



A76XX Series_TCPIP _Application Note

LTE Module

SIMCom Wireless Solutions Limited

SIMCom Headquarters Building, Building 3, No. 289 Linhong
Road, Changning District, Shanghai P.R. China

Tel: 86-21-31575100

support@simcom.com

www.simcom.com

Document Title:	A76XX Series_TCPIP_Application Note
Version:	1.01
Date:	2021.07.02
Status:	Released

GENERAL NOTES

SIMCOM OFFERS THIS INFORMATION AS A SERVICE TO ITS CUSTOMERS, TO SUPPORT APPLICATION AND ENGINEERING EFFORTS THAT USE THE PRODUCTS DESIGNED BY SIMCOM. THE INFORMATION PROVIDED IS BASED UPON REQUIREMENTS SPECIFICALLY PROVIDED TO SIMCOM BY THE CUSTOMERS. SIMCOM HAS NOT UNDERTAKEN ANY INDEPENDENT SEARCH FOR ADDITIONAL RELEVANT INFORMATION, INCLUDING ANY INFORMATION THAT MAY BE IN THE CUSTOMER'S POSSESSION. FURTHERMORE, SYSTEM VALIDATION OF THIS PRODUCT DESIGNED BY SIMCOM WITHIN A LARGER ELECTRONIC SYSTEM REMAINS THE RESPONSIBILITY OF THE CUSTOMER OR THE CUSTOMER'S SYSTEM INTEGRATOR. ALL SPECIFICATIONS SUPPLIED HEREIN ARE SUBJECT TO CHANGE.

COPYRIGHT

THIS DOCUMENT CONTAINS PROPRIETARY TECHNICAL INFORMATION WHICH IS THE PROPERTY OF SIMCOM WIRELESS SOLUTIONS LIMITED. COPYING, TO OTHERS AND USING THIS DOCUMENT, ARE FORBIDDEN WITHOUT EXPRESS AUTHORITY BY SIMCOM. OFFENDERS ARE LIABLE TO THE PAYMENT OF INDEMNIFICATIONS. ALL RIGHTS RESERVED BY SIMCOM IN THE PROPRIETARY TECHNICAL INFORMATION, INCLUDING BUT NOT LIMITED TO REGISTRATION GRANTING OF A PATENT, A UTILITY MODEL OR DESIGN. ALL SPECIFICATION SUPPLIED HEREIN ARE SUBJECT TO CHANGE WITHOUT NOTICE AT ANY TIME.

SIMCom Wireless Solutions Limited

SIMCom Headquarters Building, Building 3, No. 289 Linhong Road, Changning District, Shanghai P.R. China

Tel: +86 21 31575100

Email: simcom@simcom.com

For more information, please visit:

<https://www.simcom.com/download/list-863-en.html>

For technical support, or to report documentation errors, please visit:

<https://www.simcom.com/ask/> or email to: support@simcom.com

Copyright © 2021 SIMCom Wireless Solutions Limited All Rights Reserved.

About Document

Version History

Revision	Date	Chapter	Description
V1.00	2020.6.19		New version
V1.01	2020.8.25	3.2.3 TCP Client Works in Buffer Access Mode	Modify the tittle
	2021.02.03	All	Add support on A7678 Series

Scope

This document presents the AT Command Set for SIMCom A76XXand A7678 Series, including A76XX-XXXX, A5360E, A76X0X-XXXX and A7678C.

Contents

About Document.....	2
Version History.....	2
Scope.....	2
Contents.....	3
1 Introduction.....	5
1.1 Purpose of the document.....	5
1.2 Related documents.....	5
1.3 Conventions and abbreviations.....	5
1.4 The process of Using TCPIP AT Commands.....	6
1.5 Error Handling.....	9
1.5.1 Executing FTP(S) AT Commands Fails.....	9
1.5.2 PDP Activation Fails.....	9
1.5.3 Error Response of TCPIP Server.....	9
1.5.4 Description of Data Access Mode.....	9
2 AT Commands for TCPIP.....	11
2.1 TCPIP Services AT.....	11
3 TCPIP Examples.....	12
3.1 Configure and Activate context.....	12
3.1.1 Network Environment.....	12
3.1.2 Configure Context.....	12
3.1.3 Activate context.....	12
3.1.4 Deactivate Context.....	13
3.2 TCP Client.....	13
3.2.1 TCP Client Works in Direct Push Mode.....	13
3.2.2 TCP Client Works in Buffer Access Mode.....	14
3.2.3 TCP Client Works in Transparent Access Mode.....	15
3.3 UDP Client.....	17
3.3.1 UDP Client Works in Direct Push Mode.....	17
3.3.2 UDP Client Works in Buffer Access Mode.....	18
3.3.3 UDP Client Works in Transparent Access Mode.....	19
3.4 TCP Server.....	20
3.4.1 Transparent Mode.....	20
3.4.2 Non-Transparent Mode.....	21
3.4.3 Query Connection Status.....	22
4 Appendix.....	23
4.1 Summary of Error Codes.....	23

4.2 Unsolicited Result Codes.....	24
-----------------------------------	----

SIMCom
Confidential

1 Introduction

1.1 Purpose of the document

Based on module AT command manual, this document will introduce TCPIP application process. Developers could understand and develop application quickly and efficiently based on this document.

1.2 Related documents

[1] A76XXSeries_AT Command Manual

1.3 Conventions and abbreviations

In this document, the GSM engines are referred to as following term:

ME (Mobile Equipment);

MS (Mobile Station);

TA (Terminal Adapter);

DCE (Data Communication Equipment) or facsimile DCE (FAX modem, FAX board);

In application, controlling device controls the GSM engine by sending AT Command via its serial interface.

The controlling device at the other end of the serial line is referred to as following term:

TE (Terminal Equipment);

DTE (Data Terminal Equipment) or plainly "the application" which is running on an embedded system;

Other Conventions:

PDP(Packet Data Protocol);

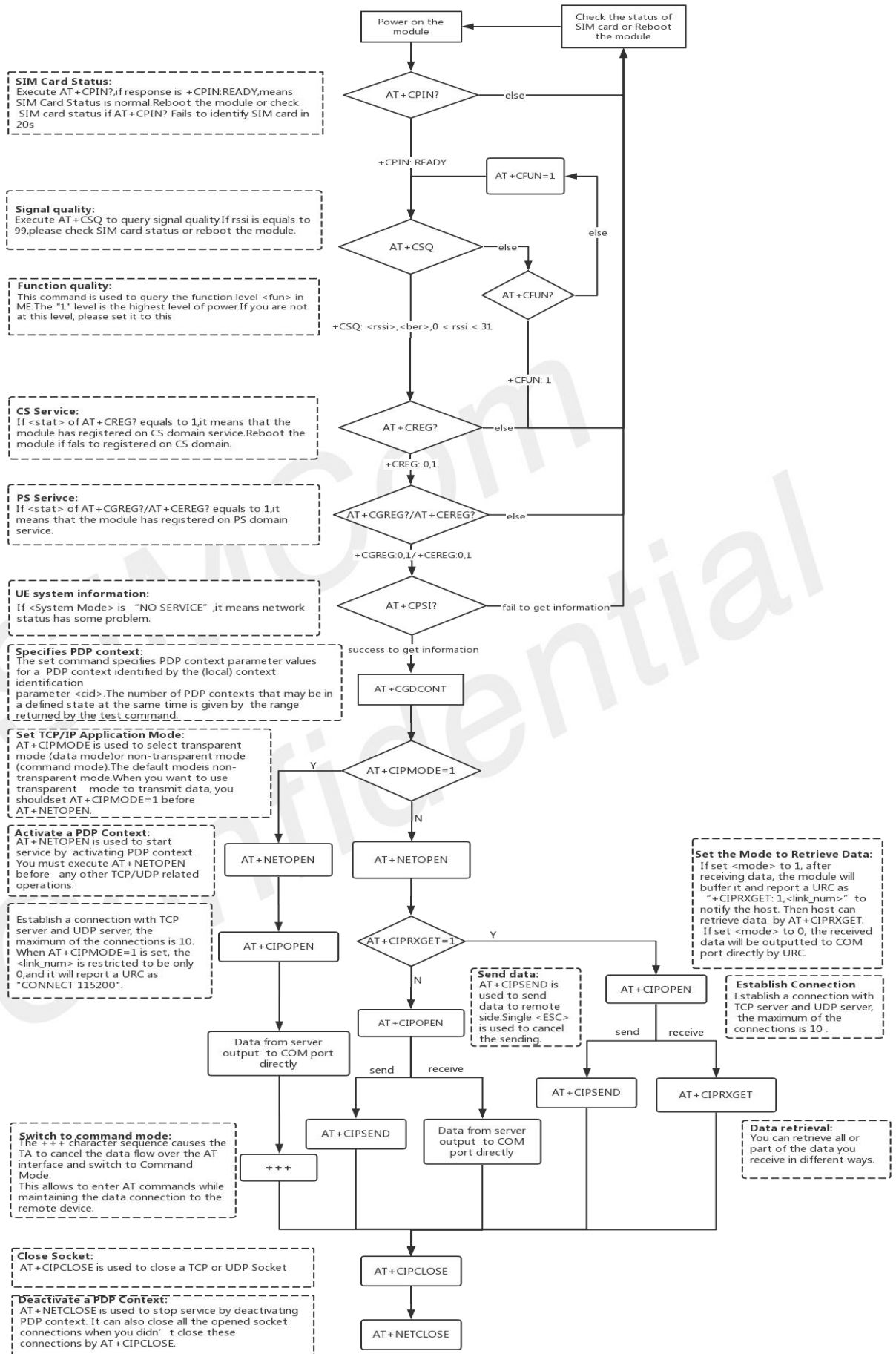
TCP(Terminal Control Protocol);

UDP(User Datagram Protocol);

1.4 The process of Using TCPIP AT Commands

Figure illustrates how to use TCP/IP AT commands:

SIMCom
Confidential



NOTE: If you need to use the TCP server, you'll need special SIM cards.

SIM Card Status:
Execute AT+CPIN?, if response is +CPIN:READY, means SIM Card Status is normal. Reboot the module or check SIM card status if AT+CPIN? fails to identify SIM card in 20s

Signal quality:
Execute AT+CSQ to query signal quality. If rssi is equals to 99, please check SIM card status or reboot the module.

Function quality:
This command is used to query the function level <fun> in ME. The "1" level is the highest level of power. If you are not at this level, please set it to this

CS Service:
If <stat> of AT+CREG? equals to 1, it means that the module has registered on CS domain service. Reboot the module if fails to registered on CS domain.

PS Service:
If <stat> of AT+CGREG?/AT+CREG? equals to 1, it means that the module has registered on PS domain service.

UE system information:
If <System Mode> is "NO SERVICE", it means network status has some problem.

Specifies PDP context:
The set command specifies PDP context parameter values for a PDP context identified by the (local) context identification parameter <cid>. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command.

Set TCP/IP Application Mode:
AT+CIPMODE is used to select transparent mode (data mode) or non-transparent mode (command mode). The default mode is non-transparent mode. When you want to use transparent mode to transmit data, you should set AT+CIPMODE=1 before AT+NETOPEN.

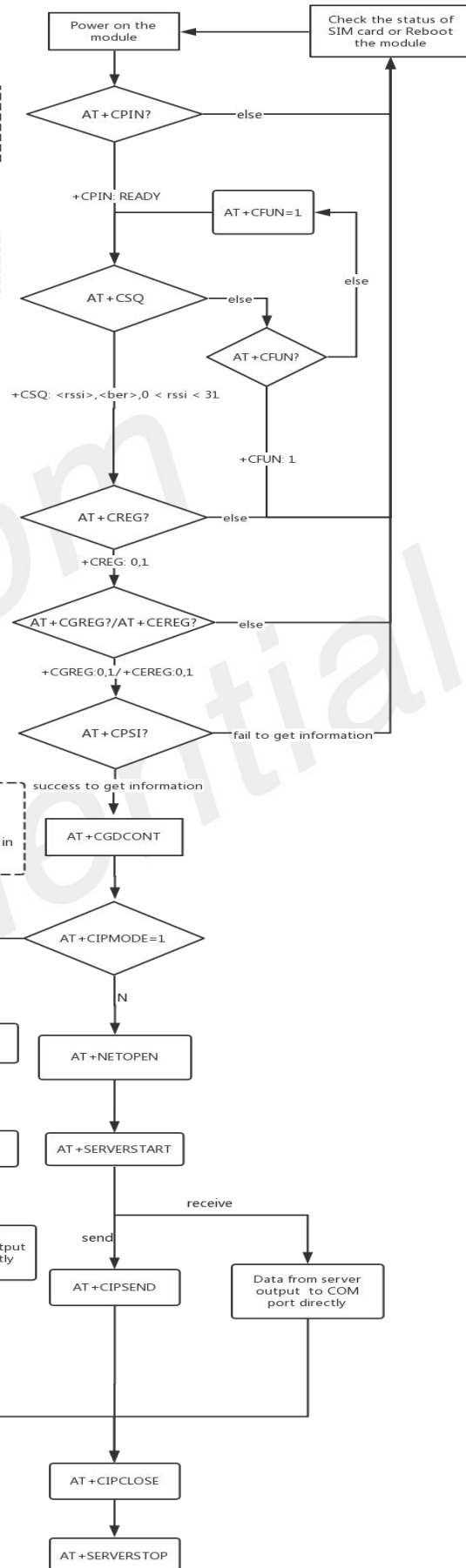
Activate a PDP Context:
AT+NETOPEN is used to start service by activating PDP context. You must execute AT+NETOPEN before any other TCP/UDP related operations.

Open Connection:
Establish a connection with TCP server and UDP server, the maximum of the connections is 10. When AT+CIPMODE=1 is set, the <link_num> is restricted to be only 0

Switch to command mode:
The +++ character sequence causes the TA to cancel the data flow over the AT interface and switch to Command Mode. This allows to enter AT commands while maintaining the data connection to the remote device.

Close Socket:
AT+CIPCLOSE is used to close a TCP or UDP Socket

Deactivate a PDP Context:
AT+NETCLOSE is used to stop service by deactivating PDP context. It can also close all the opened socket connections when you didn't close these connections by AT+CIPCLOSE.



1.5 Error Handling

1.5.1 Executing FTP(S) AT Commands Fails

When executing TCPIP AT commands, if ERROR response is received from the module, please check whether the U(SIM) card is inserted and whether it is +CPIN: READY returned when executing AT+CPIN?.

1.5.2 PDP Activation Fails

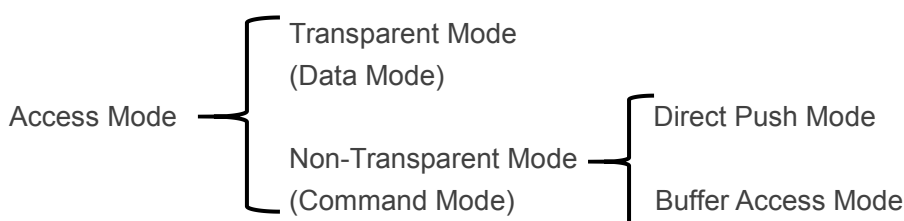
If it is failed to activate a PDP context with AT+NETOPEN command, please make sure the PDP is not activated. You can use AT+NETOPEN? to query it.

If all above configurations are correct, but activating the PDP context by AT+NETOPEN command still fails, please reboot the module to resolve this issue. After rebooting the module, please check the configurations mentioned above for at least.

1.5.3 Error Response of TCPIP Server

If you encounter other errors, please refer to chapter 4 to correct them.

1.5.4 Description of Data Access Mode



The default mode is direct push mode.

1. Direct Push Mode

In direct push mode, user can send data by AT+CIPSEND. The received data will be outputted to COM port directly by URC as "+RECV FROM:<IP ADDRESS>:<PORT><CR><LF>+IPD(data length)<CR><LF><data>".

2. Buffer Access Mode

AT+CIPRXGET=1 is used to enter into buffer access mode. In buffer access mode, user sends data by AT+CIPSEND. After receiving data, the module will buffer it and report a URC as "+CIPRXGET: 1,<link_num>" to notify the host. Then host can retrieve data by AT+CIPRXGET.

3. Transparent Access Mode

AT+CIPMODE=1 is used to enter into transparent access mode. In transparent mode, the data received from COM port will be sent to internet directly, and the received data from Internet will be output to COM port directly as well. "+++" is used to exit from transparent access mode. When "+++" returns OK, the module will be switched to command mode. In transparent access mode, host cannot execute any AT command. Note: Currently, only one socket is available under transparent mode, either TCP client or TCP server. In transparent mode, the first server (<server_index> = 0) and the first client socket (<link_num> = 0) are used for transparent mode operation. Other servers (<server_index> = 1-3) and other client sockets (<link_num> = 1-9) are still used in command mode.

4. Switch Between Data Mode and Command Mode

(1) Data mode -> Command mode

Software switching: escape sequence +++. Please take care, this is a complete command, do not separate each character. And the time delay before and after this sequence should be more than 1000 milliseconds, the interval of each character should not be more than 900 milliseconds.

Hardware switching: DTR pin could be used to trigger data mode and command mode. Command AT&D1 should be configured before application.

(2) Command Mode -> Data Mode

ATO is used to enter into transparent access mode from command mode. If it enters into transparent access mode successfully, CONNECT<text> will be returned.

2 AT Commands for TCPIP

2.1 TCPIP Services AT

Command	Description
AT+NETOPEN	Start Socket Service
AT+NETCLOSE	Stop Socket Service
AT+CIOPEN	Establish Connection in Multi-Socket Mode
AT+CIPSEND	Send data through TCP or UDP Connection
AT+CIPRXGET	Set the Mode to Retrieve Data
AT+CIPCLOSE	Close TCP or UDP Socket
AT+IPADDR	Inquire Socket PDP address
AT+CIPHEAD	Add an IP Header When Receiving Data
AT+CIPSRIP	Show Remote IP Address and Port
AT+CIPMODE	Set TCP/IP Application Mode
AT+CIPSENDMODE	Set Sending Mode
AT+CIPTIMEOUT	Set TCP/IP Timeout Value
AT+CIPCCFG	Configure Parameters of Socket
AT+SERVERSTART	Startup TCP Server
AT+SERVERSTOP	Stop TCP Server
AT+CIPACK	Query TCP Connection Data Transmitting Status
AT+CDNSGIP	Query the IP Address of Given Domain Name

3 TCPIP Examples

3.1 Configure and Activate context

3.1.1 Network Environment

TCP/IP application is based on GPRS network. Please make sure that GPRS network is available before TCP/IP setup.

AT+CSQ

+CSQ: 23,0

OK

AT+CREG?

+CREG: 0,1

OK

AT+CGREG?

+CGREG: 0,1

OK

3.1.2 Configure Context

AT+CGDCONT=1,"IP","CMNET"

OK

3.1.3 Activate context

AT+NETOPEN

OK

+NETOPEN: 0

AT+IPADDR

+IPADDR: 10.148.0.17

OK

3.1.4 Deactivate Context

AT+NETCLOSE

OK

+NETCLOSE: 0

AT+IPADDR

+IPADDR: Network not opened

ERROR

3.2 TCP Client

3.2.1 TCP Client Works in Direct Push Mode

//Set up TCP Client Connection

AT+NETOPEN

OK

+NETOPEN: 0

AT+CIOPEN=1,"TCP","117.131.85.139",5253

OK

+CIOPEN: 1,0

// set up a TCP connection, <link_num> is 1.
Before using AT+CIOPEN, host should activate
PDP Context with AT+NETOPEN first.

//Send Data To Server

AT+CIPSEND=1,5

// send data with fixed length

```
>HELLO  
OK
```

```
+CIPSEND: 1,5,5
```

```
//Receive Data From Server
```

```
RECV FROM:117.131.85.139:5253
```

```
// data from server directly output to COM
```

```
+IPD16
```

```
data from server
```

```
//Close TCP Connection
```

```
AT+CIPCLOSE=1
```

```
OK
```

```
+CIPCLOSE: 1,0
```

3.2.2 TCP Client Works in Buffer Access Mode

```
//Set up TCP Client Connection
```

```
AT+NETOPEN
```

```
OK
```

```
+NETOPEN: 0
```

```
AT+CIPRXGET=1
```

```
// buffer access mode, get data by AT+CIPRXGET
```

```
OK
```

```
AT+CIOPEN=1,"TCP","117.131.85.139",5253
```

```
OK
```

```
+CIOPEN: 1,0
```

```
//Send Data to Server
```

```
AT+CIPSEND=1,5
```

```
// send data with fixed length
```

```
>hello
```

```
OK
```

```
+CIPSEND: 1,5,5
```

```
//Receive Data from Server
```

```
+CIPRXGET: 1,1           // URC to notify host of data from server
AT+CIPRXGET=4,1          // query the length of data in the buffer of socket
+CIPRXGET: 4,1,16        with
                           // <link_num>=1

OK
AT+CIPRXGET=2,1,5        // get data in ASCII form
+CIPRXGET: 2,1,5,11data  // read 5 bytes data and left 11 bytes

OK
AT+CIPRXGET=3,1,5        // get data in hex form
+CIPRXGET: 3,1,5,6
66726F6D20

OK
AT+CIPRXGET=4,1          // read the length of unread data in buffer
+CIPRXGET: 4,1,6

OK
AT+CIPRXGET=2,2          // the connection identified by link_num=2 has not
+IP ERROR: No data       been established

ERROR
AT+CIPRXGET=2,1
+CIPRXGET: 2,1,6,0
server

OK
AT+CIPRXGET=4,1          // all the data in buffer has been read, the rest_len
+CIPRXGET: 4,1,0         is 0.

OK
```

//Close TCP Connection

AT+CIPCLOSE=1

OK

+CIPCLOSE: 1,0

3.2.3 TCP Client Works in Transparent Access Mode

//Set up TCP Client Connection

AT+CIPMODE=1

// Enter into transparent mode by at+cipmode=1

OK

AT+NETOPEN

OK

+NETOPEN: 0

AT+CIPOEPN=0,"TCP","117.131.85.139",5253

CONNECT 115200

// only <link_num>=0 is allowed to operate with transparent mode.

//Send Data to Server

All data got from com port will be sent to internet directly

//Receive Data From Server

DATA FROM SERVERDATA FROM SERVER

OK

//all the received data from server will be output to com port directly

//sequence of +++ to quit transparent mode

AT+CIOPEN?

+CIOPEN: 0,"TCP","117.131.85.139",5253,-1

+CIOPEN: 1

+CIOPEN: 2

+CIOPEN: 3

+CIOPEN: 4

+CIOPEN: 5

+CIOPEN: 6

+CIOPEN: 7

+CIOPEN: 8

+CIOPEN: 9

OK

ATO

//ATO to enter transparent mode again

CONNECT 115200

HELLO CLIENT

OK

//Close TCP Connection

AT+CIPCLOSE=0

OK

CLOSED

+CIPCLOSE: 0,0

3.3 UDP Client

3.3.1 UDP Client Works in Direct Push Mode

//Set up UDP Client Connection

AT+NETOPEN

OK

+NETOPEN: 0

AT+CIPOPEN=1,"UDP",,,5000

+CIPOPEN: 1,0

OK

// when set a UDP connection, the remote IP address and port is not necessary, but the local port must be specified.

//Send data to Server

AT+CIPSEND=1,"117.131.85.139",5254

>HELLOSERVER

OK <CTRL+Z>

+CIPSEND: 1,11,11

AT+CIPSEND=1,5,"117.131.85.139",5254

>HELLO

OK

// for UDP connection, when sending data, user must specify the remote IP address and port
//send data with changeable length, <CTRL+Z> to end

//send data with fixed length

+CIPSEND: 1,5,5

//Receive Data From Server

RCV FROM:117.131.85.139:5254

+IPD14

HELLO CLIENT

//data from server output to COM port directly

//Close UDP Connection

AT+CIPCLOSE=1

+CIPCLOSE: 1,0

OK

3.3.2 UDP Client Works in Buffer Access Mode

//Set up UDP Client Connection

AT+NETOPEN

OK

+NETOPEN: 0

AT+CIPRXGET=1

// buffer access mode, get data by AT+CIPRXGET

OK

AT+CIOPEN=1,"UDP",,,5000

// when set a UDP connection, the remote IP address and port is not necessary, but the local port

+CIOPEN: 1,0

must be specified.

OK

//Send Data to Server

AT+CIPSEND=1,"117.131.85.139",5254

// for UDP connection, when sending data, user must specify the remote IP address and port

>HELLOSERVER

//send data with changeable length, <CTRL+Z> to end

OK <CTRL+Z>

+CIPSEND: 1,11,11

AT+CIPSEND=1,5,"117.131.85.139",5254

//send data with fixed length

>HELLO

OK

+CIPSEND: 1,5,5

//Receive Data From Server

+CIPRXGET: 1,1

// URC to notify host of data from server

AT+CIPRXGET=4,1

// query the length of data in the buffer of socket with <link_num>=1

+CIPRXGET: 4,1,16

OK

AT+CIPRXGET=2,1,5

// get data in ASCII form

+CIPRXGET: 2,1,5,11

data

OK

AT+CIPRXGET=3,1,5

// get data in hex form

+CIPRXGET: 3,1,5,6

66726F6D20

OK

AT+CIPRXGET=4,1

// read the length of unread data in buffer

+CIPRXGET: 4,1,6

OK

AT+CIPRXGET=2,2

// the connection identified by link_num=2 has not been established

+IP ERROR: No data

ERROR

AT+CIPRXGET=2,1

+CIPRXGET: 2,1,6,0

server

OK

AT+CIPRXGET=4,1

// all the data in buffer has been read, the rest_len is 0.

+CIPRXGET: 4,1,0

OK

//Close UDP Connection

AT+CIPCLOSE=1

OK

+CIPCLOSE: 1,0

3.3.3 UDP Client Works in Transparent Access Mode

//Set up UDP Client Connection

AT+CIPMODE=1

OK

AT+NETOPEN

OK

+NETOPEN: 0

**AT+CIOPEN=0,"UDP","117.131.85.139",5254,
5000**

//only <link_num>=0 is allowed to operate with transparent mode.

CONNECT 115200

//Send Data to Server

**All data got from com port will be sent to
internet directly**

//Receive Data From Server

HELLO CLIENT

///data

HELLO CLIENT

from server output to COM port directly

OK

// sequence of +++ to quit transparent mode

AT+CIOPEN?

+CIOPEN: 0,"UDP","117.131.85.139",5254,-1

+CIOPEN: 1

+CIOPEN: 2

+CIOPEN: 3

+CIOPEN: 4

+CIOPEN: 5

+CIOPEN: 6

+CIOPEN: 7

+CIOPEN: 8

+CIOPEN: 9

OK

AT+CIOPEN=0,"UDP","117.131.85.139",5254,
5000

//only <link_num>=0 is allowed to operate with
transparent mode.

CONNECT 115200

3.4 TCP Server

3.4.1 Transparent Mode

AT+CIPMODE=1

OK

AT+NETOPEN

OK

+NETOPEN: 0

AT+SERVERSTART=8080, 0

//only <server_index>=0 is allowed to operate with
transparent mode.

OK

+CLIENT: 0,0,192.168.108.5:57202

//only <link_num> 0 can be used for transparent
mode operation.

CONNECT 115200

OK

// sequence of +++ to quit data mode

AT+CIPCLOSE=0

// close client connection

OK

CLOSED

```
+CIPCLOSE: 0,0
AT+SERVERSTOP=0           // close server socket
+SERVERSTOP: 0,0

OK
```

3.4.2 Non-Transparent Mode

```
AT+NETOPEN
OK

+NETOPEN: 0
AT+SERVERSTART=8080, 0     //only <server_index>=0 is allowed to operate with
OK                          transparent mode.
AT+SERVERSTART=9090, 1
OK
AT+SERVERSTART=7070, 2
OK
AT+SERVERSTART=6060, 3
OK
+CLIENT: 0,0,192.168.108.5:57202 //If a socket is accepted, the following URC will be
                                reported:
AT+CIPOPEN?                //User can use AT+CIPOPEN? to check the
+CIPOPEN: 0,"TCP","192.168.108.5",57202,1 //last parameter of 1 indicates this is an accepted
+CIPOPEN: 1                socket, this server index is 1
+CIPOPEN: 2
+CIPOPEN: 3
+CIPOPEN: 4
+CIPOPEN: 5
+CIPOPEN: 6
+CIPOPEN: 7
+CIPOPEN: 8
+CIPOPEN: 9

OK
AT+CIPSEND=0,5             // only supports fixed-length to send
>HELLO
OK

+CIPSEND: 0,5,5
AT+SERVERSTOP=0           // if unspecified, it will close 0 channel
+SERVERSTOP: 0,0
OK
AT+SERVERSTOP=1
+SERVERSTOP: 1,0
```

OK
AT+SERVERSTOP=2
+SERVERSTOP: 2,0

OK
AT+SERVERSTOP=3
+SERVERSTOP: 3,0

OK
AT+NETCLOS
OK
+NETCLOSE: 0

3.4.3 Query Connection Status

AT+CIOPEN=1,"TCP","117.131.85.139",5253
OK

+CIOPEN: 1,0

AT+CIOPEN?

// query the current state of all sockets

+CIOPEN: 0

+CIOPEN: 1,"TCP","117.131.85.139",5253,-1

+CIOPEN: 2

+CIOPEN: 3

+CIOPEN: 4

+CIOPEN: 5

+CIOPEN: 6

+CIOPEN: 7

+CIOPEN: 8

+CIOPEN: 9

OK
AT+CIPCLOSE?

+CIPCLOSE: 0,1,0,0,0,0,0,0,0,0

OK
AT+CIPCLOSE=1
OK

+CIPCLOSE: 1,0

AT+CIPCLOSE?

+CIPCLOSE: 0,0,0,0,0,0,0,0,0,0

OK

4 Appendix

4.1 Summary of Error Codes

When you use these commands : AT+CIPACK AT+CIPRXGET, If something goes wrong, they maybe reported as +IP ERROR: <err_info>.

The fourth parameter <errMode> of AT+CIPCCFG (TODO) is used to determine how <err_info> is displayed.

If <errMode> is set to 0, the <err_info> is displayed with numeric value.

If <errMode> is set to 1, the <err_info> is displayed with string value.

The default is displayed with string value.

The following list is the description of the <err_info>.

Numeric Value	String Value
0	Connection time out
1	Bind port failed
2	Port overflow
3	Create socket failed
4	Network is already opened
5	Network is already closed
6	No clients connected
7	No active client
8	Network not opened
9	Client index overflow
10	Connection is already created
11	Connection is not created
12	Invalid parameter
13	Operation not supported
14	DNS query failed
15	TCP busy
16	Net close failed for socket opened
17	Sending time out
18	Sending failure for network error
19	Open failure for network error

20	Server is already listening
21	Operation failed
22	No data

When you use these commands : AT+NETOPEN, AT+NETCLOSE, AT+CIOPEN, AT+CIPSEND, AT+CIPCLOSE, AT+SERVERSTART, AT+SERVERSTOP ,If something goes wrong, they will report the wrong number

The following list is the description of the <err>.

<err>	Description of <err>
0	operation succeeded
1	Network failure
2	Network not opened
3	Wrong parameter
4	Operation not supported
5	Failed to create socket
6	Failed to bind socket
7	TCP server is already listening
8	Busy
9	Sockets opened
10	Timeout
11	DNS parse failed for AT+CIOPEN
12	Unknown error

4.2 Unsolicited Result Codes

Information	Description
+CIP EVENT: NETWORK CLOSED UNEXPECTEDLY	Network is closed for network error(Out of service, etc). When this event happens, user's application needs to check and close all opened sockets, and then uses AT+NETCLOSE to release the network library if AT+NETOPEN? shows the network library is still opened.

+IPCLOSE: <client_index>,<close_reason>	Socket is closed passively. <client_index> is the link number. <close_reason>: 0 - Closed by local, active 1 - Closed by remote, passive 2 - Closed for sending timeout or DTR off
+CLIENT: < link_num>,<server_index>,<client_IP>:<port>	TCP server accepted a new socket client, the index is<link_num>, the TCP server index is <server_index>. The peer IP address is <client_IP>, the peer port is <port>.

SIMCom
Confidential