

# Running AT Commands Remotely Application Note

80000NT10029a Rev.3 – 2010-10-04



## APPLICABILITY TABLE

PRODUCT
GT863-PY
GT864-QUAD
GT864-PY
GM862-QUAD-PY
GM862-QUAD
GM862-GPS
GC864-PY
GC864-QUAD
GC864-PY w/ SIM holder
GC864-DUAL
GE863-PY
GE863-QUAD
GE863-GPS
GE863-SIM
GE863-PRO <sup>3</sup>
GE864-PY
GE864-QUAD
GE864-QUAD V2
GE864-DUAL V2
GE864-QUAD AUTOMOTIVE
GE864-QUAD AUTOMOTIVE V2
GE864-QUAD ATEX
GE865-QUAD
GL865-DUAL



**Running AT commands remotely Application Note**  
80000NT10029a Rev.3 – 2010-10-04

**SW Version**

**7.03.02 / 7.02.07**

**10.0x.xx2**



# Contents

- 1. INTRODUCTION ..... 6**
  - 1.1. CONTACT INFORMATION, SUPPORT.....6
  - 1.2. TEXT CONVENTIONS.....7
  - 1.3. RELATED DOCUMENTS.....7
  - 1.4. DOCUMENT HISTORY .....7
- 2. SMSATRUN..... 8**
  - 2.1. REQUIREMENTS .....8
  - 2.2. ALERTS.....9
  - 2.3. DESCRIPTION.....10
  - 2.4. EXAMPLE SMS ATRUN.....15
  - 2.5. DELAYS AND NOTES.....17
    - 2.5.1. *Delays*.....17
    - 2.5.2. *Notes*.....18
- 3. TCPATRUN ..... 20**
  - 3.1. CONFIGURATION .....21
  - 3.2. TCPATRUN IN SERVER MODE .....22
    - 3.2.1. *Description*.....22
    - 3.2.2. *Authentication mode*.....25
  - 3.3. TCPATRUN IN CLIENT MODE.....26
    - 3.3.1. *Description*.....26
  - 3.4. AT COMMANDS IN SEQUENCE.....28
  - 3.5. TRANSPARENCE.....29
  - 3.6. ALERTS.....29
  - 3.7. EXAMPLES.....31
    - 3.7.1. *TCPATRUN in server mode and AT commands*.....31
    - 3.7.2. *TCPATCMDSEQ feature*.....32
  - 3.8. DELAYS AND NOTES.....33
    - 3.8.1. *Delays*.....33
    - 3.8.2. *Notes*.....34
  - 3.9. TCPATRUN AND PYTHON .....35



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

### DISCLAIMER

The information contained in this document is the proprietary information of Telit Communications S.p.A. and its affiliates ("TELIT"). The contents are confidential and any disclosure to persons other than the officers, employees, agents or subcontractors of the owner or licensee of this document, without the prior written consent of Telit, is strictly prohibited.

Telit makes every effort to ensure the quality of the information it makes available. Notwithstanding the foregoing, Telit does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information.

Telit disclaims any and all responsibility for the application of the devices characterized in this document, and notes that the application of the device must comply with the safety standards of the applicable country, and where applicable, with the relevant wiring rules.

Telit reserves the right to make modifications, additions and deletions to this document due to typographical errors, inaccurate information, or improvements to programs and/or equipment at any time and without notice. Such changes will, nevertheless be incorporated into new editions of this application note.

Copyright: Transmittal, reproduction, dissemination and/or editing of this document as well as utilization of its contents and communication thereof to others without express authorization are prohibited. Offenders will be held liable for payment of damages. All rights are reserved.

Copyright © Telit Communications SpA 2010.



## 1. Introduction

The information presented in this document is believed to be accurate and reliable. However, no responsibility is assumed by Telit Communications S.p.A. for its use, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Telit Communications S.p.A. other than for circuitry embodied in Telit products. This document is subject to change without notice.

### 1.1. Contact Information, Support

For general contact, technical support, to report documentation errors and to order manuals, contact Telit Technical Support Center (TTSC) at:

[TS-EMEA@telit.com](mailto:TS-EMEA@telit.com)  
[TS-NORTHAMERICA@telit.com](mailto:TS-NORTHAMERICA@telit.com)  
[TS-LATINAMERICA@telit.com](mailto:TS-LATINAMERICA@telit.com)  
[TS-APAC@telit.com](mailto:TS-APAC@telit.com)

Alternatively, use:

<http://www.telit.com/en/products/technical-support-center/contact.php>

For detailed information about where you can buy the Telit modules or for recommendations on accessories and components visit:

<http://www.telit.com>

To register for product news and announcements or for product questions contact Telit Technical Support Center (TTSC).

Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.

Telit appreciates feedback from the users of our information.



## 1.2. Text Conventions



***Danger – This information MUST be followed or catastrophic equipment failure or bodily injury may occur.***



***Caution or Warning – Alerts the user to important points about integrating the module, if these points are not followed, the module and end user equipment may fail or malfunction.***



***Tip or Information – Provides advice and suggestions that may be useful when integrating the module.***

All dates are in ISO 8601 format, i.e. YYYY-MM-DD.

## 1.3. Related Documents

- AT Commands Reference guide, 80000ST10025a

## 1.4. Document History

Revision	Date	Changes
ISSUE #0	2009-08-28	Release First ISSUE# 0
ISSUE #1	2010-05-07	Added commands in paragraph 3.9.2 (Notes) Correction of delays table regarding AT+CFUN command in paragraph 3.9.1 Correction in the pictures in paragraph 2.3 "Alerts" moved from paragraph 3.3.1 to paragraph 3.6 and integrated with a note about Idle Activity Timeout
ISSUE #2	2010-05-18	Deletion of note 15 in par. 3.9.2 Notes Shifted paragraph 3.9 in par. 2.5, renamed Delays and Notes. Added par. 3.9 regarding Python use. Changed the images in par 2.3 and in par. 3.1 (before 3.1). Added notes about ATZ command and about Packet Size
ISSUE#3	2010-10-04	Added GL865-DUAL to the applicability table

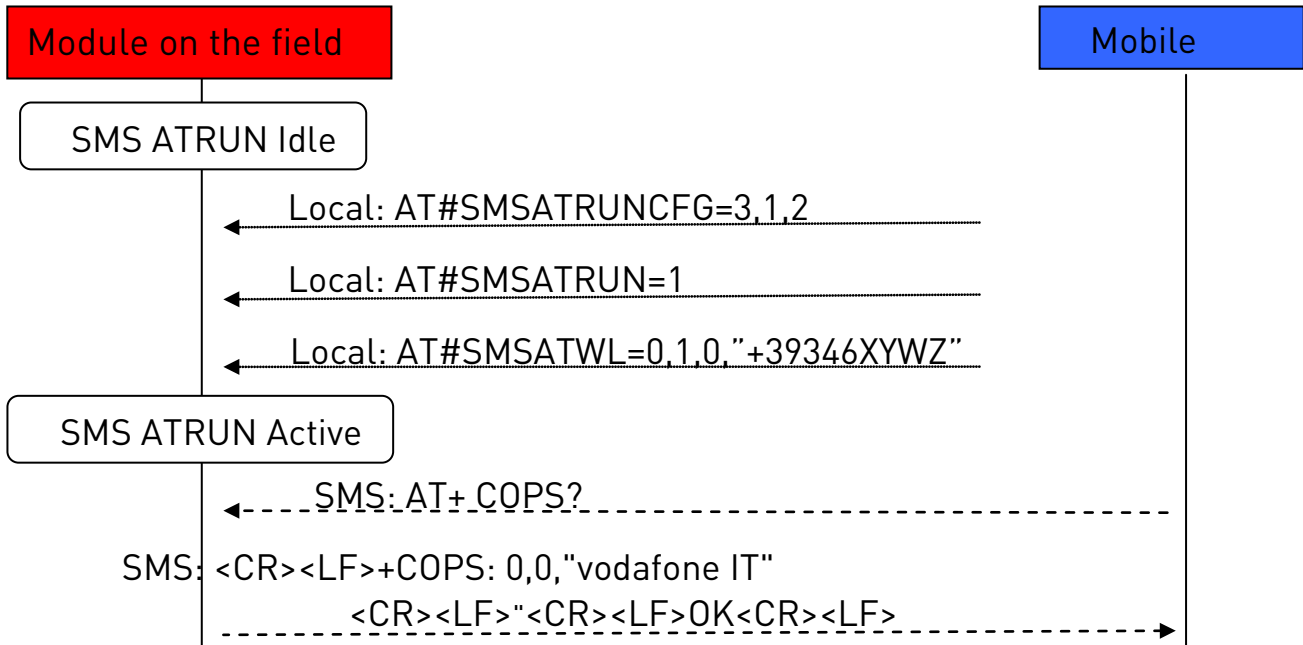








Running AT commands remotely Application Note  
80000NT10029a Rev.3 – 2010-10-04



First of all, the SMS ATRUN service is locally configured and enabled in the *module on the field* by the command #SMSATRUNCFG and :

**AT#SMSATRUNCFG=3,1,2**

// the service is configured to run on the third instance, the unsolicited is enabled and  
// the timeout for the response to the AT commands is set to 2 minutes

**AT#SMSATRUN=1** // the service is enabled

and the *mobile* number is added in the white list by the command #SMSATWL:

**AT#SMSATWL=0,1,0,\"+39346XYWZ\"** //the number +39346XYWZ is added in the  
// first position of the white list

At this point the SMS ATRUN service is enabled and AT commands can be issued to the *module on the field* via SMS from another device containing the SIM with the number added in the white list.



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

For example another *mobile* can set the GPIO of the *module on the field* issuing the command AT#GPIO=2,1 using the following standard AT command

```
AT+CMGS=346XYZWZ
> AT#GPIO=2,1
```

The *module on the field* let run the AT+COPS? on the third instance and will send back to the *mobile* an SMS with the response. The *mobile* reads the arrived SMS:

```
+CMTI: "SM",1
AT+CMGR=1
+CMGR: "REC UNREAD","+39346XYZWZ",",", "08/05/28,11:04:37+08"
```

```
+COPS: 0,0,"vodafone IT"
```

OK

If the *mobile* doesn't want to receive back the response, it can issue the following command:

```
AT+CMGS=346XYZWZ
> HAT+COPS?
```

If the *mobile* wants to communicate to the *module on the field* using the Digest ATRUN SMS, a common password has to be inserted locally in the white list of the *module on the field* by the command AT#SMSATWL:

```
AT#SMSATWL=0,2,1,"1234567890123456" //the password has to be 16
// characters length and is
// inserted, in this example, in
// position 2 of the white list
```

A maximum number of 2 passwords can be inserted in the white list.

Now the *mobile* can also change the SIM because the sender number isn't checked from *module on the field* in the case of Digest ATRUN SMS.

So the *mobile* can interrogate the *module on the field*, to know the selected network operator, issuing the command AT+COPS? using the following standard AT command

```
AT+CMGS=53
>079193432900200011000C919343868676110015AD26D0D0D000110157696A437032715633734E7
62B61727532305A5561773D3D41542B434F50533F
```

Where the User Data portion is formatted according to Table 1 as follows:



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

Offset	Size	Value	Description
0	3	D0D0D0	RUNAT SMS Code
3	1	00	Transaction Id
4	1	11	Segment 1 of 1
5	1	01	Session Id
6	24	57696A437032715633734E762B61727532305A 5561773D3D	Digest: B64(MD5(B64(MD5(Pwd))):B64(MD5(AT+COPS?))))
30		41542B434F50533F	AT+COPS?

The *module on the field* let run the AT+COPS? on the third instance and will send back to the *mobile* an SMS with the response. The *mobile* reads the arrived SMS:

```
+CMTI: "SM",1
AT+CMGR=1
+CMGR: 0,"",83
0791934329002000040C9193438686761100158050922102818040D1D1D10011014A756E726137723
06B494563483971563534326E33513D3D0D0A2B434F50533A20302C302C22766F6461666F6E6520
4954220D0A0D0A4F4B0D0A
```

Where the User Data portion is formatted according to Table 2 as follows:

Offset	Size	Value	Description
0	3	D1D1D1	Response RUNAT SMS Code
3	1	00	Transaction Id
4	1	11	Segment 1 of 1
5	1	01	Session Id
6	24	4A756E72613772306B494563483 971563534326E33513D3D	Digest: B64(MD5(B64(MD5(Pwd))):B64(MD5(<CR><LF>+COPS: 0,0,"vodafone IT"<CR><LF>"<CR><LF>OK<CR><LF>>))))
30		0D0A2B434F50533A20302C302C 22766F6461666F6E65204954220 D0A0D0A4F4B0D0A	<CR><LF>+COPS: 0,0,"vodafone IT"<CR><LF>"<CR><LF>OK<CR><LF>>



Running AT commands remotely Application Note  
80000NT10029a Rev.3 – 2010-10-04



---

**ALERTS:**

Some AT commands can fail if set through SMS ATRUN service, due to an interaction of the AT command execution with the SMS service. These problems can be avoided inserting a delay before AT command execution (e.g. AT+COPS=? using SMS ATRUN).

This delay is set through the command AT#ATRUNDELAY:

**AT#ATRUNDELAY=1,5** sets for SMS ATRUN commands a delay of 5 seconds

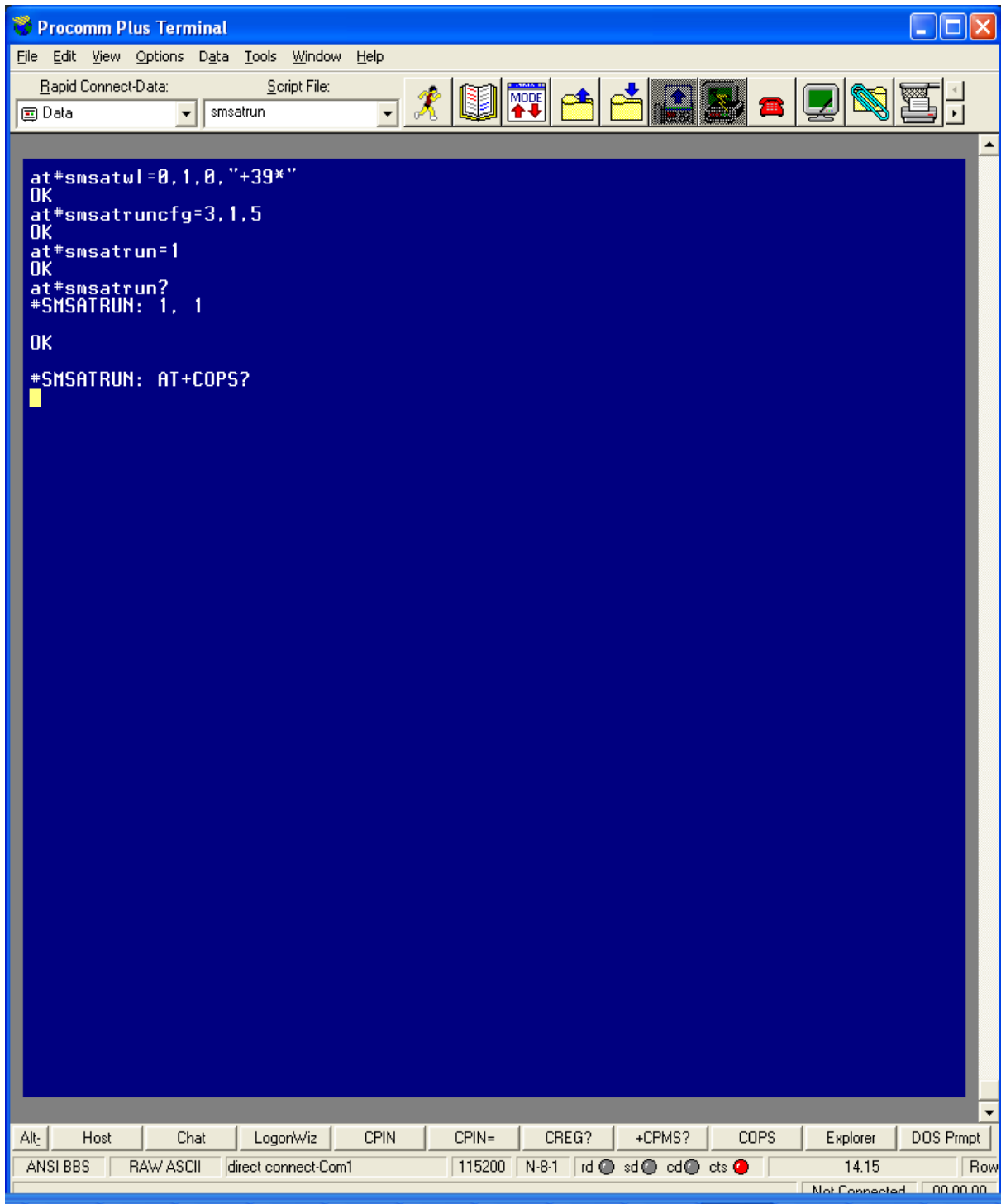
Once set, the delay is valid till a new AT#ATRUNDELAY is issued.

Some other commands can block the AT command sending-AT command response receiving mechanism if set through ATRUN service, see below.

---



## 2.4. Example SMS ATRUN



```

at#smsatwl=0,1,0,"+39*"
OK
at#smsatruncfg=3,1,5
OK
at#smsatrun=1
OK
at#smsatrun?
#SMSATRUN: 1, 1

OK
#SMSATRUN: AT+COPS?

```

Figure 1: Module in the field





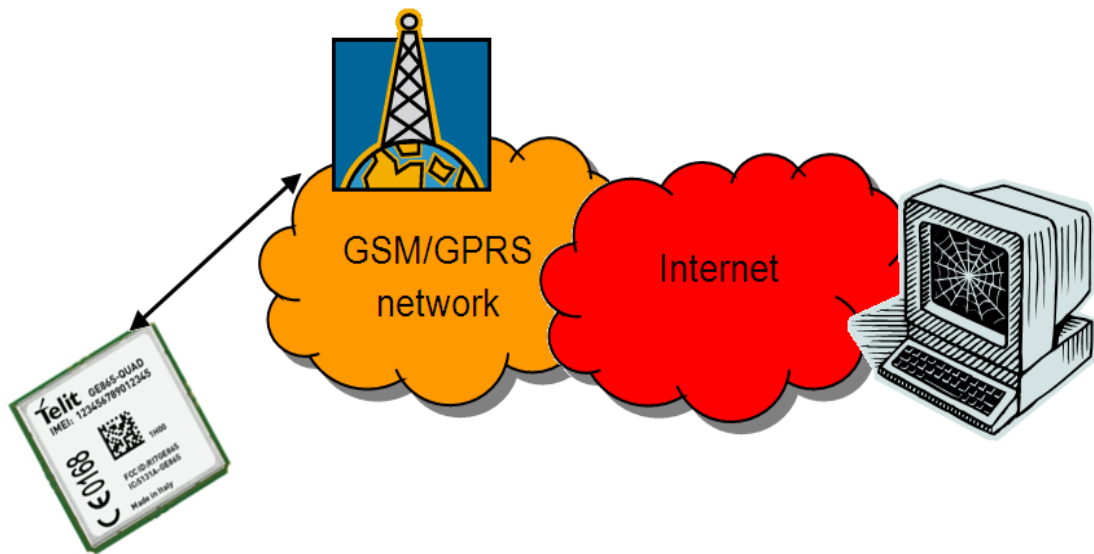






### 3. TCPATRUN

AT command execution can also be requested via TCP. The AT interface input and output will be redirected to the TCP socket.



There are two kinds of TCP ATRUN service, differing the one from the other in the way the module connects to the remote PC.

In the first kind of TCP ATRUN, the module acts as a server, that is to say it is in TCP listen state, waiting for an incoming TCP/IP connection from a client.

In the second one, the module acts as a client, that is to say it tries, as soon as the service is enabled, to connect to a server.

In next paragraphs we describe the AT command sequence to start the TCP ATRUN service and how the service behaves. Please refer to the AT commands specification for the exact commands syntax.



### 3.1. Configuration

To start TCPATRUN service, either in server mode or in client mode, it is necessary that a PDP context is opened, through the command AT#SGACT:

**AT#SGACT=1,1** for example, here we open the PDP context with id 1

The answer, in case of success, will be the IP address of the module:

**#SGACT: "212.141.113.92"**

It is recommended to use the command "AT#SGACTCFG" to set the context "auto-activation": in this way the context will be automatically activated in case of switching off/on, in case of deactivation from Network and in case of SIM removal.

**AT#SGACTCFG=1,1** the auto-activation is set on context 1

It is suggested to configure the socket parameters through the command AT#SCFG (please read the paragraph "Alerts"), using the socket identifier (the first parameter). For example:

**AT#SCFG=1,1,300,0,600,1**

Now, some configuration parameters have to be set for both type of service, through the command AT#TCPATRUNCFG:

**AT#TCPATRUNCFG =1,2,1024,12345,`"212.141.125.127`",1,5,1,5,2**

The first parameter is the identifier of the socket that will be used, the second one is the instance (in the module the AT interface can use three instances) that will be dedicated to the TCP ATRUN, the third one is the TCP port for listen (used in server mode), the fourth one and the fifth one are respectively the TCP port and IP address of the host (for the client mode), the sixth one is for enabling the unsolicited that advises of the connection/disconnection from server, the seventh one is the timeout for an AT command execution, the eighth one is for choosing the authentication mode (in server mode), the ninth one is the number of attempts to connect that the module will do in case of automatic re-start of the service (in client mode), the last one is the delay (in minutes) from one attempt and the other (in client mode).



## 3.2. TCPATRUN in server mode

### 3.2.1. Description

The TCP ATRUN service in server mode is enabled locally by the AT command `AT#TCPATRNL`.

This command puts the module in listen state on a determined port, so that a client can connect from remote opening a TCP socket to the IP address and port of the module. Once connected, the client can issue AT commands to the module.

Note that the TCP ATRUN service in server mode is useful with static IP address.

The TCP access to the module is controlled by a firewall.

In the case of TCP ATRUN service, this firewall can be set (and saved in memory) by the following command

```
AT#TCPATRUNFRWL=1,"212.141.112.216","255.255.255.255"
```

The two strings represent the IP address and mask

The service is also protected by the authentication parameters: Username and Password.

These parameters are set (and saved in memory) through the following command

```
AT#TCPATRUNAUTH=1,"testuser","testpassw"
```

The user can choose between two kinds of authentication procedure, as we will see in the next paragraph.

Finally, the TCP ATRUN service can be enabled using "AT#TCPATRNL":

```
AT#TCPATRNL =1
```

Now, referring to the parameters configuration that has been set in this example through the command `AT#TCPATRUNCFG`:

the socket 1 is in listen state on port 1024, and the IP address is the one given by the `AT#SGACT` command.

The client can connect to the module if its address is included in the firewall.

It can use Telnet or a dedicated program to open the socket and communicate with the module. Once connected, username and password have to be transmitted in a way that depends on the authentication mode set with `AT#TCPATRUNCFG`.



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

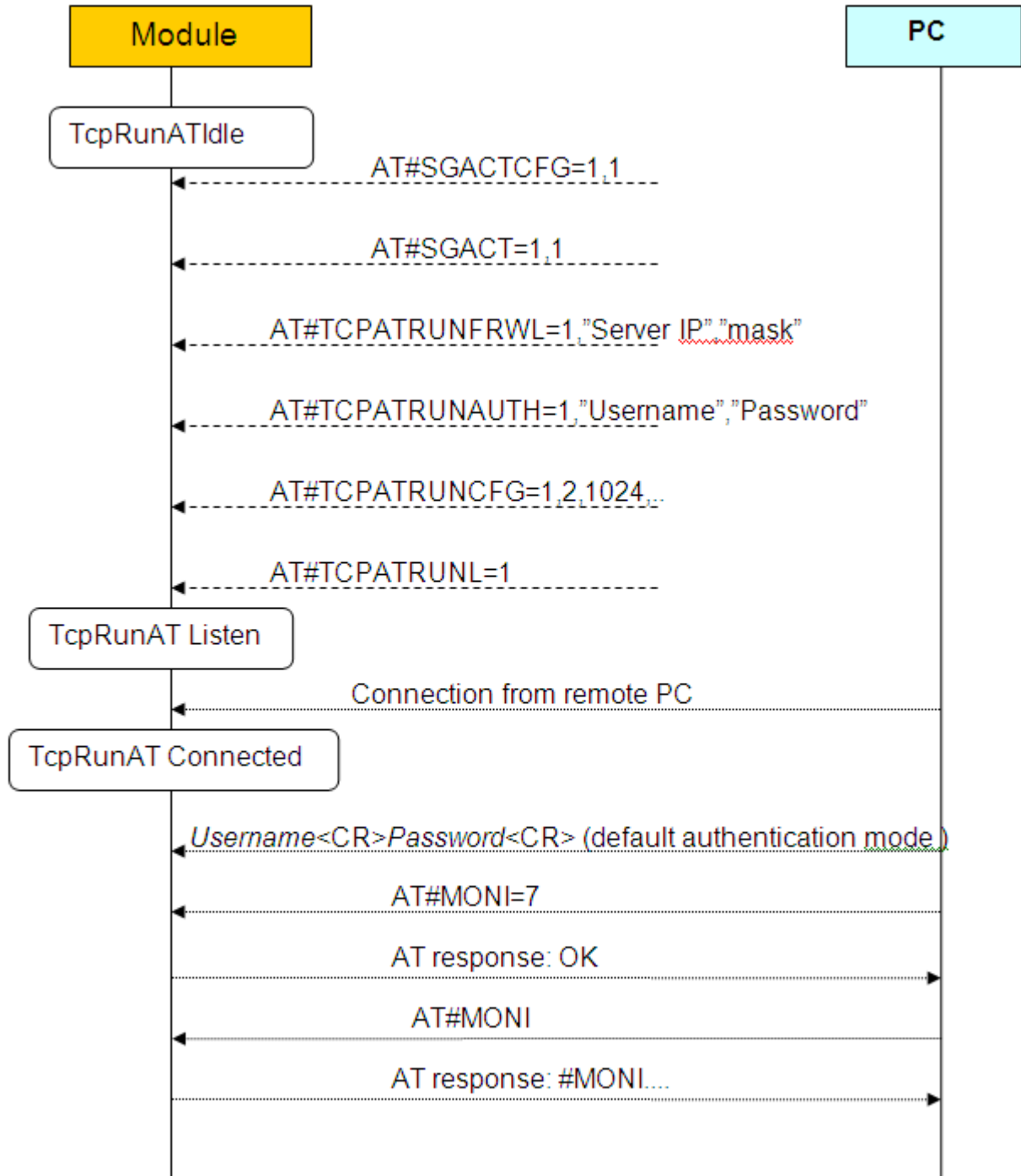
After authentication, an AT command can be issued and the module will parse the command and send the response.

For example, client can interrogate the module about all the cells in the neighbour of the serving cell, by the commands AT#MONI=7 and AT#MONI.

If the TCP ATRUN service is not disabled, it automatically starts at every start-up of the module as long as the PDP context auto-activation has been set by AT#SGACTCFG. The same happens in case of a deactivation from Network and in case of SIM removal.



Running AT commands remotely Application Note  
80000NT10029a Rev.3 – 2010-10-04



## 3.2.2. Authentication mode

Setting the *authentication mode* parameter in the command AT#TCPATRUNCFG, the user can choose between two kinds of authentication: “automatic” and “step-by-step”. The first one is useful for easily automate the procedure, the second one is more graphically pleasant.

### 3.2.2.1. Automatic mode (default)

Once the connection between server and module is opened and before issuing any AT command, username and password have to be transmitted in the following way:

*userid*<CR>*passw*<CR> (where <CR> means Carriage Return and is equal to 0x0d)

After these strings, AT commands can be sent to the module.

The couple *Username* – *Password* has to be included in the authentication parameters list built through the command AT#TCPATRUNAUTH, otherwise the module will immediately close the connection.

If a timeout expires in the module without receiving one of the two parameters, then the module will close the connection.

### 3.2.2.2. Step-by-step

Once the connection between server and module is opened, the module first requests the Username and then, if the Username has been received and accepted (*userid*<CR>), requests the Password. Once the Password has been received and accepted (*passw*<CR>), the module transmits a “Login successful” message.

Username: *userid*

Password: *passw*

Login successful.

From this moment AT commands sent by the server are parsed by the module.

Also in this case, if the couple *Username* – *Password* is not included in the authentication parameters list, or if the timeout expires without receiving a parameter, the module will close the connection.



### 3.3. TCPATRUN in client mode

#### 3.3.1. Description

This TCP ATRUN service is enabled locally by the AT command AT#TCPATRUND:

**AT#TCPATRUND = 1**

This command opens a TCP socket to connect to the Host IP address and to the Host port specified in AT#TCPATRUNCFG.

To allow the module to connect, the server (the remote PC) has to have been put in listen through a dedicated program at the port specified with AT#TCPATRUNCFG command (in the previous example 12345).

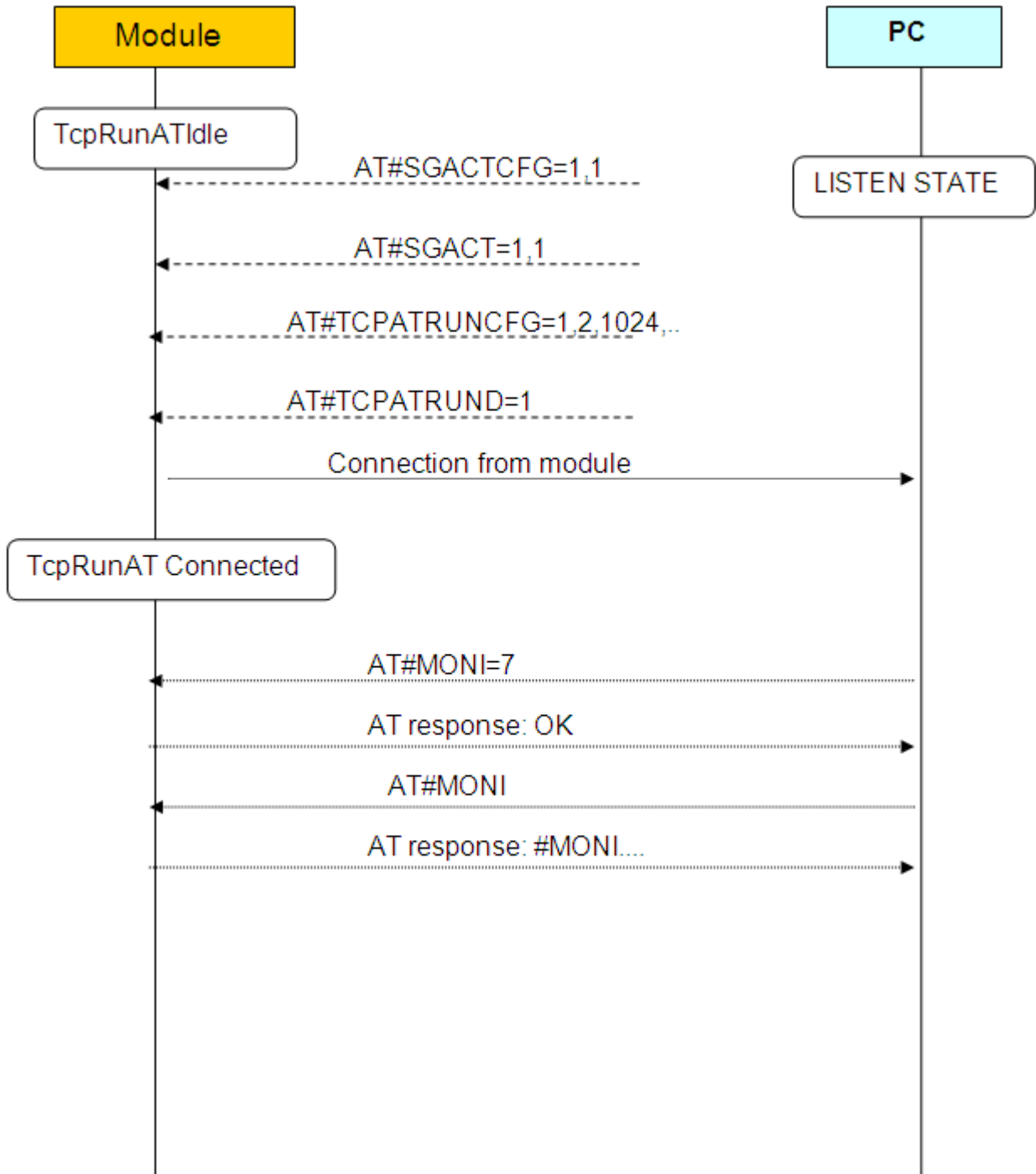
If the response to AT#TCPATRUND command is "OK", the module is connected to the remote PC and ready to receive AT commands.

If the TCP ATRUN service is not disabled, it automatically starts at every start-up of the module as long as the PDP context auto-activation has been set through AT#SGACTCFG.

The same happens in case of a deactivation from Network and in case of SIM removal. This means that the module will automatically try to connect to the Host IP (the one set through AT#TCPATRUNCFG), and if connection is not successful, it will retry. The number of attempts and the delay between one and the other will be those set through AT# TCPATRUNCFG.



Running AT commands remotely Application Note  
80000NT10029a Rev.3 - 2010-10-04



### 3.4. AT commands in sequence

The standard usage of AT interface consists in waiting for response to an AT command before issuing another one. Each command ends with the delimiter character (<CR> is the default). In alternative, AT commands can be issued separated by “;”, being treated like a single command. For example, the response to

```
AT# SGACT=1,1;#SS
```

could be

```
#SGACT: 212.141.252.44
#SS: 1,1,212.141.121.05, 1024,212.141.121.148,1332
#SS: 2,0
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
```

OK

What happens if the remote application doesn't wait for an AT command response before issuing another one?

If AT commands are issued with the delimiter character but without waiting for response, some commands can be ignored, for example in the sequence

```
AT#SGACT=1,1
AT#SS
```

The AT#SS would be ignored.

In the case of AT commands issued through TCP ATRUN service, a new feature has been introduced that solves this problem.

This feature is enabled through the command

**AT#TCPATCMDSEQ=1.**

If enabled, the response to the sequence

```
AT#SGACT=1,1
AT#SS
```

could be for example

```
#SGACT: 212.141.252.44
```



OK

```
#SS: 1,1,212.141.121.05, 1024,212.141.121.148,1332
#SS: 2,0
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
```

OK

The command AT#TCPATCMDSEQ can be set from any instance, but it just affects AT commands issued through ATRUN service.

### 3.5. Transparency

From TCP ATRUN service, it is possible to connect directly to a serial port of the module, in transparent mode, that is to say that data will be transferred directly between TCP ATRUN and the serial port specified. So any data in input on the serial port will be sent to the remote PC via TCP/IP (and vice-versa). This connection is opened issuing the command AT#TCPATCONSER=<port>,<rate> from the TCP ATRUN instance.

After this command has been issued, if no error has occurred, then a “CONNECT” will be returned from module to advise that the TCP ATRUN instance is in *online mode* and connected to the port specified.

To exit from online mode and close the connection, the escape sequence (the default is “+++”) has to be sent on the TCP ATRUN instance, taking into account that the escape sequence needs to be sent in one single packet. (So the use of Telnet for Windows, sending every single byte in a TCP packet, is not appropriate in this case).

### 3.6. Alerts

For the connection used by the TCP ATRUN service, a particular socket identifier is used (see the first parameter of AT#TCPATRUNCFG). So all the settings on that socket can affect the connection: please refer to the module Easy GPRS User Guide for the use of the AT commands concerning TCP sockets.

In particular, by the command AT#SCFG, the user can read and change the packet size, the Transmission Timeout and the Idle Activity Timeout.



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

To summarize briefly the meaning of the Transmission Timeout: if the bytes to be sent from the module are less than packet size, then they will be sent just at Transmission Timeout expiry. So in the case of TCP ATRUN service, it is suggested to set the minimum Transmission Timeout to avoid to wait for responses. It is strongly recommended to avoid having it set to infinite.

It is also recommended not to set a Packet Size smaller than the default one.

The meaning of the Idle Activity Timeout is this: if there is no data exchange within the Idle Activity Timeout period, the connection is closed. This is useful in the following situation: if the TCP/IP socket is closed by the remote PC in an "unilateral" way, that is to say without signal exchange with the other side of the connection, on the local side (the module on the field) the connection is still up, but cannot be used anymore.

In this situation, if the Idle Activity Timeout has been set locally on the module, it will automatically close the connection, otherwise the only way to close it is using the AT command AT#TCPATRUNCLOSE locally.

After the connection local closure, the TCP ATRUN status is still enabled, so the service re-starts automatically.

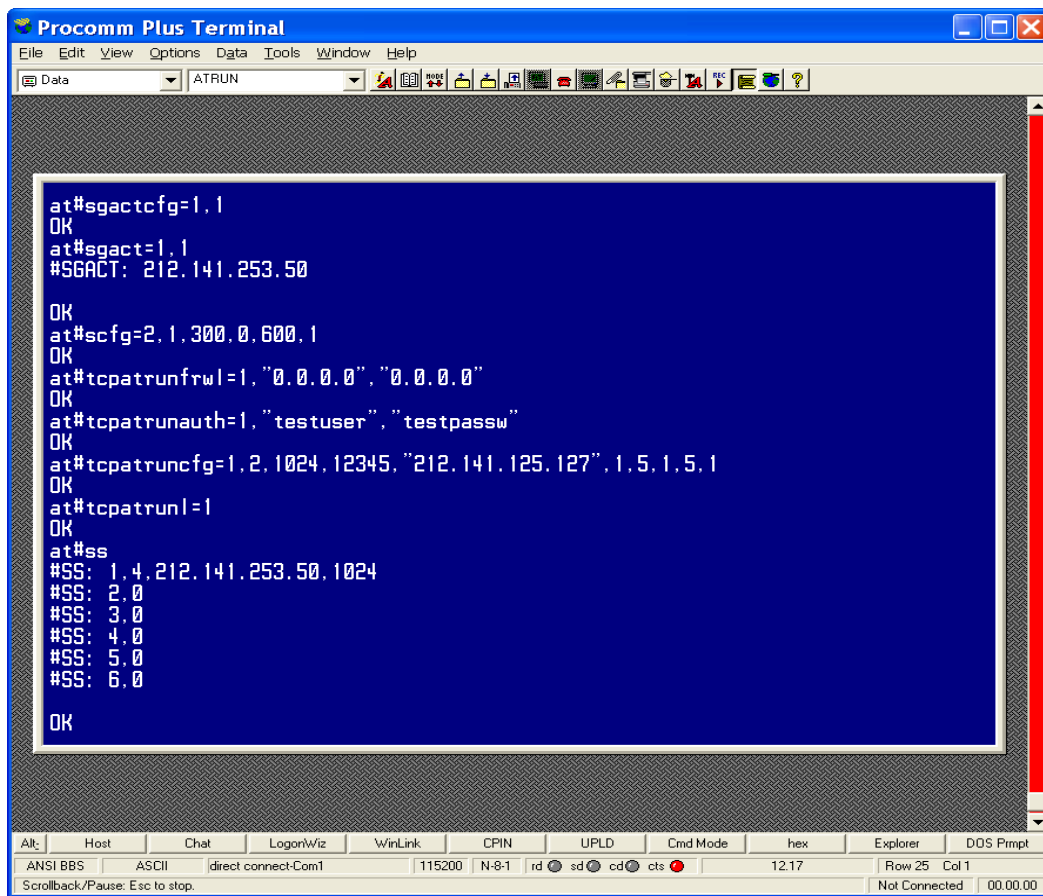
If the TCPATRUN service, in server or client mode, is activated on the first instance (<muxInstance> = 1 in AT#TCPATRUNCFG), the control of that instance will be loosed and there will be the need of the multiplexer to deactivate the service by the second or third instance.



### 3.7. Examples

#### 3.7.1. TCP ATRUN in server mode and AT commands

##### Remote Mobile



```

Procomm Plus Terminal
File Edit View Options Data Tools Window Help
Data ATRUN
at#sgactcfg=1,1
OK
at#sgact=1,1
#SGACT: 212.141.253.50
OK
at#scfg=2,1,300,0,600,1
OK
at#tcpatrunfrwl=1,"0.0.0.0","0.0.0.0"
OK
at#tcpatrunauth=1,"testuser","testpassw"
OK
at#tcpatruncfg=1,2,1024,12345,"212.141.125.127",1.5,1.5,1
OK
at#tcpatrunl=1
OK
at#ss
#SS: 1,4,212.141.253.50,1024
#SS: 2,0
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
OK
Alt Host Chat LogonWiz WinLink CPIN UPLD Cmd Mode hex Explorer DOS Prmpt
ANSI BBS ASCII direct connect-Com1 115200 N-8-1 rd sd cd cts 12.17 Row 25 Col 1
Scrollback/Pause: Esc to stop. Not Connected 00:00:00
  
```



## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04

### PC ( telnet )

```

C:\ Telnet 212.141.112.216
Username:
testuser
Password:
testpassw
Login successful.
at+creg?

+CREG: 0,1

OK
at#moni=7

OK
at#moni

#MONI: Cell BSIC LAC CellId ARFCN Power C1 C2 TA RxQual PLMN
#MONI: S 75 55FA 1297 983 -72dbm 32 32 1 0 I WIND
#MONI: N1 20 D5BD 5265 59 -70dbm 38 0
#MONI: N2 70 55FA 0000 979 -76dbm 28 23
#MONI: N3 72 55FA 1281 975 -81dbm 23 17
#MONI: N4 70 55FA 12EA 985 -85dbm 19 13
#MONI: N5 72 55FA 1D0D 751 -88dbm 14 22
#MONI: N6 73 55FA 1D23 754 -92dbm 10 19

OK
  
```

### 3.7.2. TCPATCMDSEQ feature

```

C:\ Telnet 212.141.121.85
Username:
testuser
Password:
testpassw
Login successful.
at

OK
at#sgactcfg=2,1
at#sgact=2,1
at#ss

OK

#SGACT: 212.141.252.44

OK

#SS: 1,1,212.141.121.85,1024,212.141.121.148,1332
#SS: 2,0
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0

OK
-
  
```



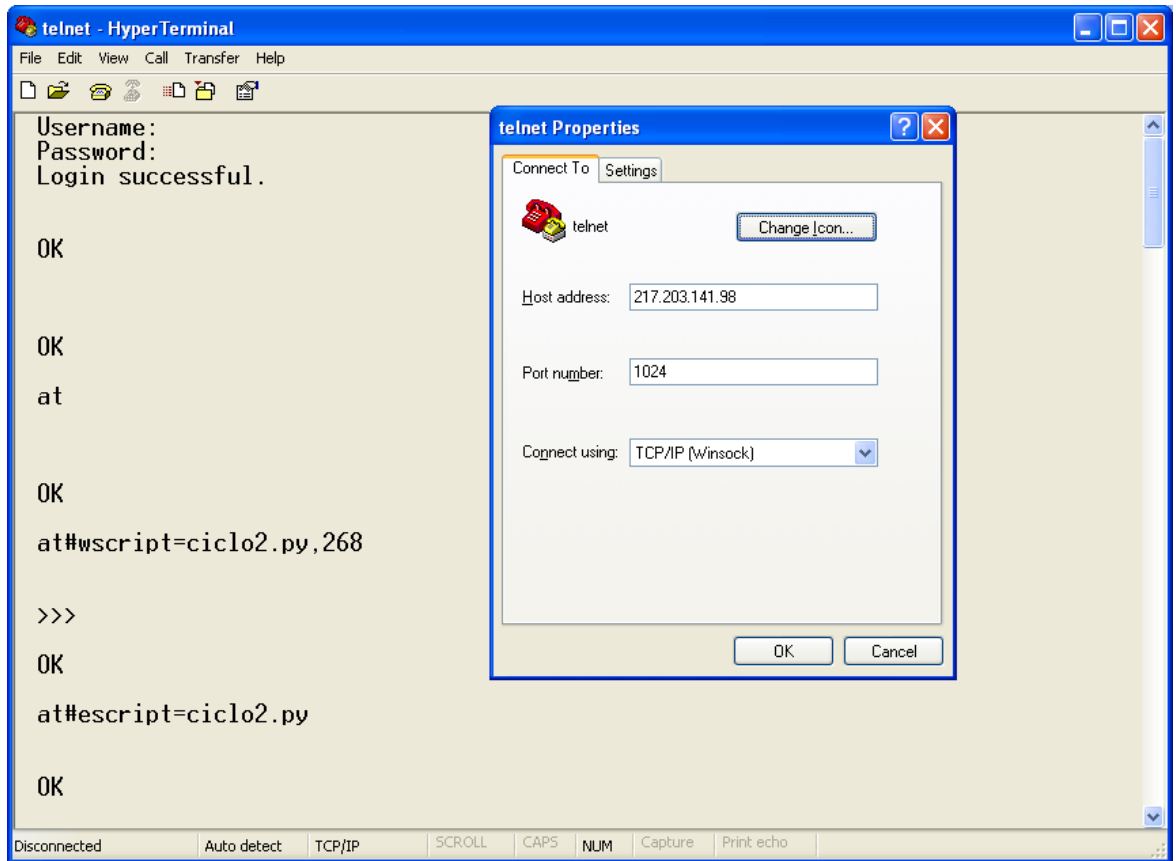






## Running AT commands remotely Application Note

80000NT10029a Rev.3 – 2010-10-04



To upgrade a Python script that is running using TCPATRUN service we recommend this procedure that can be easily execute using remote Telnet:

- 1) download a new PY script with the same procedure above
- 2) enable the new PY script with AT#ESCRIP=ciclo3.py”
- 3) reboot the module with AT#REBOOT to start the execution of the new python script (ciclo3.py)

Below is reported a series of screen-shots of the steps described above.





