



# SIM7020 Series\_ TCPIP\_Application Note

LPWA Module

## **SIMCom Wireless Solutions Limited**

Building B, SIM Technology Building, No.633, Jinzhong Road

Changning District, Shanghai P.R. China

Tel: 86-21-31575100

[support@simcom.com](mailto:support@simcom.com)

[www.simcom.com](http://www.simcom.com)

<b>Document Title:</b>	SIM7020 Series_TCPIP_Application Note
<b>Version:</b>	1.04
<b>Date:</b>	2020.6.10
<b>Status:</b>	Release

## 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**

Building B, SIM Technology Building, No.633 Jinzhong Road, Changning District, Shanghai P.R. China

Tel: +86 21 31575100

Email: [simcom@simcom.com](mailto: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](mailto:support@simcom.com)

**Copyright © 2020 SIMCom Wireless Solutions Limited All Rights Reserved.**

# About Document

## Version History

Version	Date	Owner	What is new
V1.00	2018.4.10	Xiaolun.Wang	First Release
V1.01	2018.4.18	Albert.Meng	Change AT+SHBOD
V1.02	2018.12.27	Jiaxiang.Wang	Modify and add chapter 4 about TCPIP Application which compatible with SIM800 serial modules. Add transparent mode
V1.03	2019.12.20	Wenjie.Lai	Revised
V1.04	2020.6.10	Lei.Wang	All

## Scope

This document applies to the following products

Name	Type	Size(mm)	Comments
SIM7020C	NB1	17.6*15.7	Band 1/3/5/8
SIM7020E	NB1	17.6*15.7	Band 1/3/5/8/20/28
SIM7020G	NB2	17.6*15.7	Band 1/2/3/4/5/8/12/13/17/18/19/20/25/26/28/66/70/71/85
SIM7060G	NB2+GNSS	24*24	Band 1/2/3/4/5/8/12/13/17/18/19/20/25/26/28/66/70/71/85

# Contents

<b>About Document</b> .....	<b>3</b>
Version History .....	3
Scope .....	3
<b>Contents</b> .....	<b>4</b>
<b>1 Introduction</b> .....	<b>6</b>
1.1 Purpose of the document .....	6
1.2 Related documents .....	6
1.3 Conventions and abbreviations .....	6
<b>2 AT Commands for TCPIP</b> .....	<b>7</b>
2.1 AT Commands for TCPIP Application. ....	7
2.2 AT Commands for TCPIP Application to Compatible with SIM800 Serials .....	7
<b>3 Bearer Configuration</b> .....	<b>9</b>
3.1 PDN Auto-activation .....	9
3.2 APN Manual Configuration .....	9
<b>4 DNS parser and PING</b> .....	<b>11</b>
4.1 DNS Parser .....	11
4.2 PING .....	11
<b>5 TCPIP Connection</b> .....	<b>12</b>
5.1 TCPIP Architecture .....	12
5.2 TCP client .....	12
5.3 UDP Connection .....	13
5.4 TCP Server .....	13
5.5 Manually get data .....	14
5.6 Multiple Sockets .....	16
5.7 Hex and ASCII Message .....	16
5.8 TCP ACK .....	16
5.9 Incoming Message Indication .....	17
5.10 Retention Scene when Module is Waked from PSM Mode .....	17
5.11 Summary of Socket Error codes .....	18
5.12 TCPIP Connection flow chart .....	18
<b>6 TCPIP Application compatible with SIM800 serial module</b> .....	<b>20</b>
6.1 TCPIP Architecture .....	20
6.2 TCP Client Connection .....	20
6.3 UDP Client Connection .....	21
6.4 UDP Extended Mode .....	21
6.5 TCP Server .....	23

6.6	Multi Clients Connection .....	24
6.7	Multi Mixing Connection .....	25
6.8	DNS Parser and Connection.....	27
6.9	Data Sending Related.....	28
6.9.1	Data Sending Related .....	28
6.9.2	Fixed Length Sending .....	28
6.9.3	Select Data Transmitting Mode .....	28
6.9.4	Query Data Transmitting Amount .....	29
6.10	Data Receiving Related .....	31
6.10.1	Receive Data Automatically.....	31
6.10.2	Receive Data Manually .....	31
6.11	GPRS States Exchange Related .....	32
6.12	Connection Closing Related.....	33
6.13	Error Handling.....	34
6.14	Transparent Mode.....	34
6.14.1	What is Transparent Mode .....	34
6.14.2	How to enable Transparent Mode .....	34
6.14.3	How to Establish Connection and Enter into Transparent Mode after Enable Transparent Mode	35
6.14.4	How to Switch Between Data Mode and Command Mode.....	35

SIMCom  
Confidential

# 1 Introduction

## 1.1 Purpose of the document

Based on module AT command manual, this document will give an entire and complete concept and TCPIP architecture introduction.

Developers could understand and develop application quickly and efficiently based on this Document.

## 1.2 Related documents

[1] SIM7020 Series\_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;

## 2 AT Commands for TCPIP

### 2.1 AT Commands for TCPIP Application.

Command	Description
AT+CSOC	Create a TCP/UDP socket
AT+CSOCON	Connect socket to remote address and port
AT+CSOB	Bind local address and local port
AT+CSOLIS	Open Listening Function of the Server
AT+RETENTION	Retention of socket scene
AT+CSOSEND	Send data to remote via socket
AT+CSODSEND	Send data to remote via socket with data mode
AT+CSOCL	Close socket
AT+CSOSENDFLAG	Set TCP send flag
AT+CSORCVFLAG	Set receive flag
AT+CSOSTATUS	Get socket status
AT+CSOACK	Query previous connection data transmitting state
AT+CSOALIVE	Set TCP keepalive parameters
AT+CSORXGET	Get Data from Network Manually
+CSONMI	Socket message arrived indicator
+CSOERR	Socket error indicator

### 2.2 AT Commands for TCPIP Application to Compatible with SIM800 Serials

Command	Description
AT+CIPMUX	Start up multi-IP connection
AT+CIPSTART	Start up TCP or UDP connection
AT+CIPSEND	Send data through TCP or UDP connection
AT+CIPQSEND	Select data transmitting mode
AT+CIPACK	Query previous connection data transmitting state
AT+CIPCLOSE	Close TCP or UDP connection
AT+CIPSHUT	Deactivate GPRS PDP context

<b>AT+CLPORT</b>	Set local port
<b>AT+CSTT</b>	Start task and set APN, user name, password
<b>AT+CIICR</b>	Bring up wireless connection
<b>AT+CIFSR</b>	Get local IP address
<b>AT+CIPSTATUS</b>	Query current connection status
<b>AT+CDNSCFG</b>	Configure domain name server
<b>AT+CDNSGIP</b>	Query the IP address of given domain name
<b>AT+CIPHEAD</b>	Add an IP head at the beginning of a package received
<b>AT+CIPHEXS</b>	Show data in hex mode of a package received
<b>AT+CIFSREX</b>	Get local IP address
<b>AT+CIPATS</b>	Set auto sending timer
<b>AT+CIPSPRT</b>	Set prompt of '>' when module sends data
<b>AT+CIPSERVER</b>	Configure module as server
<b>AT+CIPCSGP</b>	Set CSD or GPRS for connection mode
<b>AT+CIPSRIP</b>	Show remote IP address and port when received data
<b>AT+CIPSHOWTP</b>	Display transfer protocol in IP head when received data
<b>AT+CIPUDPMODE</b>	UDP extended mode
<b>AT+CIPRXGET</b>	Get data from network manually
<b>AT+CIPTKA</b>	Set TCP keep alive parameters
<b>AT+CIPMODE</b>	Open transparent mode
<b>AT+CIPCHAN</b>	Enter transparent mode

For detail information, please refer to "SIM7020 Series\_AT Command Manual".



## 3 Bearer Configuration

### 3.1 PDN Auto-activation

//example of PDN Auto-Activation

**AT+CPIN?**

+CPIN:READY

//Check SIM card status

OK

**AT+CSQ**

+CSQ: 20,0

//Check RF signal

OK

**AT+CGREG?**

+CGREG: 0,1

//Check PS service

OK

**AT+CGACT?**

+CGACT: 1,1

//Activated automatically

OK

**AT+COPS?**

+COPS: 0,0,"CHN-UNICOM",9

Check operator info

CHN-UNICOM is operator's name9 is NB-IOT network

OK

**AT+CGCONTRDP**

+CGCONTRDP:  
1,5,"shnbiot","10.250.0.213.255.255.255.0"

//Get APN and IP address from network

OK

### 3.2 APN Manual Configuration

//example of APN manual configuration

```
AT+CFUN=0 //Disable RF
+CPIN: NOT READY

OK
AT*MCGDEFCONT="IP","3GNET" //Configure new APN

OK
AT+CFUN=1 //Enable RF
OK

+CPIN: READY
AT+CGREG? //Inquiry PS service
+CGREG: 0,1

OK
AT+CGCONTRDP //Attached PS domain and got IP address
+CGCONTRDP: automatically
1,5,"3GNET","10.250.0.253.255.255.255.0"

OK
```

Confidential

## 4 DNS parser and PING

### 4.1 DNS Parser

//example of DNS parser

**AT+CDNSGIP=www.baidu.com**

//Got Baidu host name's IP address using DNS parser

OK

+CDNSGIP:

1,"www.baidu.com","111.13.100.92"

### 4.2 PING

//example of ping

**AT+CIPPING="61.135.169.121"**

//Ping this IP address

OK

+CIPPING: 1,61.135.169.121,11,52

+CIPPING: 2,61.135.169.121,2,52

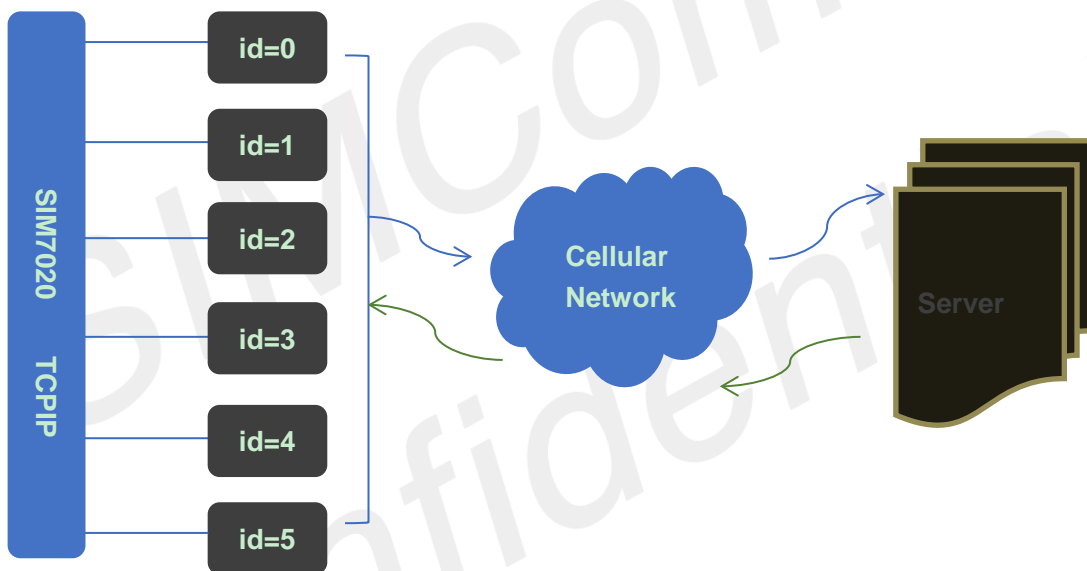
+CIPPING: 3,61.135.169.121,3,52

+CIPPING: 4,61.135.169.121,2,52

## 5 TCPIP Connection

### 5.1 TCPIP Architecture

SIM7020 TCPIP supports multi-client and one-way TCP server architecture, and supports six-way sockets, including TCP or UDP.



### 5.2 TCP client

//example of tcp client

**AT+CSOC=1,1,1**

**+CSOC: 0**

//Created one TCP socket, <socket\_id>=0

OK

**AT+CSOCON=0,5245,"116.247.119.165"**

//Connected remote TCP server

OK

**AT+CSOSEND=0,0,"Hello World"**

//Send TCP data out

OK

**AT+CSOCL=0**

//Close socket

OK

### 5.3 UDP Connection

//example of UDP Connection

```
AT+CSOC=1,2,1 //Created one UDP socket, <socket_id>=0
+CSOC: 0

OK
AT+CSOCON=0,5246,"116.247.119.165" //Connected remote UDP peer
OK
AT+CSOSEND=0,0,"Hello World" //Send UDP data out
OK
AT+CSOCL=0 //Close socket
OK
```

### 5.4 TCP Server

SIM7020 series of module can act as TCP server. Users should use the command "AT+CSOC" to create one socket for server listening, then use the command "AT+CSOB" to bind a local port and the local IP address, which will be the server IP address and port for connection from client. After successfully binding IP address and port, users should use the command "AT+CSOLIS" to open the server listening function. At this point, the server is established successfully, waiting for one client to connect.

//example of TCP server

```
AT+CSOC=1,1,1 //Created one TCP socket, <socket_id>=0
+CSOC: 0

OK
AT+CSOB=0,5678,"10.78.103.220" //Bind local port and local IP address
OK
AT+CSOLIS=0 //Open the server listening function,the listening
port is 5678
OK
+NEW CLIENT CONNEX: 1 //Remote client access
```





OK

## 5.6 Multiple Sockets

//example of multiple sockets

```
AT+CSOC=1,1,1 //Created one TCP socket, <socket_id>=0
+CSOC: 0

OK
AT+CSOC=1,2,1 //Created one UDP socket, <socket_id>=1
+CSOC: 1

OK
AT+CSOCON=0,5245,"116.247.119.165" //Connected remote TCP server
OK
AT+CSOCON=1,5246,"116.247.119.165" //Connected remote UDP peer
OK
AT+CSOSEND=0,0,"Hello World" //Send TCP data out
OK
AT+CSOSEND=1,10,"3132333435" //Send UDP data
OK
AT+CSOCL=0 //Close socket 0
OK
AT+CSOCL=1 //Close socket 1
OK
```

## 5.7 Hex and ASCII Message

Command AT+CSOSEND=<socket\_id>,<len>,<data> supports both Hex and Ascii code message. If <data> is pure hex, the <len> parameter must be configured correct bytes and must be even number. Also Incoming message from remote socket are printed in Hex code.

## 5.8 TCP ACK



//example of TCP ack

```

AT+CSOC=1,1,1 //Created one TCP socket, <socket_id>=0
+CSOC:0

OK
AT+CSOSENDFLAG=1 //Configure TCP ACK report
OK
AT+CSOCON=0,5245,"116.247.119.165" //Connected remote TCP server
OK
AT+CSOSEND=0,0,"Hello World" //Send TCP data out, 11 bytes had been sent out
OK //successfully.

SEND: 0,11
AT+CSOCL=0 //Close socket
OK

```

## 5.9 Incoming Message Indication

//example of incoming message indication

```

+CSONMI: 0,6,313233 //Incoming data "123" from remote side

```

## 5.10 Retention Scene when Module is Waked from PSM Mode

After UDP connection is established, if user want to continue to use this connection when module is waked from PSM mode, the command AT+RETENTION should be set to 1.

//example of retention scene when module is waked from PSM mode

```

AT+RETENTION? //inquiry the flag of retention
+RETENTION: 0
AT+RETENTION=1 //Enable retention scene
OK
AT+CSOC=1,2,1 //Created one UDP socket,
+CSOC:0 //<socket_id>=0

OK

```

```

AT+CSOCON=0,5004,"117.131.85.139" //Connected remote UDP server
OK
AT+CPSMS=1 //Enable PSM mode
OK
+CPSMSTATUS: "ENTER PSM" //Enter into PSM mode
+CPSMSTATUS: "EXIT PSM" //Wake module from PSM mode
AT+CSOSEND=0,0,"Hello World!!!" //Send data to UDP server
OK
+CSOENMI: 0,12,313233343536 //Receive data from UDP server

```

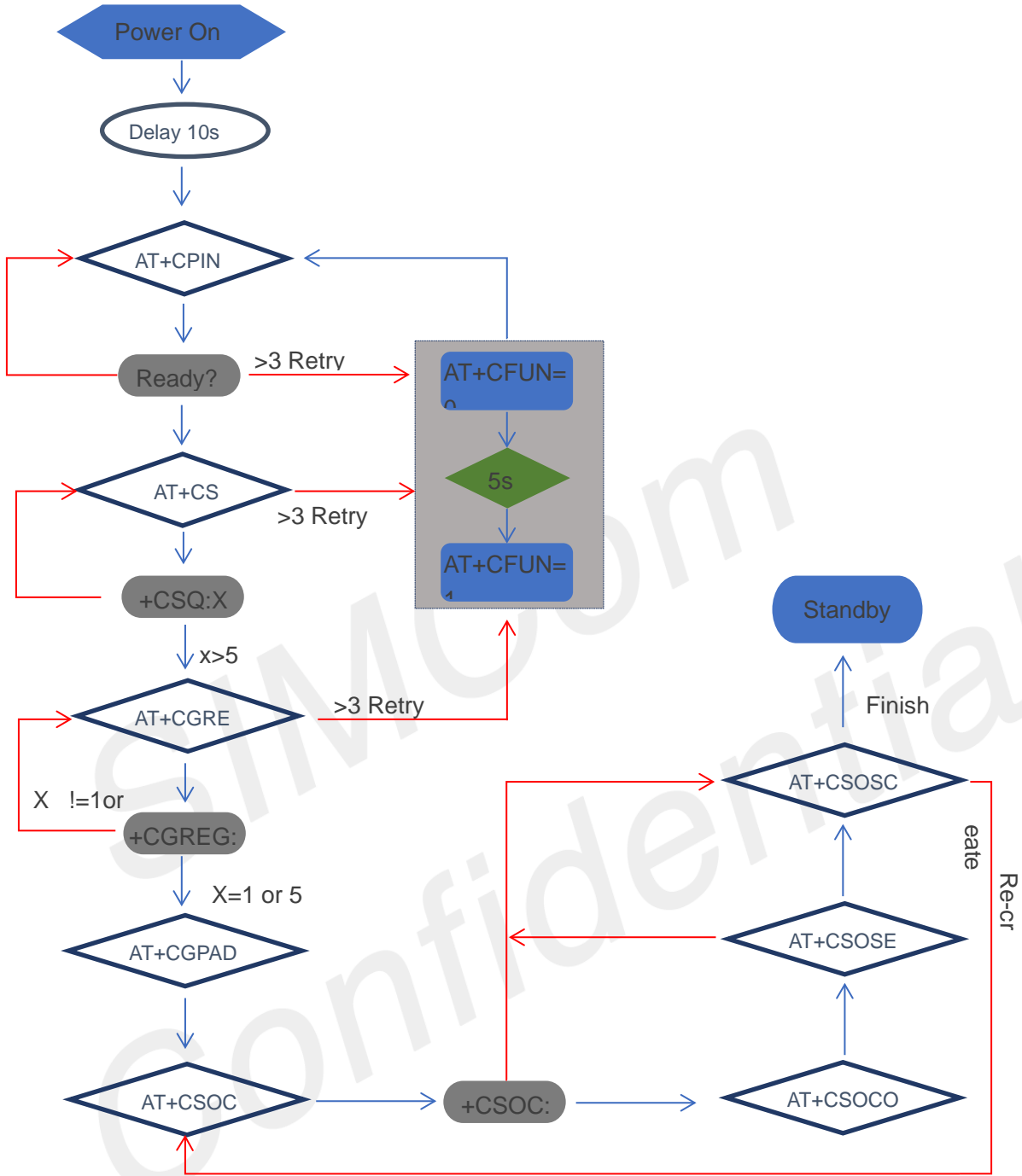
## 5.11 Summary of Socket Error codes

//example of socket error code

```
+CSOERR: 0,2 <socket id>, <error code>
```

Code	Description
-1	Common error
1	Route error (host unreachable)
2	Connection abort error
3	Connection Reset error
4	Connected error
5	Illegal error
6	Buffer error
7	Block error
8	Address in use error
9	Already connecting error
10	Already connected error
11	Bearer error

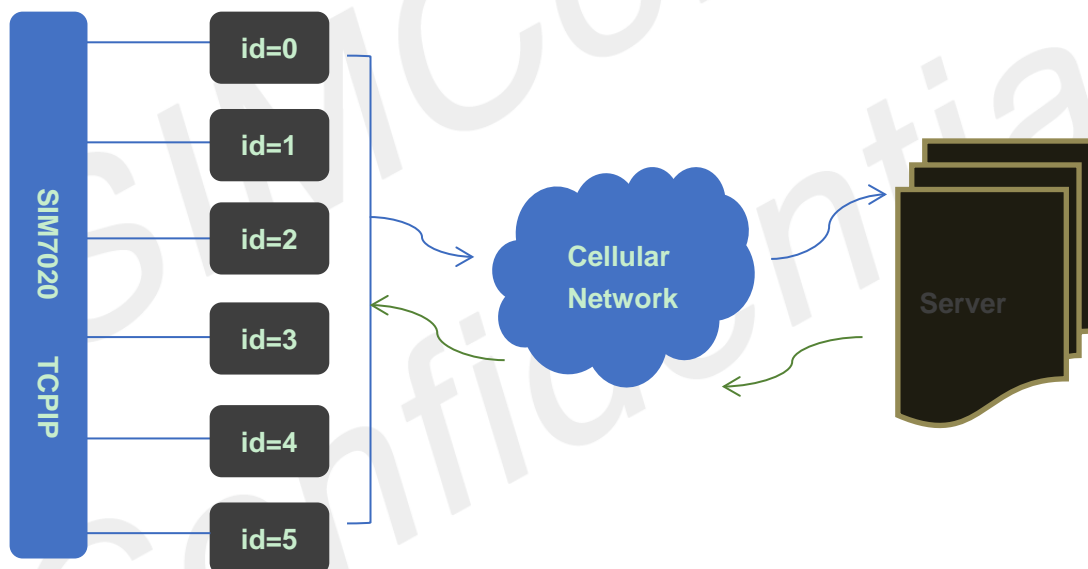
## 5.12 TCPIP Connection flow chart



## 6 TCPIP Application compatible with SIM800 serial module

### 6.1 TCPIP Architecture

SIM7020 TCPIP supports multi-client and one-way TCP server architecture, and supports six-way sockets, including TCP or UDP.



### 6.2 TCP Client Connection

//example of TCP client connection

```

AT+CSTT //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection(GPRS or CSD)
OK
AT+CIFSR //Get local IP address
10.78.245.128
    
```

```

AT+CIPSTART="TCP", "116.228.22
1.51", "8500"
OK

CONNECT OK //The TCP connection has been established successfully
//Send data to remote server, CTRL+Z (0x1a) to send.

AT+CIPSEND
> hello TCP serve
SEND OK //Remote server receives data. For TCP, "SEND OK" means
data has been sent out and received successfully by the remote
server

hello SIM7020 //Received data from remote server
CLOSED //Remote server closed the connection

```

### 6.3 UDP Client Connection

```

//example of UDP client connection

AT+CSTT //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection(GPRS or CSD)
OK
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTART="UDP", "116.228.221.
51", "9600" //Start up the connection
OK

CONNECT OK // The UDP connection has been established successfully
//Send data to remote server, CTRL+Z (0x1a) to send.

AT+CIPSEND
>SIM7020 UDP test
SEND OK //Data has been sent out from the serial port, but it is unknown
if the data reaches the UDP server.

UDP test //Received data from remote server
CLOSED OK //Remote server closed the connection

```

### 6.4 UDP Extended Mode

In UDP extended mode, SIM7020 series can receive UDP data from any IP address and port, meanwhile it can send UDP data to any IP address and port.

```
//example of UDP extended mode
```

```
AT+CSTT //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection(GPRS or CSD)
OK
AT+CIFSR //Get local IP address
10.78.245.128
AT+CLPORT="UDP",8888 //Set local UDP port
OK
AT+CIPSRIP=1 //Display IP address and Port of sender
OK
AT+CIPHEAD=1 //Add IP head in receiving data
OK
AT+CIPUDPMODE=1 //Enable the UDP Extended Mode
OK
AT+CIPSTART="UDP","116.228.221.51",9600 //Startup UDP connection to remote server
OK

CONNECT OK //UDP connection has been established successfully.
AT+CIPUDPMODE? //Check UDP mode's status
+CIPUDPMODE:
1,"116.228.221.51",9600

OK
AT+CIPSEND //Send data to 116.228.221.51: 9600
> Hello 9600
SEND OK
RECVFROM: 116.228.221.51:9600 //Receiving data from 116.228.221.51:9600

+IPD,5:test1
AT+CIPUDPMODE=2,"116.228.221.51",1234 //Re-set UDP port to be sent data to.
OK
AT+CIPSEND //Send data to 116.228.221.51: 1234
> Hello 1234
SEND OK
RECV FROM: 116.228.221.51:1234 //Receiving data from 116.228.221.51: 1234

+IPD,5:test2
AT+CIPUDPMODE=2,"10.78.103.220",9600 //Re-set UDP address and port to be sent data to.
```

```
"5678
OK
AT+CIPSEND //Send data to 10.78.103.220:5678
> Hello 5678
SEND OK
RCV FROM: 10.78.103.220: 5678 //Receiving data from 10.78.103.220: 5678

+IPD,5:test3
AT+CIPUDPMODE=2,"211.136.131.6 //Re-set UDP address to be sent data to.
5",4500
OK
AT+CIPUDPMODE? //The destination UDP address has been Updated, and UDP
+CIPUDPMODE: extended mode is still on.
1,"211.136.131.65",4500

OK
AT+CIPSEND //Send data to 211.136.131.65:4500
>Hello 4500
SEND OK
```

## 6.5 TCP Server

In single connection mode, when configured as TCP server, SIM7020 series allows one client to connect in. User should use the command "AT+CIPSERVER=1,"<port>" to start the server function and set listening port. If it is successful, response "SERVER OK" will be returned, and now the server starts to listen to the TCP port. And then, user should use command "AT+CIFSR" to get local IP address, which is the server IP address.

If the client connects to the server successfully, the IP address of remote client will be displayed at server side. And the server can receive TCP data from remote side. Also user can use command AT+CIPSEND to send data to remote client. User can use "AT+CIPSERVER=0" to close the listening status. Also, user can use AT+CIPCLOSE to close the TCP connection.

```
//example of TCP server

AT+CGATT? //GPRS Service's status
+CGATT: 1

OK
AT+CIPSERVER=1,1234 //Start the TCP server, listening port:1234
OK

SERVER OK //TCP server is established successfully
```

```

AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTATUS //Query current status
OK

STATE: SERVER LISTENING //TCP server : listening
REMOTE IP: 10.78.103.220 //Remote client 10.78.103.220 connected in
hello server //Receive data from remote client
AT+CIPSEND //Send data to remote client
> hello client
SEND OK
AT+CIPSERVER=0 //Just close the listening status, the current connection is still
OK active.

SERVER CLOSE //TCP server is closed
hello server //Receive data from remote client
AT+CIPCLOSE //Close TCP connection
CLOSE OK

```

## 6.6 Multi Clients Connection

Being a client, SIM7020 series can establish both TCP and UDP connection to remote server. Total 6 connections are supported. The command "AT+CIPSTART=<n>,<mode>,<IP Address>,<port>" is needed. When the connection is established successfully, "<n>, CONNECT OK" will be returned. And then user can use command AT+CIPSEND=<n> to send data to the connection <n>. User should write data after the promoting mark ">" and use CTRL+Z (0x1a) to send. If sending is successfully, "<n>, SEND OK" will be returned.

//example of multi clients connection

```

AT+CIPMUX=1 //Enable multi-connection
OK
AT+CSTT //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTART=0,"TCP","116.228.2
21.51","8500" //Establish a TCP connection, connection number 0
OK

```



```

0,CONNECT OK
AT+CIPSTART=1,"UDP","116.228.221.51","9600" //Establish a UDP connection, connection number 1
OK

1,CONNECT OK
AT+CIPSEND=0 //Send data to connection 0
>TCP test
0,SEND OK
AT+CIPSEND=1 //Send data to connection 1
>UDP test
1,SEND OK
+RECEIVE,0,16: //Received data from connection 0, data length 16
SIM7020 TCP test
+RECEIVE,1,16: //Received data from connection 1, data length 16
SIM7020UDP test
AT+CIPSTATUS //Query the current connection status
OK

STATE: IP PROCESSING
C:
0,0,"TCP","116.228.221.51","8500",
"CONNECTED"
C:1,0,"UDP","116.228.221.51","960
0","CONNECTED"
C: 2,,"","",""INITIAL"
C: 3,,"","",""INITIAL"
C: 4,,"","",""INITIAL"
C: 5,,"","",""INITIAL"

```

## 6.7 Multi Mixing Connection

Being a TCP server, SIM7020 series allows remote clients to connect in; meanwhile, it can establish TCP/UDP connections to remote servers as well.

Before launching the server function, command group “AT+CSTT, AT+CIICR, AT+CIFSR” should be executed to activate the PDP context and get local IP address. Then user can send command “AT+CIPSERVER=1, <port>” to start the server function. If it is successfully, response “SERVER OK” will be returned, and now the server starts to listen to the TCP port.

If the clients connect to the server successfully, the IP addresses of remote clients together with connection numbers <n> will be displayed at server side. Then the server can receive TCP data from remote clients. Also user can use command AT+CIPSEND=<n> to send data to remote client <n>. Simultaneously, user can connect the server SIM800 series to remote servers by TCP/UDP using command

“AT+CIPSTART=<n>,<mode>,<IP Address>,<port>”. Command “AT+CIPSERVER=0” can be used to close the listening status. User can use the command AT+CIPCLOSE=<n> to close one specific connection with number <n> and use AT+CIPSHUT to close all connections.

//example of multi mixing connection

```

AT+CIPMUX=1 //Enable multi-connection
OK
AT+CSTT //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSERVER=1,8888 //Start server;listeninig port:8888
OK

SERVER OK

0,REMOTE IP: 10.76.40.73 //Remote client connect in, connection number 0
allocated
+RECEIVE,0,26: //Received data from remote client, data length 26
connection TCP server test
AT+CIPSEND=0 //Send data to remote client
>TCP test
0,SEND OK
AT+CIPSTART=1,"TCP","116.228.221.51 //Establish TCP connection to remote server
","8500"
OK

1,CONNECT OK
AT+CIPSTART=2,"UDP","116.228.221.51 //Establish UDP connection to remote server
","9600"
OK

2,CONNECT OK
AT+CIPCLOSE=2 //Close the UDP connection with remote server
2,CLOSE OK
AT+CIPSTATUS //Query current connection status
OK

STATE: IP PROCESSING
S: 0,0,"8888","LISTENING"
C:
0,0,"TCP","10.76.40.73","2020","CONNE
CTED"

```

```

C:
1,0,"TCP","116.228.221.51","8500","CONN
NECTED"
C:
2,0,"UDP","116.228.221.51","9600","CL
OSED"
C: 3,,"","","","INITIAL"
C: 4,,"","","","INITIAL"
C: 5,,"","","","INITIAL"
AT+CIPSERVER=0 //Close server listening
OK

SERVER CLOSE
AT+CIPSHUT //Deactivate the PDP context &close all connections
SHUT OK

```

## 6.8 DNS Parser and Connection

```

//example of DNS parser and connection

AT+CSTT //Startup task and set APN
OK

AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)

AT+CIFSR //Get local IP address
10.78.245.128

AT+CDNSGIP=www.baidu.com //Query IP address of www.baidu.com
OK

+CDNSGIP: //Parsing succeed, the IP address has two results.
1,"www.baidu.com","119.75.218.77","119.75.
217.56"

AT+CDNSGIP="abctest" //Query "abctest"
OK

+CDNSGIP: 0,8 //Parsing error.
AT+CIPSTART="TCP","WWW.SIM.COM",80 //Establish TCP connection.
OK

CONNECT OK
AT+CIPSND //Send data

```

> DNS test

SEND OK

## 6.9 Data Sending Related

SIM7020 series provides 3 ways to send data: changeable data length sending, fixed data length sending and timed sending. SIM7020 series also provides a method to let user know how much data is sent out from the module and received by remote server on an active TCP connection.

### 6.9.1 Data Sending Related

SIM7020 series provides 3 ways to send data: changeable data length sending, fixed data length sending and timed sending. SIM7020 series also provides a method to let user know how much data is sent out from the module and received by remote server on an active TCP connection.

### 6.9.2 Fixed Length Sending

User can send the fixed length of data with "AT+CIPSEND=<LENGTH>", then input data after getting promoting mark ">". Data will be sent automatically when the length of the input data equals to the value "LENGTH". User does not need the terminal symbol CTRL+Z (0x1a) in this case. For multi connection mode, the command is "AT+CIPSEND=<n>,<LENGTH>".

### 6.9.3 Select Data Transmitting Mode

SIM7020 series supports quick sending mode.

When command "AT+CIPQSEND=0", it is in normal sending mode. In this mode, after user sends data by "AT+CIPSEND", if the server receives TCP data, it will give ACK message to module, and the module will respond "SEND OK".

When command "AT+CIPQSEND=1", it is in quick sending mode. When the data is input to the serial port of module by "AT+CIPSEND", it will respond "DATA ACCEPT:", while not respond "SEND OK". In such case, user can continuously use "AT+CIPSEND" to send data to the server.

**Single connection:**

```
//example of single connection
```

```
AT+CIPQSEND=1
```

```
//Enable quick sending mode
```

```
OK
AT+CIPSEND //Data has been sent, not sure whether to be
>hello accepted
DATA ACCEPT: 5
```

#### Multi connection:

```
//example of multi connection
```

```
AT+CIPSTART=0,"TCP","116.236.221.75",510 //Establish TCP connection
7
OK

0,CONNECT OK
AT+CIPQSEND=1 //Enable quick sending mode
OK
AT+CIPSEND=0 //Data has been sent, not sure whether to be
> 1234567890 accepted
DATA ACCEPT: 0,10
```

#### 6.9.4 Query Data Transmitting Amount

The command "AT+CIPACK" is used to query previous connection data transmitting state. In single connection, the execution command "AT+CIPACK" will return "+CIPACK: <txlen>,<acklen>, <nacklen>".

---- The 1st parameter <txlen> is the data amount which has been sent;

---- The 2nd parameter <acklen> is the data amount confirmed successfully by the server;

---- The 3rd parameter <nacklen> is the data amount without confirmed by the server.

As long as the connection is still active, user can know how much TCP data user sent to server and how much is received successfully by the server in total. By this means, user can get the total data transmitting amount.

```
//example of query data transmitting amount
```

```
AT+CIPQSEND=1 //Enable quick sending mode
OK
AT+CIPSTART="TCP","116.236.221.75",5107 //Establish TCP connection
OK

CONNECT OK
AT+CIPSTATUS //Query connection's status
OK

STATE: CONNECT OK
```

```
AT+CIPSEND
> 012345678912
DATA ACCEPT: 12
AT+CIPACK //12 be Send , 12 be confirmed
+CIPACK: 12,12,0

OK
```

For multi connection, the correct command type is “AT+CIPACK=<n>”.  
<n> is the connection number.

//example of query data transmitting amount for multi connection

```
AT+CIPQSEND=1 //Enable quick sending mode
OK
AT+CIPSTATUS
OK

STATE: IP PROCESSING
C: 0,"","","","INITIAL"
C: //Query connection's status
1,0,"TCP","116.228.221.51","8500","CONNECTED"
C:
2,0,"UDP","116.228.221.51","9600","CONNECTED"
C: 3,"","","","INITIAL"
C: 4,"","","","INITIAL"
C: 5,"","","","INITIAL"
AT+CIPSEND=1 //Send data at 1 channel
>TCP
DATA ACCEPT: 1,3
AT+CIPACK=1 //3 be send, 3 be confirm
+CIPACK: 3,3,0

OK
AT+CIPSEND=2 //Send data at 2 channel
>UDP
DATA ACCEPT: 2,3
AT+CIPACK=2 //3 be send, 3 be unconfirmed.
+CIPACK: 3,0,3

OK
```

## 6.10 Data Receiving Related

### 6.10.1 Receive Data Automatically

The module will receive data automatically if there is data coming from remote server. Several commands can help to get the information header.

- “AT+CIPHEAD=1” helps to add IP header in the format “+IPD (data length): payload”.
- “AT+CIPSRIP=1” helps to show the data source information in the format “RECV FROM: <IP ADDRESS>:<PORT >”.
- “AT+CIPSHOWTP” helps to show the protocol (TCP/UDP) in the IP header. It takes effect only if “CIPHEAD” is enabled.

With this information, user can easily know the source of the data frame, the amount of the payload and the protocol. It can also help user to distinguish the received data from AT command responses.

### 6.10.2 Receive Data Manually

The module provides user a way to get data from the network manually instead of pushing data to the TE automatically.

“AT+CIPRXGET=1” is used to enable getting data from network manually, which should be set before connection. If it is set to “0” (default value), data will be pushed to TE directly.

“AT+CIPRXGET=<mode>[,<len>]” is used to get data with a given length. If it is multi IP connection, the connection ID should be given. E.g.: “AT+CIPRXGET=<mode>,<id>[,<len>]”

```

AT+CIPRXGET=1 //Enable getting data from network manually
OK
AT+CIPSTART="TCP","116.228.221.51",5555 //Establish TCP connection
OK

CONNECT OK
+CIPRXGET:1 //Data incoming from server
//The mode is set to 2, the output data will be in
//normal mode, with the length not exceeding1460
//bytes at a time.
AT+CIPRXGET=2,1460
+CIPRXGET:2,11,0
HELLO WORLD
OK

```

```

+CIPRXGET:1 //Data incoming from server
//The mode is set to 3, user can get data in HEX
//mode with the length not exceeding 730 bytes at a
//time.

AT+CIPRXGET=3,730

+CIPRXGET:3,11,0
48454C4C4F20574F524C44
OK
  
```

## 6.11 GPRS States Exchange Related

For single connection, there are 10 GPRS states in total; for multi connection, there are 7 GPRS states. After some AT commands are executed, the corresponding state will be changed. User can get a general idea from the following diagrams:

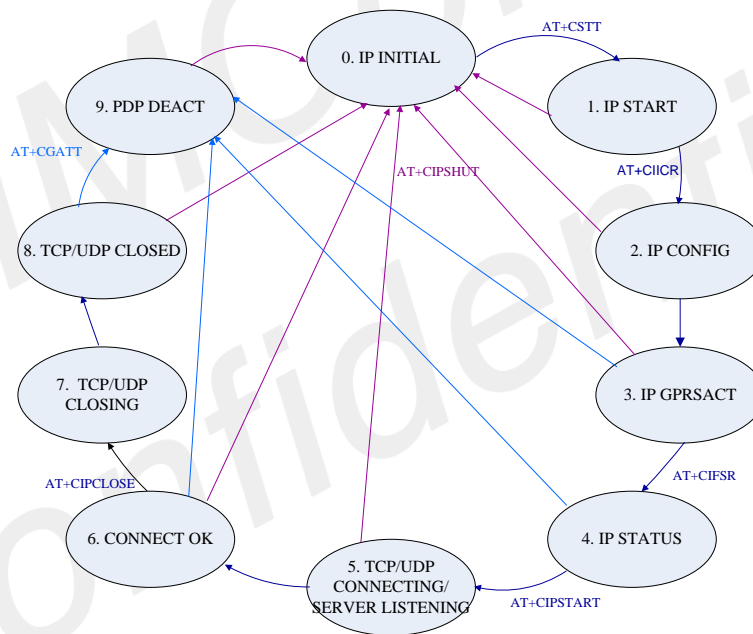
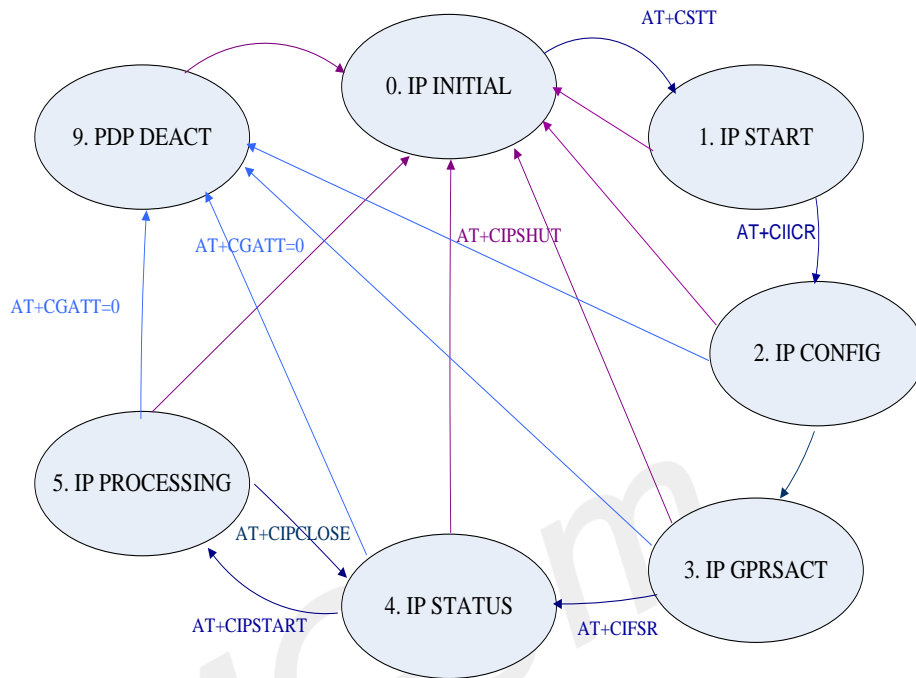


Figure3: GPRS States Diagram for single connection





**Figure4: GPRS States Diagram for multi connection**

- IP INTIAL: GPRS initial status
- IP START: Start a TCP/UDP task
- IP CONFIG: Configure PDP context
- IP GPRSACT: Context active already
- IP STATUS: Get local IP address
- TCP/UDP CONNECTING: Connecting to server now
- SERVER LISTENING: Listening to server port now
- IP PROCESSING: Processing the existing connection now
- CONNECT OK: Connection to the server is successful
- TCP/UDP CLOSING: Closing connection now
- TCP/UDP CLOSED: Connection closed (local IP/PDP context still there)
- PDP DEACT: Context deactivated

## 6.12 Connection Closing Related

User can use the command “AT+CIPCLOSE=<mode>” to close the TCP or UDP connection.

If <mode> is “0”, it is slow closing;

If <mode> is “1”, it is quick closing.

In slow closing, the module will interactive with the server when it closes the TCP connection. Thus, the time of returning “CLOSE OK” will be a bit long. This method is suitable for steady network.

In quick closing, the module will disconnect the connection compulsorily and return “CLOSE OK”

immediately, without interaction with the server.

The default setting is slow closing, so the “<mode>=0” can be omitted. And in multi connection, the connection number <n> should be added in front of <mode>.

Be noted that command “AT+CIPCLOSE” only closes current TCP/UDP connection, but PDP context is still active. Also user can close connection by AT+CIPSHUT, with current PDP context being deactivated.

## 6.13 Error Handling

If an error occurs in TCP/UDP connection, for example TCP sending data error or TCP connection dropping, it is suggested to close the connection by command “AT+CIPCLOSE” and then restart the connection by “AT+CIPSTART”.

If the error still occurs, command “AT+CIPSHUT” is recommended to shut off the PDP context and then restart the connection.

If these two methods above can't help to solve it, SIMCom recommends user to reset the module.

## 6.14 Transparent Mode

### 6.14.1 What is Transparent Mode

SIM7020 series supports transparent mode which provides a special data mode for data receiving and sending by TCP/IP application task. Once the connection is established under transparent mode, the module will be in data mode. All received data from serial port will be treated as data packet to be transferred later, similarly all data received from remote side will be sent to serial port directly. In transparent mode, all AT commands are not available. Methods are provided to switch back and forth between data mode and command mode. Once it is switched to command mode, all AT commands can be used again.

### 6.14.2 How to enable Transparent Mode

To enable transparent mode, the command AT+CIPMODE should be set to 1. The transparent mode is only supported under single connection.

```
AT+CIPMUX=0
```

```
//Enable single-connection
```

```
OK
AT+CIPMODE=1 //Enable transparent mode
OK
```

### 6.14.3 How to Establish Connection and Enter into Transparent Mode after Enable Transparent Mode

After enable transparent mode, SIM7020 series can work as two modes too: TCP client, UDP client. Once the connection is established, "CONNECT OK" will be returned in the serial port. After establish connection, executing "AT+CIPCHAN" to enter into transparent mode.

```
AT+CIPSTART="TCP","116.228.221.51","8500" //Start up the connection
OK The TCP connection has been established
CONNECT OK successfully
AT+CIPCHAN //Enter into transparent mode successfully
CONNECT
1234567890 //Receive data from server
123456 //Send data to server
```

### 6.14.4 How to Switch Between Data Mode and Command Mode

To switch from data mode to command mode, following methods are available:

- (1) The default escape sequence is +++ , and to use this sequence, there should be 1000ms idle period before this sequence and 1000ms idle period after this sequence. Besides, the interval between each + should not exceed 1000ms, otherwise it will be treated as TCP/IP data.
- (2) If the remote server closes the connection, the module will be switched back to command mode automatically.
- (3) If the module is deactivated from PDP context (+PDP DEACT) during data transferring, module will be switched back to command mode automatically.

ATO command can be used to switch the module from command mode to data mode again if the connection is active.

```
+++ //switch from data mode to command mode
OK
```

```
AT+CSQ //AT command work normally
+CSQ: 25,0

OK
ATO //switch the module to data mode
CONNECT
123456 //Send data to server
1234567890 //Receive data from server
```

SIMCom  
Confidential