Multisocket User Guide

80000ST10031a Rev. 1 - 07/09/07

Making machines talk.

This document is related to the following products:

APPLICABILITY TABLE

PRODUCT	PART NUMBER	APPLICABILITY
EZ10-QUAD-PY	3990150467	
GT863-PY	3990250466	
GT864-QUAD	4990150069	√
GT864-PY	4990150070	√
GM862-GPRS	3990250631	
GM862-QUAD	3990250655	
GM862-QUAD-PY	3990250656	
GM862-GPS	3990250657	√
GM862-GPS	3990250689	√
GM862-QUAD-PY	3990250658	√
GM862-QUAD	3990250659	√
GC864-QUAD	3990250675	√
GC864-PY	3990250676	√
GC864-QUAD-C2	3990250681	√
GC864-PY-C2	3990250686	√
GE863-QUAD	3990250653	
GE863-PY	3990250654	
GE863-GPS	3990250660	√
GE863-GPS	3990250690	√
GE863-PY	3990250661	√
GE863-QUAD	3990250662	√
GE864-PY	3990250650	√
GE864-QUAD	3990250648	√

Contents

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.

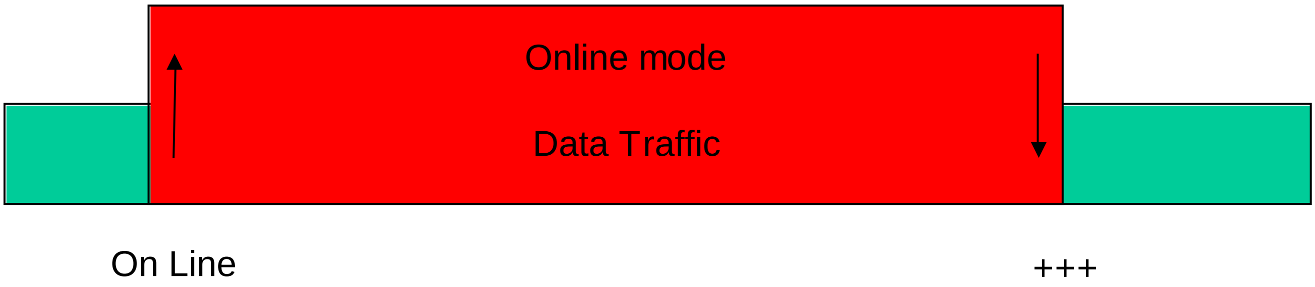
All rights reserved.

1 Multisocket Overview

New functionality of the Telit modules, multisocket is an extension of Telit Easy GPRS feature, which allows the user to have two contexts activated (that means two different IP address), more than one socket connection (with a maximum of 6) and simultaneous FTP client service.

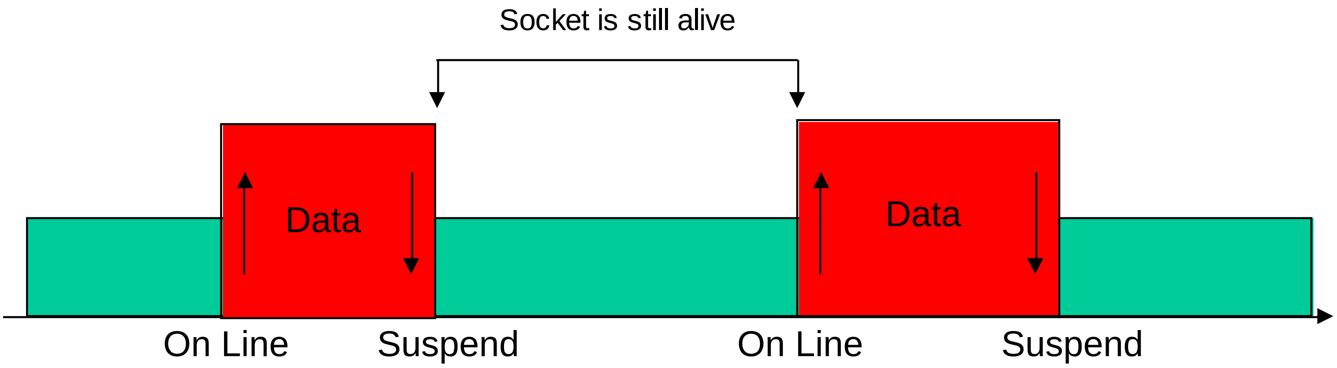
The basic idea of multisocket is the possibility of suspend a socket connection with the escape sequence +++.

With IP Easy we can use a SKTD to open a socket connection and go online. When the online activities are concluded we use +++ sequence to close the connection (see the figure below).



The green part represents the module command mode while the red part is the online mode.

Now, the online mode can be suspended with the escape sequence by using the multisocket feature. During suspend mode the data received by the socket will be buffered. These data will be displayed after socket resumption, as shown in the figure below:



This new feature allows users to switch between online mode and command mode without closing the connection and eventually opening another socket (or resuming the suspended one) or FTP connection.

Another new feature is the possibility to associate any socket connection to a specific context, this means that we can use different IP addresses for the connections (max 2). Socket identifier is called Connection Id (selects which socket we want to use from 1 up to 6) and every Connection Id is associated to a context.

1.1 Commands Overview

Let's take a look at the new AT commands sequence that activates GPRS context, sets and opens the socket connection. You can also find the explanation regarding new listen command and how to use FTP and IP Easy at the same time.

1.2 Socket configuration with #SCFG

Before opening a connection we have to set the socket parameters with the new #SCFG command. It is possible to set all the timeout values and packet size for each socket connection with a single AT command. The command syntax is:

AT#SCFG = <Conn Id>, <Cntx Id>, <Pkt sz>, <Global To>, <Conn To>, <Tx To>

Where:

- **Conn Id** -the connection identifier
- **Cntx Id** -the context identifier
- **Pkt sz** -the minimum data packet sent to the net (default 300 bytes)
- **Global To** -inactivity timeout (default 90 sec.)
- **Conn To** -connection timeout (default 60 sec, expressed in tenths of second)
- **Tx To** -data sending timeout (default 5 sec, expressed in tenths of second)

The first two parameters are new and they represent the association between the socket connection and the context set with +CGDCONT. It means that we can have socket connection working on different IP addresses.
The other parameters replace the old IP Easy commands #DSTO, #SKTTO, #SKTCT and #PKTSZ.

If we try to modify the socket configuration of an online connection an error will appear. So it's recommended to set the socket configuration at the beginning and to keep this configuration for the working session and also to dedicate the first Connection Id associated to context one for simultaneous FTP and IP Easy connection.

The values set with this command are saved in NVM.

Example:

We want to associate the Connection Id number 2 to the context number 3 with a minimum packet size of 512 bytes, global timeout of 30 sec, connection timeout of 30 sec and transmission timeout of 10 sec.

Command:

AT#SCFG = 2, 3, 512, 30, 300,100

Answer:

OK *if command execution is correct*
ERROR *if a parameter is wrong or the connection Id is working online*

1.3 Context activation with #SGACT

This command allows activation of one of the contexts defined with AT command +CGDCONT. With multisocket it is possible to activate simultaneously two context of the five that have been set. We can write username and password directly from command line (if required). At least one Connection Id must be associated to the context we want to activate otherwise an error will be appear.

The command syntax is:

#SGACT: <Cntx Id>,<Status>, [<Username>],[<Password>]

Where:

- **Cntx Id** is the context that we want to activate/deactivate.
- **Status** is the context status (0 means deactivation, 1 activation).

Example:

We want to activate context number two defined with +CGDCONT.

Command:

AT#SGACT = 2,1

Answer:
OK *if activation success.*
ERROR *if activation fails.*

1.4 Opening a connection with #SD

This is the new multisocket command that will be used to open a connection with a remote host, the command is similar to the old #SKTD command with the exception of the Connection Id parameter.

The command syntax is:

AT#SD = <Conn Id>,<Protocol>, <Remote Port>, <IP address> [, <Linger Time> [, <Local Port>]]

Where:

- **Conn Id** is the connection identifier.
- **Protocol is** 0 for TCP and 1 for UDP.
- **Remote Port** is the port of the remote machine.
- **IP address** is the remote address.

To open the remote connection the context to which the Connection Id is associated must be active, otherwise an error will appear.

For example if we want to connect to a web server with Connection Id number 3 the command is:

```
AT#SD = 3 , 0 , 80 , “www.telit.com”
```

If the command is successful we’ll have a CONNECT message, and the socket number 3 will be connected to the Telit webserver.
The main difference is that now the +++ sequence does not close the socket, but only suspends it. We can suspend the connection and open another one with a different Connection Id.
A typical command sequence is:

```
AT#SD = 3 , 0 , 80 , “www.telit.com”
CONNECT
(send, receive data….)

(+++)
OK
```

OK is returned after the escape sequence, it means that the socket has been suspended correctly.
Now the connection number 3 is suspended and the module is in command mode so we can give another #SD command.

```
AT#SD = 2 , 0 , 80 , “www.google.com”
CONNECT
(send, receive data….)

(+++)
OK
```

If we try to open a connection while the **ConnId** is in suspended state or online an error will be occur.

If a suspended connection receives some data the user will receive an unsolicited SRING indication from the module. In case we receive some data from the suspended connection with Telit server we'll receive this unsolicited message:

SRING: 3

where 3 is the number of the **ConnId** with data pending.

1.5 Resuming a suspended connection with #SO

This is the new command to resume a suspended connection, the command syntax is:

AT#SO = <Conn Id>

Example:

```
AT#SD = 2 , 0 , 80 , “www.google.com”
CONNECT
data sending

(+++)

OK

SRING: 2

AT#SO = 2
CONNECT
data sending

(+++)
```

If there is data pending on the socket (SRING displayed) after this command, pending data will be displayed.
It's not necessary to have a SRING message to resume a socket and is possible to resume it also if there aren't data pending on that connection.
Using AT#SO on a Connection Id in idle state we obtain a NO CARRIER message.

1.6 Closing a connection with #SH

With the new management of the escape sequence we need a command to close the socket connection. The AT command syntax to use is:

AT#SH = <conn Id>

Example:

```
AT#SD = 2 , 0 , 80 , “www.google.com”
CONNECT
data sending

(+++)

OK

AT#SH = 2
OK
```

Now the connection is closed. If we send this command with an idle Connection Id we obtain in any case an OK message.

1.7 Listen with #SL

The new listen command is now extended to 6 connections, it’s possible to set from 1 to 6 socket listening on a specific port for the incoming connections. Another difference with the old IP Easy is that now we receive an unsolicited indication when someone tries to connect, so we can decide to accept or refuse the incoming connection.
The command syntax is:

AT#SL = <Conn Id>, <Listen state>, <Listen port>[, <Linger Time>]

It’s not possible to have two **ConnId** listening on the same port.

Example:

Suppose that we want to listen on port 6543 Connection Id number 2

```
AT#SL = 2, 1, 6543
OK
```

Now the module is listening for incoming connection on port 6543 with Connection Id number 2, if a remote host is trying to connect we'll receive a SRING unsolicited indication with the listening Connection Id:

```
SRING: 2
```

1.8 Accept an incoming connection with #SA

After receiving the SRING indication for an incoming connection we can decide to refuse the remote host connection with #SH command or accept the connection with #SA command.
The command syntax is:

AT#SA = <conn Id>

Example:

We are listening on Connection Id 3 and port 6543

```
AT#SL = 3, 1, 6543
OK
```

A remote host is trying to connect, we receive the unsolicited indication.

```
SRING: 3
```

Now we accept the connection

```
AT#SA = 3
CONNECT
```

We pass in online mode and the connection is established. With the escape sequence we suspend the socket and the module is back to command mode. To resume the suspended connection we can use the #SO command described above.

1.9 Checking the socket status with #SS

With the old IP Easy socket connection the possible states were: online state or closed, while with multisocket suspension we have other socket states. With the new command AT#SS we can see the status of all the six sockets.

The command syntax is:

AT#SS

Suppose that we have suspended some sockets and we are in command mode, in order to verify which Connection Id has been opened, we can use AT#SS command to have a snapshot of sockets status.

The command result is:

#SS: <ConnId>,<Status>,<Local IP>,<Local Port>,<Remote IP>,<Remote Port>

For every Connection Id with have the information about our local IP address, local port, remote IP and port if we are connected.
The Status field represents the socket status:

0 – Socket Closed.
1 – Socket with an active data transfer connection.
2 – Socket suspended.
3 – Socket suspended with pending data.
4 – Socket listening.
5 – Socket with an incoming connection. Waiting for the user accept or shutdown command.

Example:

```
AT#SS
#SS: 1,4,217.201.131.110,21
#SS: 2,2,217.201.131.110,1033,194.185.15.73,10510
#SS: 3,3,217.201.131.110,1034,194.185.15.73,10510
#SS: 4,1,217.201.131.110,1035,194.185.15.73,10510
#SS: 5,0
#SS: 6,0

OK
```

In this case we can see Connection Id 1 in listen mode on port 21, number 2 suspended with no data pending, number 3 suspended with pending data and number 1 is online. The last two connections are closed.

2.1 Using FTP and IP Easy together

Another new functionality of multisocket is the simultaneous FTP client service with socket connections. We can use socket suspension mode to give FTP commands as in the old IP Easy, keeping socket alive and eventually resuming socket connections when we need to.

Note: It is suggested to leave Connection Id 1 associated to context 1 for using this functionality.

3.1 Using CMUX and Multisocket

Using CMUX we can have up to three virtual port to execute normal AT commands; if we join CMUX with multisocket we can share the six connections on the three ports (six is the total number in any case) and we can have up to three sockets active (online) at the same time.
FTP with CMUX is locked on the opening port. So if we open an FTP client connection on another virtual port the FTP commands will show an error message until the connection with FTP server is not closed. When the connection is closed we can open another FTP session on another virtual port. In any case we can always have only one FTP session opened.

4.1 Using old interface command on Multisocket

The old commands like #SKTD or #SKTL are available also on multisocket platform and they work like in the old IP Easy platform. If we open a connection with #SKTD we can't suspend the connection, and the +++ sequence will close definitively the connection.
In particular with #SKTD command we have the possibility to open three simultaneous connections using CMUX virtual ports. They are closed using the +++ sequence.

Note: #SKTOP has some limitations. It is available only on the first virtual port of CMUX and it is recommended not to use it with the new multisocket commands because #SKTOP deactivates the context when the connection is closed. This can generate the closure of suspended sockets. It's strongly recommended in any case to avoid using old IP Easy command with new multisocket commands.

5.1 Dial Up with Multisocket

With multisocket we recommend you to use the first context for a dialup connection and use the other available context for IP Easy socket connection.
The first context must be deactivated to make dialup connection work correctly, if we activate IP Easy and dialup at the same time the performance get worse. Anyway now is possible to make web browsing and IP Easy socket connection at the same time.

6.1 Examples

How to configure socket with SCFG command:

Associate the first Connection Id with context 1 and Packet size of 1000 byte:

```
AT#SCFG = 1,1,1000,90,600,50
OK
```

Associate the third Connection Id with context 3 and a global timeout of 60 seconds:

```
AT#SCFG = 3,2,300,60,600,50
OK
```

Let’s see an exhaustive example of how to open a socket connection, suspend it, opening a new socket, suspend it and resume the first connection.

Setting the contexts:

```
AT+CGDCONT = 1, “IP”, “ibox.tim.it”
OK

AT+CGDCONT = 2, “IP”, “ibox.tim.it”
OK
```

Activating context 1 and 2:

```
AT#SGACT = 1,1
#SGACT: 217.201.147.102
OK

AT#SGACT = 2,1
#SGACT: 217.201.152.103
OK
```

Open the first connection and suspend it:

```
AT#SD = 1,0,80,”www.telit.net”
CONNECT
(+++)
```

```
OK

Command mode, open another socket:
AT#SD = 2,0,80,”www.google.com”
CONNECT
(+++)
OK

Now we resume the first connection:
AT#SO = 1
CONNECT
(+++)
OK

To close the socket:
AT#SH = 1
OK

AT#SH = 2
OK

Now the sockets are closed. We can check the
status using AT#SS
```

Another example of listening socket with an incoming connection:

```
Setting the context:
AT+CGDCONT = 1, “IP”, “ibox.tim.it”
OK

Activating context 1 :
AT#SGACT = 1,1
#SGACT: 217.201.147.102
OK

Listening on Connection Id number 3 and port
6543:
AT#SL = 3,1,6543
OK

Now we are waiting for an incoming connection
on port 6543:

SRING: 3

When we receive the SRING on the listening
```

Conn Id we can decide to refuse the connection with AT#SH or accept it with:

```
AT#SA = 3
CONNECT
(online mode) → (+++)
OK
```

With +++ we are in suspended mode, now we can open another connection or listen on another port:

```
AT#SL = 1, 1, 6544
OK
```

Now we are listening also on connection Id number 1.

Now we want to check the sockets status:

```
AT#SS

#SS: 1,4,217.201.131.110,6544
#SS: 2,2,217.201.131.110,6543,194.185.15.73,10510
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
```

We have **Conn Id** 1 in listen on port 6544 and **Conn Id** 2 is suspended and is connected to host 194.185.15.73 on port 10510.

```
SRING: 1
```

Incoming connection, this time we want to refuse it and resume the suspended one.

```
AT#SH = 1
OK

AT#SO = 3
CONNECT
(online mode) → (+++)
OK
```

Close the **Conn Id** 3:

```
AT#SH = 3
OK
```

The third example is IP Easy and FTP at the same time:

Setting the contexts:

```
AT+CGDCONT = 1, “IP”, “ibox.tim.it”
OK
```

Activating context 1 :

```
AT#SGACT = 1,1
#SGACT: 217.201.147.102
OK
```

Open the first connection and suspend:

```
AT#SD = 1,0,80,”www.telit.net”
CONNECT
(+++)
OK
```

Command mode, open another socket:

```
AT#SD = 2,0,80,”www.google.com”
CONNECT
(+++)
OK
```

Now we open the FTP connection:

```
AT#FTPOPEN = “194.185.15.73:21","val","valfield"
OK
```

Check the sockets status:

```
AT#SS

#SS: 1,2,217.201.131.110,80,194.185.15.73,10510
#SS: 2,2,217.201.131.110,80,194.185.15.73,10510
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
```

We have the two IP Easy connection suspended and we can see that FTP connection doesn’t modify the multisocket status.

Now we can use all the FTP commands, for example:

```
AT#FTPLIST
```

To have a list of the FTP server files or :

```
AT#FTPGET = “filename.ext”
```

To get a file from the FTP server.

We resume one of the socket connection suspended with:

```
AT#SO = 1
CONNECT
(Online mode) → (+++)
OK
```

Now we have closed all the connections, to close the socket:

```
AT#SH = 1
OK

AT#SH = 2
OK
```

Now we close FTP connection:

```
AT#FTPCLOSE
OK
```

All the connections are closed.

Note: As we said before FTP is locked on the CMUX instance when opened, and FTP uses always context number one, so this must be active.

2 Document Change Log

Revision	Date	Changes
ISSUE#0	04/05/07	Initial Release
ISSUE#1	07/09/07	Added products into applicability table