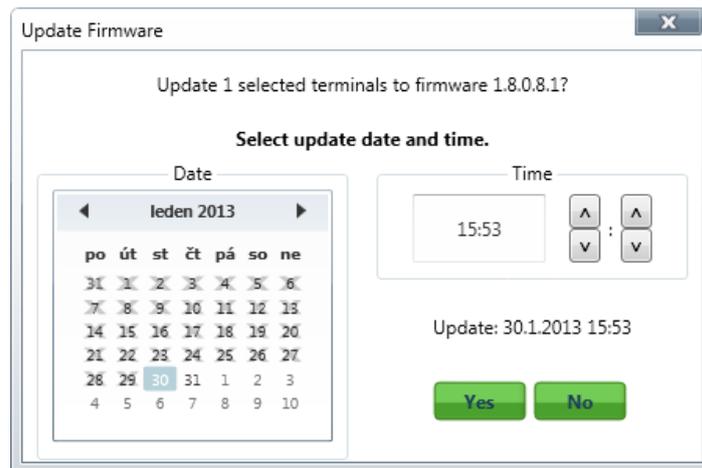
Update Firmware

Scheduled terminal update

Click on **Update** to open a new window for you to schedule the upgrading date and time for the selected terminals. Push Yes to execute update immediately. As the current system time is running on the server, the updating procedure will start immediately. To execute update later, set the required date and time. The update will start automatically on the selected day and at the selected time.

> **ⓘ Note**
>
> - There is a server limitation as to the count of terminals for updating. If you want to update more terminals than 10 at the same time, the first 10 terminals will start updating at the specified time, the next 10 terminals will start updating in one minute and so on until all the selected terminals have been updated. The purpose of this limitation is to protect the server stability and keep the required rate.
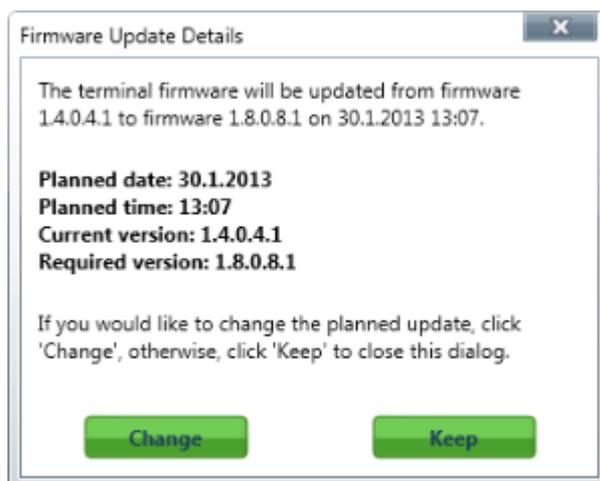
Firmware Update Details

Push the **Info** button in the **Details** column to open the **Firmware update details**, including the upgrade date. Click on **Change** to re-schedule the upgrade, or push **Keep** to retain the original setting and close the window.

5.4 AT API

The AT API is launched on the 2N® SmartCom Server and requires a licence. You have to enable its use in the 2N® SmartCom Control Panel. Refer to the figure below for the basic function.
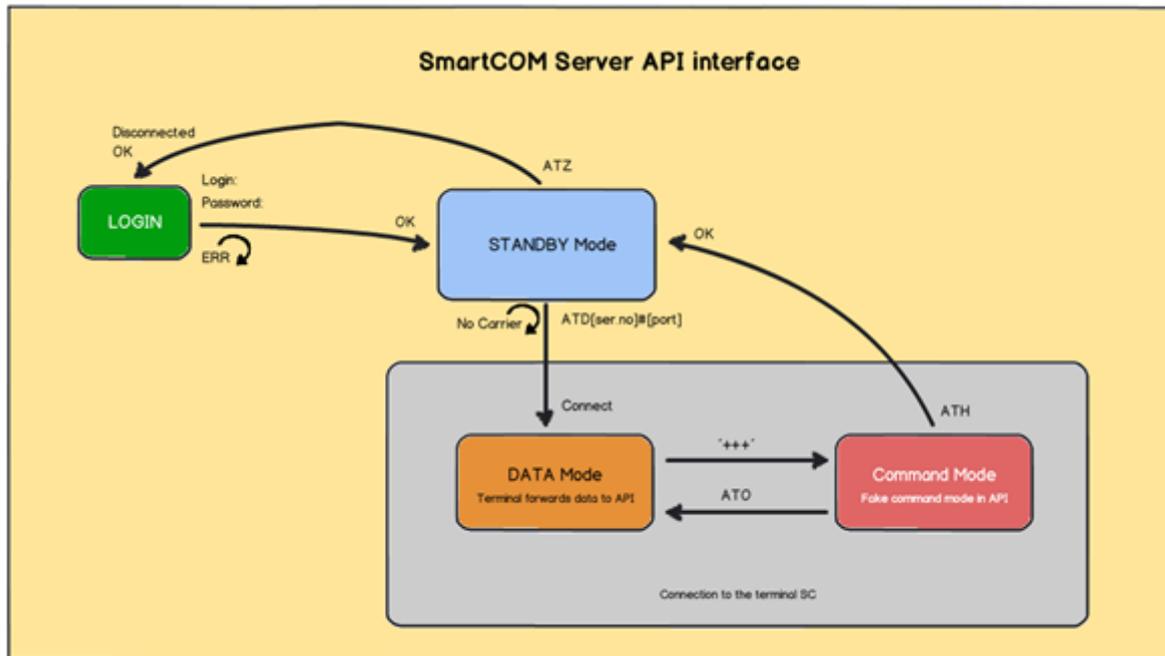
Figure: AT API

AT API Use

Having logged in, get connected via Telnet to the 2N® SmartCom Server address on the port selected for the AT API. After connection, you will be asked to enter the user name and password. You can use any login available on the server. The login dialogue looks as follows:

```
2N® SmartCom AT API
Login:
Admin
Password:
2n
OK
```

Now you are in the AT API environment in the standby mode and can enter the supported commands. Enter the **ATZ** command or close the terminal application to log out.

Basic Commands

Basic commands for the 2N® SmartCom AT API.

```
at
```

- Enter the command to check the interface function. No action is executed and the OK reply is always sent.

```
ate
```

- Enter the command to disable echo (to display just replies, not commands).

```
ate0
```

- Enter the command to disable echo.

```
ate1
```

- Enter the command to enable echo.

```
ati
```

- Enter the command to display the device model – 2N® SmartCom AT API.

```
ati3
```

- Enter the command to display the current server version: x.x.x.x

```
ati4
```

- Enter the command to display the serial number of the **2N® SmartCOM Server**.

```
at+cgmi
```

- Enter the command to display the device manufacturer – 2N Telekomunikace a.s.

```
at+cgmm
```

- Enter the command to display the device model – 2N® SmartCom AT API.

```
at+cgmr
```

- Enter the command to display the current server version: x.x.x.x.x

```
at+gmi
```

- Enter the command to display the device manufacturer – 2N Telekomunikace a.s.

```
at+gmm
```

- Enter the command to display the device model – 2N® SmartCom AT API.

```
at+gmr
```

- Enter the command to display the current server version: x.x.x.x.x

```
at+gsn
```

- Enter the command to display the GSM module IMEI.

```
atz
```

- Enter the command to log out the current user and terminate the TCP connection.

Server Settings

```
at^sccfg?
```

- Enter the command to read out and display the current server settings.
- ^SCCFG: "LISTEN PORT",50505
- ^SCCFG: "HTTP PORT",93
- ^SCCFG: "HTTP ENABLED",1
- ^SCCFG: "AT API PORT",1562
- ^SCCFG: "AT API ENABLED",1
- ^SCCFG: "LOG LEVEL",3
- ^SCCFG: "MAX USER TCP CONNECTIONS",100
- ^SCCFG: "MAX TERMINAL TCP CONNECTIONS",100
- ^SCCFG: "KEEPALIVE TIMEOUT",60
- ^SCCFG: "KEEPALIVE SENDING PERIOD",180
- ^SCCFG: "TERMINAL FW URL","http://$(SERVER_IP):93/firmware/1.4.0.4.2.bin"

```
at^sccfg="listen_port",1560
```

- Enter the command to set the server listening port to 1560.

```
at^sccfg="http_port",8080
```

- Enter the command to set the web server http port to 8080.

```
at^sccfg="http_enabled",1
```

- Enter the command to enable/disable the use of the http interface: 0 – disabled, 1 – enabled.

```
at^sccfg="at_port",1562
```

- Enter the command to set the AT API port to 1562.

```
at^sccfg="server_restart"
```

- Enter the command to restart the 2N® SmartCom Server.

> ⚠ **Caution**
>
> - All the above parameters are saved into the server database but will not become effective until the new database is loaded into the server memory after restart.

Terminal Connection

```
atd&lt;ser_no>#&lt;port>
```

- Enter the command to establish connection to the selected terminal and selected port.

> ⓘ **Note**
>
> - Suppose you want to establish connection to terminal 50–0295–0562 and its RS 232 port. Enter the following command into the API:
>
>   ```
>   ATD5002950562#1
>   ```

```
+++
```

- Enter the command to switch on the command mode for you to enter the basic AT commands for server information and the **ATH** command for connection termination.

```
ato
```

- Enter the command to switch back to the data mode.

```
ath
```

- Enter the command to terminate connection with the currently connected terminal without leaving the API environment. You can go on entering commands and establish new connections.

Terminal Communication

```
at^scstate&lt;ser_no>
```

- Enter the command to know the terminal state. The reply is as follows:

```
Terminal state:<br>
imei: 355915030753263<br>
operator name: T-Mobile CZ (23001)<br>
signal: 21<br>
iccid: 8942001105088009270<br>
imsi: 230015001013627<br>
terminal serial number: 50-0369-0075<br>
terminal firmware: 1.4.0.4.0<br>
module firmware: L23_00gg.WISMO228 140711<br>
module manufacturer: WAVECOM MODEM<br>
module type: WISMO228
```

```
at^screl&lt;ser_no>#&lt;relay>?
```

- Enter the command to know the state of the relay on the selected terminal.

```
at^screl&lt;ser_no>#&lt;relay>=&lt;state>
```

- Enter the command to set the relay contact position.

```
at^screl&lt;ser_no>#&lt;relay>=&lt;state>,&lt;state_after_term_restart>
```

- Enter the command to set the relay contact position and define the contact behaviour after the terminal restart.

```
at^scdin&lt;ser_no>#&lt;input&gt;="get_value"
```

- Enter the command to read the value from the digital input.

```
at^scadc&lt;ser_no>#&lt;input&gt;="get_value"
```

- Enter the command to read the value from the analogue input.

Port Settings

```
at^scport&lt;ser_no>#&lt;port>?
```

- Enter the command to know the current port configuration. The reply is as follows:
- ^SCPORT1, "BAUDRATE",115200
- ^SCPORT1, "DATA BITS",8
- ^SCPORT1, "PARITY",0
- ^SCPORT1, "STOP BITS",1
- ^SCPORT1, "FLOWCONTROL",0

```
at^scport&lt;ser_no>#[port]=&lt;BAR>,&lt;DB>,&lt;P>,&lt;SB>,&lt;FC>
```

- Enter the command to set the selected port configuration. Enter the parameters in a sequence. The setting options are as follows:

```
BAR - "BAUDRATE",110-230400<br>
DB - "DATA_BITS",5-8<br>
P - "PARITY",0-4<br>
SB - "STOP_BITS",1-2<br>
FC - "FLOWCONTROL",0-1
```

> ⚠️ **Caution**
>
> - You can enter all the port parameters or omit them sequentially from the last one. This means that to change the communication baud rate only, enter the following:
>
> ```
> at^scport5002961583#1=57600
> ```
>
> - To set all the port parameters at once, send the following command:
>
> ```
> at^scport5002961583#1=57600,5,0,1,0
> ```

Access to Terminal Configuration Port

Configure the terminal via AT API in the same manner as if you configured the terminal via the RS232 port or TCP port 10000 in the Standalone mode. The configuration port is accessible under number zero. Enter the ATD command as if you accessed the terminal physical ports. Upon connection, you can set even the functions that are inaccessible via the Control Panel or AT API commands such as the user-defined functions (UDF) described below.

> ⓘ **Note**
>
> - Therefore, to configure terminal No. 50–0295–0562, enter the following command into API:
>
> ```
> ATD5002950562#0.
> ```

Firmware Management

Enter the command to control and set terminal upgrades.

```
at^scupg&lt;ser_no&gt;?
```

- The current upgrade state reply is received.

```
at^scupg&lt;ser_no&gt;="&lt;version>"[,&lt;timestamp&gt;]
```

- The FW version and timestamp are set for the terminal upgrade.

ⓘ **Note**

- Enter time as UNIX TIME. Enter 1357055424 for 1.1.2013 15:50:24.

```
at^scupg&lt;ser_no&gt;="CANCEL"
```

- Enter the command to cancel the terminal upgrade.

User, Group and Terminal Management

These commands help you set, edit and delete users, groups and terminals. Log in as a user with administrator rights for this purpose.

ⓘ **Note**

- If the logged in user has no Admin rights, the users, group and terminals that are accessible to such user will only be displayed.
- The user without the Admin rights is not allowed to add, edit or delete the objects.

```
at^scuser?
```

- Enter the command to return the list of all SC Server users.

```
at^scuser="add","&lt;login&gt;","&lt;full_name&gt;","&lt;password&gt;",
&lt;user_flags&gt; [,"&lt;description&gt;"]
```

- Enter the command to add a user to the database.

```
at^scuser="edit",&lt;user_id&gt;,"&lt;login&gt;","&lt;full_name&gt;","&l
t;password&gt;",&lt;user_flags&gt;[,"&lt;description&gt;"]
```

- Enter the command to edit a user with the selected User ID. Always enter all parameters.

```
at^scuser="remove",&lt;user_id&gt;
```

- Enter the command to delete a user with the selected User ID.

```
at^scgroup?
```

- Enter the command to return the list of all SC Server groups.

```
at^scgroup="add","&lt;name&gt;"[,"&lt;description&gt;"]
```

- Enter the command to add a group to the database.

```
at^scgroup="edit",&lt;group_id&gt;,"&lt;name&gt;"[,"&lt;description&gt;"
]
```

- Enter the command to edit a group with the selected Group ID. Always enter all parameters.

```
at^scgroup="remove",&lt;group_id&gt;
```

- Enter the command to delete a group with the selected Group ID.

```
at^scgroup="list_user",&lt;group_id&gt;
```

- Enter the command to list users assigned to a group.

```
at^scgroup="list_terminal",&lt;group_id&gt;
```

- Enter the command to list terminals assigned to a group.

```
at^scgroup="list_right",&lt;group_id&gt;
```

- Enter the command to list all group rights.

```
at^scgroup="add_user",&lt;group_id&gt;,&lt;user_id&gt;
```

- Enter the command to assign a user to the specified group.

```
at^scgroup="add_terminal",&lt;group_id&gt;,&lt;terminal_id&gt;
```

- Enter the command to assign a terminal to the specified group.

```
at^scgroup="remove_user",&lt;group_id&gt;,&lt;user_id&gt;
```

- Enter the command to remove a user from the specified group.

```
at^scgroup="remove_terminal",&lt;group_id&gt;,&lt;terminal_id&gt;
```

- Enter the command to remove a terminal from the specified group.

```
at^scgroup="set_right",&lt;group_id&gt;,READ,&lt;0|1&gt;
```

- Enter the command to assign or remove READ right to/from the specified group.

```
at^scgroup="set_right",&lt;group_id&gt;,CONN,&lt;0|1&gt;
```

- Enter the command to assign or remove CONNECT right to/from the specified group.

```
at^scgroup="set_right",&lt;group_id&gt;,CONF,&lt;0|1&gt;
```

- Enter the command to assign or remove CONFIGURE right to/from the specified group.

```
at^scgroup="set_right",&lt;group_id&gt;,MONI,&lt;0|1&gt;
```

- Enter the command to assign or remove MONITOR right to/from the specified group.

```
at^scterminal?
```

- Enter the command to return the list of all SC Server terminals.

```
at^scterminal="add","&lt;name&gt;","&lt;serial_number&gt;","&lt;
password&gt;" ,"&lt;description&gt;"]
```

- Enter the command to add a terminal to the database.

```
at^scterminal="edit",&lt;terminal_id&gt;,"&lt;name&gt;","&lt;
serial_number&gt;","&lt;password&gt;"[,"&lt;description&gt;"]
```

- Enter the command to edit a terminal with the selected Terminal ID. Always enter all parameters.

```
at^scterminal="remove",&lt;terminal_id&gt;
```

- Enter the command to delete a terminal with the selected Terminal ID.

URC

URC (Unsolicited Result Codes) notifies state changes of the following objects: users, terminals and connections.

```
at^scind?
```

- Enter the command to write out to which URC the user is logged in.

```
at^scind="&lt;type&gt;",&lt;value&gt;
```

- Enter the command to log in/out the user to/from URC.
  - <type> select the URC type (user / terminal / connection / all)
  - <value> 0 = disabled / 1 = enabled

> ⓘ **Note**
>
> - To log in to the terminal URC, enter the following command into AT API: *at^scind="terminal",1*

URC format

- user connection state change (like in response to at^scuser?): ^SCUSER: <user_id>, "<login>", "<full_name>", "<user_flags>", "<connection_state>", "<description>"
- terminal connection state change (like in response to at^scterminal?): ^SCTERMINAL: <terminal_id>, "<name>", "<serial_number>", "<connection_state>", "<description>"
- connection state change at terminal port: ^SCLINK: <link_id>, "<terminal_name>", "<serial_number>", "<port_name>", "<user_name>", "<connection_state>"

6. List of Supported AT Commands

This section provides a list of all AT commands that can be used for the 2N® SmartCom PRO configuration including their formats and brief function description. Refer to the interface subsections mentioned in the headers for details.

GSM Commands	Subs 3.1
at^sccfg="mode",1	Sets the mode: TCPCLIENT, STANDALONE and DATA_OFF.
at^sccfg="gprs_apn","internet.open"	Sets the access APN to the mobile provider.
at^sccfg="gprs_user","internet"	Sets the user name for provider authentication.
at^sccfg="gprs_pass","hfsdj515d"	Sets the user password for provider authentication.
at^sccfg="local_ip"	Returns the IP address assigned by the provider.
at^sccfg="server_ip","90.182.112.54"	Sets the server IP address.
at^sccfg="server_port",1564	Sets the SC server port.
at^sccfg="auth_pass","xJ32ppp_v1"	Sets the authorisation password for the above mentioned server.
at^sccfg="sim_pin","1156"	Sets the PIN.
at^sccfg="auth_ip",1	Enables/disables IP address access authorisation.

GSM Commands	Subs 3.1
at^sccfg="encrypt",1	Sets the encryption mode.
at^sccfg="encrypt_key", "E2978FE2978FE2978FE2978FE2978F20"	Sets the encryption key.
at^sccfg="save"	Saves the changes.
at^sccfg="restart"	Restarts the interface.
at^sccfg="srestart"	Saves the changes and restarts the interface.
at^sccfg?	Displays the currently set values.
at^sccfg=?	Displays the interface setting options.
at^sccfg="listenportcmd",10000	Specifies the listening port for IP access to the configuration port.
at^sccfg="listen_port1",10001	Specifies the listening port for IP access to port1.
at^sccfg="listen_port2",10002	Specifies the listening port for IP access to port2.

Port 1 (RS 232)	Subs. 3.3
at^scport1="baudrate",115200	Sets the baud rate to 115200 bps.
at^scport1="data_bits",8	Sets the data bit count to 8.
at^scport1="stop_bits",2	Sets the stop bit count to 2.
at^scport1="parity",2	Sets the parity security type.
at^scport1="flowcontrol",1	Sets the flow control type.
at^scport1="save"	Saves the changes.

Port 1 (RS 232)	Subs. 3.3
at^scport1="restart"	Restarts the interface.
at^scport1="srestart"	Saves the changes and restarts the interface.
at^scport1?	Displays the currently set interface values.
at^scport1=?	Displays the interface setting options.
at^sc232tmclose=5	Sets tmclose to 5 s.
at^sc232tmclose?	Displays the currently set values.
at^sc232tmclose=?	Displays the setting options.

Port 2 (RS 485/M-Bus/RS 232)	Subs. 3.3
at^scport2="baudrate",9600	Sets the baud rate to 9600 bps.
at^scport2="data_bits",8	Sets the data bit count to 8.
at^scport2="stop_bits",2	Sets the stop bit count to 2.
at^scport2="parity",2	Sets the parity security type.
at^scport2="save"	Sets the flow control type.
at^scport2="restart"	Saves the changes.
at^scport2="srestart"	Saves the changes and restarts the interface.
at^scport2?	Displays the currently set interface values.
at^scport2=?	Displays the interface setting options.

Relay Outputs	Subs. 3.1
at^screl1?	Detects the state of relay 1.
at^screl1=0	Switches relay 1 to the open state.
at^screl1=0, 0	Switches relay 1 to the open state. The relay will be open upon terminal restart.
at^screl1=0, 1	Switches relay 1 to the open state. The relay will be closed upon terminal restart.
at^screl1=0, 2	Switches relay 1 to the open state. Upon terminal restart, the relay will be in the state in which it was before restart.
at^screl1=1	Switches relay 1 to the closed state.
at^screl1=1, 0	Switches relay 1 to the closed state. The relay will be open upon terminal restart.
at^screl1=1,1	Switches relay 1 to the closed state. The relay will be closed upon terminal restart.
at^screl1=1, 2	Switches relay 1 to the closed state. Upon terminal restart, the relay will be in the state in which it was before restart.
at^screl2?	Detects the state of relay 2.
at^screl2=0	Switches relay 2 to the open state.
at^screl2=0, 0	Switches relay 2 to the open state. The relay will be open upon terminal restart.
at^screl2=0, 1	Switches relay 2 to the open state. The relay will be closed upon terminal restart.
at^screl2=0, 2	Switches relay 2 to the open state. Upon terminal restart, the relay will be in the state in which it was before restart.
at^screl2=1	Switches relay 2 to the closed state.

Relay Outputs	Subs. 3.1
at^screl2=1, 0	Switches relay 2 to the closed state. The relay will be open upon terminal restart.
at^screl2=1, 1	Switches relay 2 to the closed state. The relay will be closed upon terminal restart.
at^screl2=1, 2	Switches relay 2 to the closed state. Upon terminal restart, the relay will be in the state in which it was before restart.

ADC - Input Circuits	Subs. 3.1
at^scadc1="get_value"	Performs measurement and sends the value measured.
at^scadc1="adc_value"	Performs measurement and returns the A/D converter value.
at^scadc1="calib_min"	Interface calibration according to the currently measured value.
at^scadc1="calib_max"	Interface calibration according to the currently measured value.
at^scadc1="threshold_low",200	Manual calibration setting (minimum values).
at^scadc1="threshold_high", 250	Manual calibration setting (maximum values).
at^scadc1="save"	Saves the changes.
at^scadc1?	Displays the currently set interface values.
at^scadc1=?	Displays the interface setting options.
at^scadc2="get_value"	Performs measurement and sends the value measured.
at^scadc2="adc_value"	Performs measurement and returns the A/D converter value.

ADC - Input Circuits	Subs. 3.1
at^scadc2="calib_min"	Interface calibration according to the currently measured value.
at^scadc2="calib_max"	Interface calibration according to the currently measured value.
at^scadc2="threshold_low", 200	Manual calibration setting (minimum values).
at^scadc2="threshold_high", 250	Manual calibration setting (minimum values).
at^scadc2="save"	Saves the changes.
at^scadc2?	Displays the currently set interface values.
at^scadc2=?	Displays the interface setting options.
at^scdin1="get_value"	Performs measurement and sends the value measured.
at^scdin1="adc_value"	Performs measurement and returns the A/D converter value.
at^scdin1="calib_min"	Interface calibration according to the currently measured value.
at^scdin1="calib_max"	Interface calibration according to the currently measured value.
at^scdin1="threshold_low",200	Manual calibration setting (minimum values).
at^scdin1="threshold_high",250	Manual calibration setting (minimum values).
at^scdin1="save"	Saves the changes.
at^scdin1?	Displays the currently set interface values.
at^scdin1=?	Displays the interface setting options.
at^scdin2="get_value"	Performs measurement and sends the value measured.

ADC - Input Circuits	Subs. 3.1
at^scdin2="adc_value"	Performs measurement and returns the A/D converter value.
at^scdin2="calib_min"	Interface calibration according to the currently measured value.
at^scdin2="calib_max"	Interface calibration according to the currently measured value.
at^scdin2="threshold_low",200	Manual calibration setting (minimum values).
at^scdin2="threshold_high", 250	Manual calibration setting (minimum values).
at^scdin2="save"	Saves the changes.
at^scdin2?	Displays the currently set interface values.
at^scdin2=?	Displays the interface setting options.

S0 - Pulse Input	Subs. 3.1
at^scpulse1="state"	Detects the current state.
at^scpulse1="get_value"	Reads out the counter state.
at^scpulse1="set_value",100	Sets the immediate pulse count.
at^scpulse1="stimer",60	Sets the EEPROM storing interval for counter values.
at^scpulse1="start"	Starts counting on input 1.
at^scpulse1="stop"	Stops counting on input 1.
at^scpulse1="clear"	Clears the counter.
at^scpulse1="save"	Manual value storing.
at^scpulse1?	Displays the currently set values.

S0 - Pulse Input	Subs. 3.1
at^scpulse1=?	Displays the setting options.
at^scpulse2="state"	Detects the current state.
at^scpulse2="get_value"	Reads out the counter state.
at^scpulse2="set_value",100	Sets the immediate pulse count.
at^scpulse2="stimer",60	Sets the EEPROM storing interval for counter values.
at^scpulse2="start"	Starts counting on input 1.
at^scpulse2="stop"	Stops counting on input 1.
at^scpulse2="clear"	Clears the counter.
at^scpulse2="save"	Manual value storing.
at^scpulse2?	Displays the currently set values.
at^scpulse2=?	Displays the setting options.

ETH Settings	Subs. 3.1
at^sceth="ip_method","DHCP"	Sets the IP address setting mode.
at^sceth="local_mac"	Writes out the currently used MAC address.
at^sceth="local_ip"	Writes out the currently used IPv4 address.
at^sceth="fixed_ip","192.168.1.1"	Manual IP address setting.
at^sceth="mask","255.255.255.0"	Manual network mask setting.
at^sceth="gateway","192.168.1.254"	Manual default gateway setting.
at^sceth="dns1","0.0.0.0"	Manual primary DNS setting.

ETH Settings	Subs. 3.1
at^sceth="dns2","0.0.0.0"	Manual secondary DNS setting.
at^sceth="save"	Saves the changes.
at^sceth="restart"	Restarts the Ethernet port.
at^sceth="srestart"	Saves the changes and restarts the Ethernet port.
at^sceth?	Displays the currently set values.
at^sceth=?	Displays the setting options.
at^scchprim="GSM"	Sets the primary communication channel.
at^scchprim?	Displays the currently set values.
at^scchled1="GSM"	Sets the LED indicating function.
at^scchled1?	Displays the currently set values.

IP Address Access Authorisation	Subs. 3.9
at^sccfg="auth_ip",1	Enables/disables IP address access authorisation.
at^scipauth="add","54.182.153.21"	Adds an IP address to the list.
at^scipauth="remove",2	Removes an IP address from the list.
at^scipauth="clear"	Clears all addresses from the list.
at^scipauth?	Displays the currently set values.
at^scipauth=?	Displays the setting options.

Charger	Subs. 3.1
at^scchg="max_voltage", 6800	Sets the maximum accumulator voltage.
at^scchg="max_current", 350	Sets the maximum charging current.
at^scchg="state"	Detects the current charging state.
at^scchg="start"	Starts charging.
at^scchg="stop"	Stops charging.
at^scchg="save"	Saves the changes.
at^scchg="srestart"	Saves the changes and resets the charging parameters immediately.
at^scchg?	Displays the currently set values.
at^scchg=?	Displays the setting options.

Real Time Clock	Subs. 3.1
at^scclk?	Displays the real time value.
at^scclk="RR/MM/DD, hh:mm:ss+ZZzz "	Sets the real time clock.

WM-Bus	Subs. 3.6
at^scwmbus="dev_add","MAN", 12345678,600,01,07	Adds a meter to the list.
at^scwmbus="decrypt",1	Enables decryption of messages captured on the module.
at^scwmbus="key_add",1, A5B95C144134DE257AF2ED4F384C7EB7	Sets the encryption key for meter 1.

WM-Bus	Subs. 3.6
at^scwmbus="dev_remove",1	Removes a meter from the meter list.
at^scwmbus="dev_clear"	Clears the whole meter list.
at^scwmbus="get_oldest"	Writes out the oldest record from the data zone.
at^scwmbus="get_oldest"<,n>	Provides bulk reading of n messages.
at^scwmbus="del_oldest"	Deletes the oldest record.
at^scwmbus="del_oldest"<,n>	Provides bulk deleting of n messages.
at^scwmbus="del_all"	Deletes all records in the data zone.
at^scwmbus="mode",11	Sets the device working mode.
at^scwmbus="get_space"	Displays free memory space for WM-Bus message storing.
at^scwmbus="mess_cnt"	Displays the count of WM-Bus messages received.
at^scwmbus="RF_channel",1	Specifies the WM-Bus transmission channel.
at^scwmbus="preamble_len",0	Specifies the framework format.
at^scwmbus="Install_mode",0	Selects the normal or filter off mode.
at^scwmbus="restart"	Restarts the WM-Bus module.
at^scwmbus="sniff"	Switches the WM-Bus module into the sniff mode.
at^scwmbus="module_type"	Displays information on the WM-Bus module.
at^scwmbus="cf_discard",0	Switches on the mode in which messages captured in the Compact Frame (CF) format are discarded.
at^scwmbus?	Displays the currently set interface values.

WM-Bus	Subs. 3.6
at^scwmbus=?	Displays the setting options for the Wireless M–Bus interface.

AMS	Subs. 3.8
AT^SCAMS=?	Writes out the list of AMS module control commands.
AT^SCAMS?	Writes out the contents of the AMS-served device list.
AT^SCAMS="ENABLE",1	Writes out the current on/off state of the AMS module.
AT^SCAMS="RETRY",5	Sets the count of device data reading retries.
AT^SCAMS="RETRY"	Detects the currently set count of data reading retries.
AT^SCAMS="DEV_ADD",2,2,"128","1H"	Adds a device connected to port 2 to the device list.
AT^SCAMS="DEV_ADD",128,"AIN_DIN","2/DIN","2H"	Adds a device connected to the A/D input port to the device list.
AT^SCAMS="DEV_REMOVE",4	Removes the index 4 device from the device list.
AT^SCAMS="DEV_CLEAR"	Clears all devices from the device list.
AT^SCAMS="DEV_INTERVAL",3,"1H"	Changes the reading interval to 1 hour for index 3 device in the device list.
AT^SCAMS="DEV_INTERVAL","ALL","30M"	Changes the reading interval to 30 minutes for all devices in the device list.
AT^SCAMS="DIRECT_READ",2,1,"947833"	Performs immediate data reading and listing for a defined device.
AT^SCAMSADD=2,2,"default","2h",3	Detects a device on the bus.
AT^SCDATA=?	Writes out a control command list for the stored data record reading module.

AMS	Subs. 3.8
AT^SCDATA="MESS_CNT"	Returns the count of stored data records.
AT^SCDATA="GET_SPACE"	Returns the available memory size for data record storing.
AT^SCDATA="GET_OLDEST",5	Provides bulk listing of 5 messages.
AT^SCDATA="GET_FILTERED","RECORD_TYPE==WMBUS"	Writes out all the data records stored in the data record memory.
AT^SCDATA="DEL_OLDEST"	Deletes the oldest record.
AT^SCDATA="DEL_OLDEST",5	Deletes 5 oldest records.
AT^SCDATA="DEL_ALL"	Deletes all records in the data zone.

ZigBee	Subs. 3.7
AT^SCZB="ADD", 00124B00042635C1,6	Adds a device to the list.
AT^SCZB?	Reads out the meter list.
AT^SCZB="DEL",4	Removes a device from the list.
AT^SCZB="CLEAR"	Clears all devices from the list.
AT^SCZB="PANID",0458	Sets the network identifies to the value specified in the parameter.
AT^SCZB="PANID_READ"	Reads out the current PANID value.
AT^SCZB="CHANNEL", 00001000	Sets the used channel to the value specified in the parameter.
AT^SCZB="CHANNEL_READ"	Reads out the currently set value.

ZigBee	Subs. 3.7
AT^SCZB="PRECFGKEY"[,<32 HEX>]	Sets the pre-configured key to be used for encrypted communication.
AT^SCZB=" PRECFGKEY_ENABLE",1	Enable/disable encryption key sending.
AT^SCZB="TCLK"[,<32 HEX>]	Sets the encryption key for communication key negotiation and transmission.
AT^SCZB="IEEE_ADDR"	Reads out and displays the ZigBee IEEE address.
AT^SCZB="MODULE_TYPE"	Reads out and displays information on the module.
AT^SCZB="RESTART"	Restarts the interface and creates a new PAN.
AT^SCZB="POLL",6	Reads out and updates the device list data for a defined device.
AT^SCZB="POLL_TIME",500	Reads out and updates the device list data for all devices automatically.
AT^SCZB="PERMIT_JOIN",2,60	Permits new devices to join the network.
AT^SCZB="PERMIT_JOIN_ALL", 120	Permit all the devices in the device list to let new devices join the network.
AT^SCZB="ANLGVAL",<dev_id>, <EP>	Reads out the analogue input value in a device on a selected end_point.
AT^SCZB="BASIC",<dev_id>, <EP>	Reads out all the values included in the BASIC cluster.
AT^SCZB="DEVTEMP",<dev_id>, <EP>	Reads out the current device temperature.
AT^SCZB="FAN",<dev_id>,<EP>	Returns the current state of the fan.
AT^SCZB="FAN_MODE", <dev_id>,<EP>,4	Sets the fan mode to the defined value.

ZigBee	Subs. 3.7
AT^SCZB="HMDT",<dev_id>, <EP>	Returns a moisture value in % with the accuracy of 2 decimal places.
AT^SCZB="LC",<dev_id>,<EP>	Returns the Level Control value.
AT^SCZB=" LC_MOVE_TO_LEVEL",<dev_id>, <EP>,50,10	Sets the Level Control value (flap position).
AT^SCZB="METER",<dev_id>, <EP>	Writes out the immediate consumption values.
AT^SCZB="MSTVAL",<dev_id>, <EP>	Reads out the Multi State Value.
AT^SCZB="ONOFF",<dev_id>, <EP>	Returns the value 1/0 depending on the immediate setting.
AT^SCZB="ONOFF_SET", <dev_id>,<EP>,1	Sets the switch to the value specified in the parameter.
AT^SCZB="TEMP",<dev_id>, <EP>	Returns a temperature value in °C with the accuracy of 2 decimal places.
AT^SCZB="THERM",<dev_id>, <EP>	Reads out and displays the currently set thermostat mode.
AT^SCZB="THERM_MODE", <dev_id>,<EP>,00	Sets the thermostat mode to the preset value.
AT^SCZB=" THERM_TEMP_HEAT",<dev_id>, <EP>,20,00	Sets the temperature value for heating.
AT^SCZB=" THERM_TEMP_COOL",<dev_id>, <EP>,20,00	Sets the temperature value for cooling.
AT^SCZB="TIME",<dev_id>,<EP>	Returns the currently set time value and time status.

ZigBee	Subs. 3.7
AT^SCZB="TIME_SET",<dev_id>, <EP>,<utc_time>	Sets time in a device to the value specified in the parameter.
AT^SCZB="GROUP_ADD", <dev_id>,<EP>,1,group1	Add a device with the selected dev_id to the group defined in group_id.
AT^SCZB=" GROUP_MEMBERSHIP",<dev_id>, <EP>,1	Returns the YES string if the device defined in dev_id is assigned to the group defined in group_id.
AT^SCZB="GROUP_REMOVE", <dev_id>,<EP>,1	Removes the device defined in dev_id from the group defined in group_id.
AT^SCZB=" GROUP_REMOVE_ALL", <dev_id>,<EP>	Removes the device defined in dev_id from all the groups.
AT^SCZB=" SCENE_RECALL_ALL", <group_id>,<scene_id>	Starts the scenario defined in scene_id in all the devices in the group defined in group_id.
AT^SCZB="IDENTIFY",<dev_id>, <EP>	For internal use only. Non-configurable!
AT^SCZB="IDENTIFY_START", <dev_id>,<EP>	For internal use only. Non-configurable!
AT^SCZB="2NCST",<dev_id>, <EP>	For internal use only. Non-configurable!
AT^SCZB="2NCST_WRITE", <dev_id>,<EP>	For internal use only. Non-configurable!

SC Ping	Subs. 3.5
AT^SCPING="host"," 8.8.8.8"	Sets the IP address of the Host to which the ICMP request is to be sent.

SC Ping	Subs. 3.5
AT^SCPING="interval",5	Sets an interval in minutes. 0 means that periodical pings are deactivated.
AT^SCPING="test"	Used for testing configuration before saving.
AT^SCPING="save"	Saves the changes.
AT^SCPING="restart"	Restarts the function.
AT^SCPING="srestart"	Saves the changes and immediately resets the ping parameters.
AT^SCPING?	Displays the currently set interface values.
AT^SCPING=?	Displays the interface setting options.

User Defined Functions	Subs. 3.5
AT^SCUDF="ADD","AIN1>=50:SMS P[+420123456789] m[alarm]"	Adds a condition to the condition list.
AT^SCUDF="CLEAR"	Clears all conditions from the terminal memory.
AT^SCUDF="REMOVE",2	Deletes the condition with the selected ID.
AT^SCUDF?	Writes out all the conditions saved.
AT^SCUDF=?	Displays the setting options.
AT^SCUDF="add","AIN1>50:SAVE"	Saves the condition.
AT^SCUDF="add","AIN1>50:REL1 r[1]"	Switches the relay state.
AT^SCUDF="add","AIN1>50:SMS p[+420123456789] m[text 1]"	Sends an SMS.

User Defined Functions	Subs. 3.5
AT^SCUDF="add","AIN1>50:TCP i[62.134.22.87:8026] m[text 1]"	Sends a TCP message.
AT^SCUDF="add","AIN1>50:CNT1 d[1]"	Increments the counter.

General Commands	
AT	Does not perform any action and OK is always sent to it.
ATE	Disables echo (the sent commands are no more displayed and replies are displayed only).
ATE0	Disables echo.
ATE1	Enables echo.
at+cgmi	Displays the device manufacturer – 2N Telekomunikace a.s.
at+cgmm	Displays the device model.
at+cgmr	Displays the current firmware version: x.x.x.x.x
at+cgsn	Displays the GSM module IMEI.
ATI	Displays the device model.
ATI3	Displays the current firmware version: x.x.x.x.x
ATI4	Displays the serial number.
ATO	Switches the command/transparent mode for the port.
ATA	Switches the transparent/command mode for the port.
ATH	Terminates (rejects) an incoming TCP connection ringing.
ATS0?	Displays the state of automatic answering settings for incoming connections.

General Commands	
ATS0=5	Sets the count of rings before automatic answer.
at^sc232at=1	Sets the AT mode upon power up.
at^sc232at=?	Displays the setting options or automatic AT mode enable.
at^sccfgall?	Displays the currently set values for all the interfaces.

Other Commands - boot, state, service, others	
at^scupg=" http://star.2n.cz/~fejfar /SmartCOM_SC_vxx.bin "	Performs firmware upgrade.
at^screstart	Restarts the given terminal.
at^scfres	Performs Factory reset.
at^scop?	Displays the GSM provider.
at^scms?	Specifies the GSM module.
at^iccid?	Displays the SIM card number (Integrated Circuit Card IDentifier).
at^scpras?	Displays the remaining PIN entering attempts.
at^sig?	Displays the signal intensity recalculated to dBm.
at^scerr?	Writes out the error buffer state.
at^scerrclear	Clears the error buffer.

7. Maintenance

This section describes how to maintain the **2N**® **SmartCom PRO** product.

Here is what you can find in this section:

- 7.1 Firmware Upgrade (Update)
- 7.2 Factory Default Reset
- 7.3 Repairs

7.1 Firmware Upgrade (Update)

The factory default firmware in **2N**® **SmartCom PRO** can be upgraded using the **2N**® **SmartCom Control Panel** or an AT command. Contact the 2N TELEKOMUNIKACE a.s. Technical Support staff for information on new functions and firmware versions.

Enter the following command into the terminal to upgrade firmware.at^scupg=" http://$(SERVER_IP):http_port/SmartCOM_SC_v00-04.bin"

The command has to include the address of the server for downloading and the exact name of the firmware file.

> ⚠️ **Caution**
>
> - Make sure that **2N**® **SmartCom PRO** is connected to the Internet during the whole process of firmware upgrading to avoid firmware installation failure.

7.2 Factory Default Reset

2N® SmartCom PRO can be factory reset. To do so, loosen the screws and remove the upper part of the case.
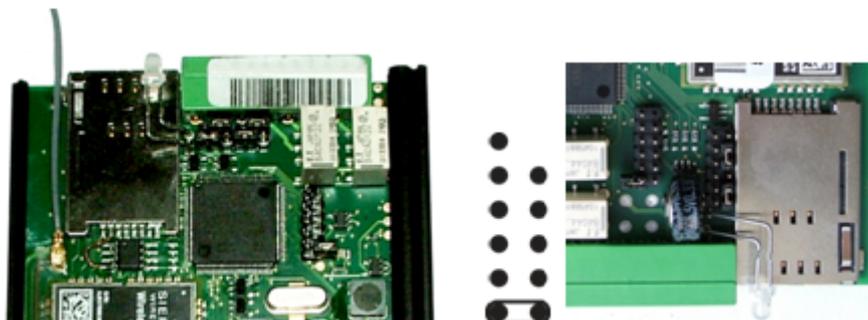
Figure: Factory Reset Jumper

Disconnect **2N® SmartCom PRO** from the power supply. Connect the jumper as shown in Fig. 11.1 (basic version on the left, PRO version on the right) and reconnect the power supply. Now the Flash memory is deleted and the factory default values are reset for all ports. Remember to remove the jumper to avoid deletion of the factory values upon every start.

Enter the following command to perform the software factory reset. The command only sets the default parameters without deleting the FLASH and EEPROM data memories.

```
at^scfres
```

- The command performs factory reset of the **2N® SmartCom PRO** device connected.

Use an SMS to initiate factory reset of the terminal too. After the SMS is sent, you will get a reply and the factory reset will be executed.

The SMS is as follows:

```
SC FRES PWD=&lt;heslo&gt;
```

- **PWD=<password>** – authentication password. Remember to set the password via AUTH_PASS as this parameter is **mandatory** for the configuration SMS!

The terminal sends the following reply:

```
Factory Reset Initialized
```

7.3 Repairs

Should you need some service interventions, please contact the 2N TELEKOMUNIKACE a.s. Technical Support staff via e-mail: **techsupport@2n.cz** or tel.: (+420)900300300.

2N® SmartCom PRO repairs may be made by an authorised servicing centre or the manufacturer only.

You can also use the error log for troubleshooting. It is a part of the memory where errors are recorded if detected. GSM connection troubles, terminal disconnection from the server and communication errors in the GSM/UMTS data channel can be detected in this way. Use the following command:

```
at^scerr?
```

The command writes out the error buffer state.

- ^SCERR: 72> 8,10
- ^SCERR: 73> 8,10

```
at^scerrclear
```

The command clears the error buffer.

Error Code Table

The table below describes the SC reset error codes in the following format:

^SCERR: {id}> {ID},{Additional value}

- id - sequence of numbers defining the error sequence after power up
- ID - error identification
- Additional value - more details and situation description
  - count of attempts left
  - CME and CMS errors - see the module datasheet
  - Command channel state - see the table below
  - Data channel state - see the table below
  - UPGRADE error code - see the table below

ID	Name	Additional value	Description
0	ERR_TYPE_NO_ERROR		No error
1	ERR_TYPE_SIM_PRESENCE		SIM not detected
2	ERR_TYPE_SIM_FEW_PIN_ATTEMPTS	PIN attempts left	Too few attempts left
3	ERR_TYPE_SIM_PIN_ERROR	CME error	Incorrect PIN
4	ERR_TYPE_SIM_BUSY		SIM is busy
5	ERR_TYPE_GSM_REGISTRATION_LOST		GSM unregistered
6	ERR_TYPE_GSM_REGISTRATION_TIMEOUT		Could not register to network
7	ERR_TYPE_COMMAND_RETRIES	Command channel state	GSM module does not respond to command channel
8	ERR_TYPE_DATA_RETRIES	Data channel state	GSM module does not respond to data channel
9	ERR_TYPE_MUX_WAIT_OPEN		MUX channel opening timeout
10	ERR_TYPE_PSPRAS_INVALID_ARG_CNT		Unexpected count of parameters (PIN attempts read)
11	ERR_TYPE_CME_COMMAND	CME error	CME error in command channel
12	ERR_TYPE_CMS_COMMAND	CMS error	CMS error in command channel

ID	Name	Additional value	Description
13	ERR_TYPE_CME_DATA	CME error	CME error in data channel
14	ERR_TYPE_CMS_DATA	CMS error	CMS error in data channel
15	ERR_TYPE_CME_UPGRADE	CME error	CME error in command channel while upgrading
16	ERR_TYPE_CMS_UPGRADE	CMS error	CMS error in command channel while upgrading
17	ERR_TYPE_CME_UPGRADE_DATA	CME error	CME error in data channel while upgrading
18	ERR_TYPE_CMS_UPGRADE_DATA	CMS error	CMS error in data channel while upgrading
19	ERR_TYPE_SC_CLIENT_UNABLE_TO_CONNECT		Could not connect to remote server
20	ERR_TYPE_SC_UPG_ERROR	UPG error code	Upgrade failed
21	ERR_TYPE_SC_UPG_BLOCK_ERROR		Did not get all requested data
22	ERR_TYPE_SC_MEMORY_ALLOCATION_ERROR		Memory allocation failed
23	ERR_TYPE_SC_SPECIFICATION_ERROR		Cannot find SC specification (serial number, etc.)
24	ERR_TYPE_PROTOCOL_RETRIES		Server not responding
25	ERR_TYPE_SC_UPG_MAXIMUM_RETRY_CNT		Upgrade failed three times
26	ERR_TYPE_MBUS_SHORT_CIRCUIT		M-Bus short circuit

ID	Name	Additional value	Description
27	ERR_TYPE_GSM_COMMAND	Command channel state	General error in command channel
28	ERR_TYPE_GSM_COMMAND_UPGRADE	Command channel state	General error in command channel while upgrading
29	ERR_TYPE_GSM_DATA	Data channel state	General error in data channel
30	ERR_TYPE_GSM_DATA_UPGRADE	Data channel state	General error in data channel while upgrading

Command Channel States

ID	Name
0	COMMAND_SUB_STATE_AT
1	COMMAND_SUB_STATE_FACTORY_RESET
2	COMMAND_SUB_STATE_FLOW_CONTROL
3	COMMAND_SUB_STATE_MUX_STARTUP
4	COMMAND_SUB_STATE_ECHO_OFF
5	COMMAND_SUB_STATE_MODUL_MANUFACTURER
6	COMMAND_SUB_STATE_MODUL_TYPE
7	COMMAND_SUB_STATE_MODUL_FW_VERSION

ID	Name
8	COMMAND_SUB_STATE_GET_IMEI
9	COMMAND_SUB_STATE_READ_OPERATOR_NAME
10	COMMAND_SUB_STATE_ERROR_REPORT
11	COMMAND_SUB_STATE_EVENT_REPORT
12	COMMAND_SUB_STATE_MESSAGE_FORMAT
13	COMMAND_SUB_STATE_SIM_PIN
14	COMMAND_SUB_STATE_SIM_PIN_ATTEMPT_STATUS
15	COMMAND_SUB_STATE_INSERT_SIM_PIN
16	COMMAND_SUB_STATE_GET_IMSI
17	COMMAND_SUB_STATE_GET_ICCID
18	COMMAND_SUB_STATE_DELETE_ALL_STORED_MESSAGES
19	COMMAND_SUB_STATE_SIGNAL_QUALITY
20	COMMAND_SUB_STATE_FIELD_STRENGTH
21	COMMAND_SUB_STATE_ASK_NETWORK_REGISTRATION
22	COMMAND_SUB_STATE_NETWORK_REGISTRATION_INDICATION
23	COMMAND_SUB_STATE_CHARACTER_SET
24	COMMAND_SUB_STATE_LIST_ALL_UNREAD_MESSAGES
25	COMMAND_SUB_STATE_SHOW_TEXT_MODE_PARAMS
26	COMMAND_SUB_STATE_NEW_MESSAGE_INDICATION

ID	Name
27	COMMAND_SUB_STATE_CLOSE_CONNECTION
28	COMMAND_SUB_STATE_OTHER
29	COMMAND_SUB_STATE_START_UP_LED_SEQUENCE
30	COMMAND_SUB_STATE_IDLE_AT
31	COMMAND_SUB_STATE_INIT_COMPLETE
32	COMMAND_SUB_STATE_SEND_RESPONSE_MESSAGE_CMD
33	COMMAND_SUB_STATE_SEND_RESPONSE_MESSAGE_TEXT
34	COMMAND_SUB_STATE_START_IP_STACK
35	COMMAND_SUB_STATE_NET_OPT_AUTO_SWITCH_ON
36	COMMAND_SUB_STATE_OPEN_GPRS_BEARER
37	COMMAND_SUB_STATE_SET_APN
38	COMMAND_SUB_STATE_SET_USER_NAME
39	COMMAND_SUB_STATE_SET_PASSWORD
40	COMMAND_SUB_STATE_START_GPRS_BEARER
41	COMMAND_SUB_STATE_GET_LOCAL_ADDRESS
42	COMMAND_SUB_STATE_GET_PEER_ADDRESS
43	COMMAND_SUB_STATE_CREATE_TCP_SERVER
44	COMMAND_SUB_STATE_CLOSE_CLIENT
45	COMMAND_SUB_STATE_WISMO_DIRECT_ACCESS

ID	Name
46	COMMAND_SUB_STATE_RESTART_CLOSE_ALL_CONNECTIONS
47	COMMAND_SUB_STATE_READ_OPERATOR_ID_START
48	COMMAND_SUB_STATE_READ_OPERATOR_ID
49	COMMAND_SUB_STATE_READ_OPERATOR_NAME_START
50	COMMAND_SUB_STATE_GSM_BANDS_GET
51	COMMAND_SUB_STATE_GSM_BANDS_SET
52	COMMAND_SUB_STATE_RTC
53	COMMAND_SUB_STATE_RTC_SAVE

Data Channel States

ID	Name
0	DATA_SUB_STATE_OTHER
1	DATA_SUB_STATE_START_DATA_EXCHANGE
2	DATA_SUB_STATE_WAIT_FOR_CONNECT
3	DATA_SUB_STATE_ECHO_OFF
4	DATA_SUB_STATE_SET_URC_TYPES
5	DATA_SUB_STATE_CLOSE_START
6	DATA_SUB_STATE_PRE_PPP_WAIT
7	DATA_SUB_STATE_SEND_PPP
8	DATA_SUB_STATE_CLOSE_AT

ID	Name
9	DATA_SUB_STATE_DATA_CHANNEL_CLOSED
10	DATA_SUB_STATE_CREATE_TCP_CLIENT
11	DATA_SUB_STATE_WAIT_FOR_CONNECTION_TO_PEER
12	DATA_SUB_STATE_TEST_AT
13	DATA_SUB_STATE_TEST_SIGNAL_QUALITY
14	DATA_SUB_STATE_TEST_READ_OPERATOR_NAME
15	DATA_SUB_STATE_TEST_SIM_CHECK
16	DATA_SUB_STATE_TEST_ASK_NETWORK_REGISTRATION
17	DATA_SUB_STATE_TEST_GET_LOCAL_ADDRESS

UPG Error Codes

Reply codes for the firmware upgrade process.

ID	Name
0	UPG_RESP_OK
1	UPG_RESP_FINISHED
2	UPG_RESP_HTTP_ERROR
3	UPG_RESP_TARGET_ERROR
4	UPG_RESP_PORT_ERROR
5	UPG_RESP_PATH_ERROR
6	UPG_RESP_TARGET_HEADER_ERROR

ID	Name
7	UPG_RESP_HW_VERSION_ERROR
8	UPG_RESP_TIMEOUT_ERROR
9	UPG_RESP_FW_MARK_ERROR
10	UPG_RESP_CRC_CHECK_ERROR
11	UPG_RESP_FW_LENGTH_ERROR
12	UPG_RESP_FW_VERSION_ERROR

8. Technical Parameters

This section provides a list of 2N® SmartCom PRO technical parameters.

Power Supply

- **DC power supply:** 5–50 V DC
- **Power input:** up to 7W
- **Connector:** terminal block for 1.5mm2 / 16 AWG

GSM / UMTS

*The used frequency depends on the module used and ground for which it is intended.

2N® SmartCom PRO uses several types of GSM/UMTS modules as ordered.

- **2G mobile network frequency:** GSM 850 MHz, EGSM 900 MHz, DCS1800 MHz, PCS 1900 MHz
- **CDMA frequency:** BC0 800 MHz, BC1 1900 MHz
- **3G mobile network frequency:** UMTS WCDMA FDD 800 MHz, 850 MHz, 900 MHz, 1900 MHz, 2100 MHz
- **SIM card:** plug-in 3 V / 1 V8 (small)
- **Transmission power:** 2 W (1W)
- **Receiver sensitivity:** -108 dBm
- **Antenna connector type:** SMA

Antenna

- **Frequency:** 850/900/1800/1900 MHz
- **Impedance:** 50 Ω
- **Maximum output power:** 2W
- **Antenna connector type:** SMA

Relays (Two)

- **Maximum voltage:** 250 V AC, 30 V DC
- **Maximum current:** 2 A
- **Maximum power:** 30 W / 62.5 VA

Input Interfaces (Two)

- **Voltage range:** 0–10 V DC
- **Current range:** 0–20 mA DC
- Digital input with threshold level setting
- S0 pulse input

RS232 Interface

- **Connector:** D-Sub 9-pin, male
- **Baud rate:** 110–230400 Baud

RS485 or M-Bus (Optional)

- **Connector:** terminal block for 1.5mm2 / 16 AWG wires
- **Count of M-Bus devices:** up to 5

Ethernet (Optional)

- **Connector:** RJ 45
- **Rate:** 100 Mbp

Wireless M-Bus (Optional)

- **Frequency:** 868 ~ 870 MHz (RC 1180)
- **Channel count:** 12
- **Data rate:** 16.384 kbps
- **Maximum output power:** 9 dBm
- **Antenna connector type:** SMA
- **Frequency:** 169.4 ~ 169.475 MHz (RC 1700)
- **Channel count:** 7
- **Data rate:** 2.4 / 4.8 / 38.4 kbps
- **Maximum output power:** 14 dBm
- **Antenna connector type:** SMA

ZigBee (Optional)

- **Frequency:** 2.400 ~ 2.4835 GHz
- **Channel count:** 16
- **Data rate:** 250 kbps
- **Maximum output power (RC2400):** 4 dBm
- **Maximum output power (RC2400HP):** 20 dBm*
- **Flash memory:** 256 kB
- **Antenna connector type:** SMA

> ⓘ **Note**
>
> * Limited to 12 dBm in Europe according to ETSI and to 10 dBm in the US for 2480MHz frequency according to FCC.

Data Memory

- **EEPROM:** 120 kB (basic)
- **FLASH:** 8 MB (optional)

Backup Power Supply

- **Accumulator type:** 6 or 12V gel-lead-acid (set the parameters before connection)
- **Charging current:** up to 350 mA

Real Time Clock

- **Power failure back-up:** 0.33F super capacitor, 2-day back-up capacity

Working Conditions

- **Temperature:** - 40°C ÷ 85°C
- **Relative humidity:** 0 ÷ 80%

Others

- **Dimensions (w/o connectors):** 105 x 100 x 30 mm
- **KNX RF, RC 232, TinyMesh**
- **CE certification**

9. Supplementary Information

This section provides supplementary information on **2N**® **SmartCom PRO**.

Here is what you can find in this section:

- 9.1 List of Abbreviations
- 9.2 Regulations, Laws and Directives
- 9.3 General Instructions and Cautions

2N

9.1 List of Abbreviations

- **APN** (Access Point Name) Name of GPRS access point.
- **CSD** (Circuit Switched Data) Circuit switched data transmission.
- **GSM** (Group Switched Mobile system) Current standard for mobile telephone systems.
- **GPRS** (General Packet Radio Service) High-speed packet oriented mobile data service.
- **HW** (Hardware) For this purpose, hardware means an electronic device, circuit, board, component, etc..
- **PIN** (Personal Identification Number) SIM card protecting code against unauthorised use.
- **PUK** (Personal Unblocking Key) SIM card unblocking code after repeated wrong PIN entering attempts.
- **SIM** (Subscriber Identity Module) Chip module for GSM identification.
- **FW** (Firmware) Software responsible for system function.
- **SW** (Software)

Europe

2N® SmartCom PRO conforms to the following directives and regulations:

Directive 1999/5/EC of the European Parliament and of the Council, of 9 March 1999 - on radio equipment and telecommunications terminal equipment and the mutual recognition of their conformity

Directive 2006/95/EC of the European Parliament and of the Council of 12 December 2006 on the harmonisation of the laws of Member States relating to electrical equipment designed for use within certain voltage limits

Directive 2004/108/EC of the Council of 15 December 2004 on the harmonisation of the laws of Member States relating to electromagnetic compatibility

Commission Regulation (EC) No. 1275/2008, of 17 December 2008, implementing Directive 2005/32/EC of the European Parliament and of the Council with regard to ecodesign requirements for standby and off mode electric power consumption of electrical and electronic household and office equipment

Directive 2011/65/EU of the European Parliament and of the Council of 8 June 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment

Regulation (EC) No. 1907/2006 of the European Parliament and of the Council of 18 December 2006 concerning the Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH), establishing a European Chemicals Agency, amending Directive 1999/45/EC and repealing Council Regulation (EEC) No. 793/93 and Commission Regulation (EC) No. 1488/94 as well as Council Directive 76/769/EEC and Commission Directives 91/155/EEC, 93/67/EEC, 93/105/EC and 2000/21/EC

Directive 2012/19/EC of the European Parliament and of the Council of 4 July 2012 on waste electrical and electronic equipment.

9.3 General Instructions and Cautions

Please read this User Manual carefully before using the product. Follow all instructions and recommendations included herein.

Any use of the product that is in contradiction with the instructions provided herein may result in malfunction, damage or destruction of the product.

The manufacturer shall not be liable and responsible for any damage incurred as a result of a use of the product other than that included herein, namely undue application and disobedience of the recommendations and warnings in contradiction herewith.

Any use or connection of the product other than those included herein shall be considered undue and the manufacturer shall not be liable for any consequences arisen as a result of such misconduct.

Moreover, the manufacturer shall not be liable for any damage or destruction of the product incurred as a result of misplacement, incompetent installation and/or undue operation and use of the product in contradiction herewith.

The manufacturer assumes no responsibility for any malfunction, damage or destruction of the product caused by incompetent replacement of parts or due to the use of reproduction parts or components.

The manufacturer shall not be liable and responsible for any loss or damage incurred as a result of a natural disaster or any other unfavourable natural condition.

The manufacturer shall not be held liable for any damage of the product arising during the shipping thereof.

The manufacturer shall not make any warrant with regard to data loss or damage.

The manufacturer shall not be liable and responsible for any direct or indirect damage incurred as a result of a use of the product in contradiction herewith or a failure of the product due to a use in contradiction herewith.

All applicable legal regulations concerning the product installation and use as well as provisions of technical standards on electric installations have to be obeyed. The manufacturer shall not be liable and responsible for damage or destruction of the product or damage incurred by the consumer in case the product is used and handled contrary to the said regulations and provisions.

The consumer shall, at its own expense, obtain software protection of the product. The manufacturer shall not be held liable and responsible for any damage incurred as a result of the use of deficient or substandard security software.

The consumer shall, without delay, change the access password for the product after installation. The manufacturer shall not be held liable or responsible for any damage incurred by the consumer in connection with the use of the original password.

The manufacturer also assumes no responsibility for additional costs incurred by the consumer as a result of making calls using a line with an increased tariff.

Electric Waste and Used Battery Pack Handling

Do not place used electric devices and battery packs into municipal waste containers. An undue disposal thereof might impair the environment!

Deliver your expired electric appliances and battery packs removed from them to dedicated dumpsites or containers or give them back to the dealer or manufacturer for environmental-friendly disposal. The dealer or manufacturer shall take the product back free of charge and without requiring another purchase. Make sure that the devices to be disposed of are complete.

Do not throw battery packs into fire. Battery packs may not be taken into parts or short-circuited either.

2N TELEKOMUNIKACE a.s.

Modřanská 621, 143 01 Prague 4, Czech Republic

Phone: +420 261 301 500, Fax: +420 261 301 599

E-mail: sales@2n.cz

Web: www.2n.cz

v1.11.3