



# SIM800 Series\_TCPIP \_Application Note

GPRS 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>	SIM800 Series_TCPIP_Application Note
<b>Version:</b>	1.03
<b>Date:</b>	2020-06-18
<b>Status:</b>	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**

Building B, SIM Technology Building, No.633 Jinzhong 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](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	2013-08-01	Ping.zhang	New version
V1.01	2013-10-12	Ping.zhang	Modify example of as a client(chapter 3.1)
			Add ":" after "+RECEIVE,X,X"
			Modify "AT+CIPSHOWT" to "AT+CIPSHOWTP"
V1.02	2016-11-17	Wenjie.lai	Scope
V1.03	2020-06-15	Fumei.zeng /Wenjie.Lai	Change the style

## Scope

This document presents the AT command of TCPIP stack and application examples.  
This document can apply to SIM800 series modules with TCPIP function.

# 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 Structure</b> .....	<b>7</b>
<b>3 Single Connection</b> .....	<b>8</b>
3.1 Non-transparent mode .....	8
3.1.1 How to Establish a TCP Client Connection .....	8
3.1.2 How to Establish a UDP Client Connection .....	9
3.1.3 How to Establish a TCP Server Connection .....	10
3.1.4 UDP Extended Mode .....	11
3.2 Transparent Mode .....	12
3.2.1 What is Transparent Mode .....	13
3.2.2 How to Configure Transparent Mode .....	13
3.2.3 How to Establish Connection under Transparent Mode .....	13
3.2.4 How to Switch Between Data Mode and Command Mode .....	14
3.2.5 How to handle incoming call and short message in data Mode .....	14
3.3 Fix Local Port of TCP/UDP Client .....	15
<b>4 Multi Connection</b> .....	<b>16</b>
4.1 As a Client .....	16
4.2 As a TCP Server .....	17
<b>5 Two GPRS Context</b> .....	<b>20</b>
<b>6 DNS Function</b> .....	<b>22</b>
<b>7 Data Sending Related</b> .....	<b>23</b>
7.1 Fixed Length Sending .....	23
7.2 Timed Sending .....	23
7.3 Query Available Data Length to be Sent .....	23
7.4 Select Data Transmitting Mode .....	24
7.5 Query Data Transmitting Amount .....	25
<b>8 Data Sending Related</b> .....	<b>27</b>
8.1 Receive Data Automatically .....	27
8.2 Receive Data Manually .....	27
<b>9 GPRS States Exchange Related</b> .....	<b>29</b>
<b>10 GPRS States Exchange Related</b> .....	<b>31</b>
<b>11 Connection Activity Checking Related</b> .....	<b>32</b>

12 Power Consumption with Existing Connection .....33  
13 Error Handling.....34

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] SIM800 Series\_AT Command Manual

## 1.3 Conventions and abbreviations

Abbreviation	Description
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
APN	Access Point Name
GPRS	General Packet Radio Service
PDP	Packet Data Protocol
DNS	Domain Name System

## 2 Structure

There are two modes of connection for SIM800 series TCP/IP application: Single connection and Multi connection. When in single connection mode, SIM800 series can work at both transparent mode and non-transparent mode; and under these two transmission modes, SIM800 series can be configured as either TCP/UDP client or TCP server. When in multi connection mode, SIM800 series can only work at non-transparent mode. In this mode, SIM800 series can work as an absolute TCP/UDP client, which can establish 6 connections in total. In this mode, it can also be configured as one TCP server, which allows 5 TCP/UDP clients to be connected; and the TCP server also can act as a client, establishing 5 connections to one remote server. The structure of the TCP/IP application is given as below.

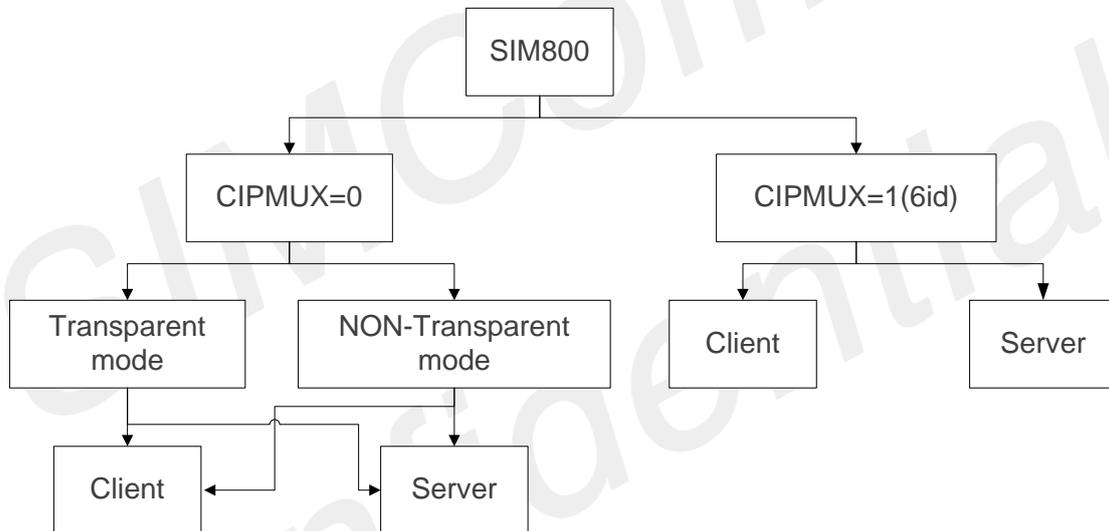


Figure1: SIM800 series TCPIP Structure

## 3 Single Connection

Command AT+CIPMUX=<n> is used for selecting TCPIP connection mode, when n=0, it is single connection; when n=1, it is multi connection. The default configuration is single connection mode. For single connection, SIM800 series supports both transparent mode and non-transparent mode.

### 3.1 Non-transparent mode

Command AT+CIPMODE=<n> is used for selecting TCPIP application mode, when n=0, it is non-transparent mode (normal mode); when n=1, it is transparent mode. The default configuration is non-transparent mode. There are three working modes for SIM800 series under this mode: TCP client, UDP client and TCP server.

#### 3.1.1 How to Establish a TCP Client Connection

Firstly, before any TCP/UDP related operation is set up, the module should be connected to GSM or GPRS network. User can use the commands "AT+CREG?" and "AT+CGATT?" to query the GSM network registration status whether the module has been attached to GPRS service.

```
//Example of establish a TCP client connection
```

```
AT+CPIN? //whether some password is required or not
```

```
+CPIN: READY
```

```
OK
```

```
AT+CSQ //received signal strength
```

```
+CSQ: 20,0
```

```
OK
```

```
AT+CREG? //the registration of the ME.
```

```
+CREG: 0,1
```

```
OK
```

```
AT+CGATT? //GPRS Service's status
```

```
+CGATT: 1
```

```
OK
```

Secondly, user should use the command group AT+CSTT, AT+CIICR and AT+CIFSR to start the task and activate the wireless connection. Lastly, user can establish TCP connection between SIM800 series and the server by AT command (AT+CIPSTART="TCP", "IP Address of server", "port number of server"). If the connection is established successfully, response "CONNECT OK" will come up from the module. Now user can send data to the server with "AT+CIPSEND". "AT+CIPSEND" will return promoting mark ">", user should write data after ">" then issue CTRL+Z (0x1a) to send. If sending is successful, it will respond "SEND OK". And if there is data coming from the server, the module will receive the data automatically from the serial port. User can close the TCP connection with "AT+CIPCLOSE" command. Below is an example of TCP connection to remote server.

//Example of TCP connection to remote server.

```

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

OK
AT+CSTT="CMNET" //Start task and set APN.
OK //The default APN is "CMNET", with no username or
password. Check with local GSM provider to get the
APN.

AT+CIICR //Bring up wireless connection (GPRS
OK or CSD)
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTART="TCP", "116.228.221.51", "8 //Start up the connection
500"
OK
CONNECT OK //The TCP connection has been established
successfully
AT+CIPSEND //Send data to remote server, CTRL+Z (0x1a) to send.
> hello TCP serve //User should write data only after the promoting mark
">", and then use CTRL+Z to send. User can use
command "AT+CIPSPRT" to set whether echo
promote ">" after issuing "AT+CIPSEND".

SEND OK //Remote server receives data. For TCP, "SEND OK"
means data has been sent out and received
successfully by the remote server, due to the TCP
connection-oriented protocol;

Hello SIM800 //Received data from remote server
CLOSED //Remote server closed the connection

```

### 3.1.2 How to Establish a UDP Client Connection

The process of establishing UDP connection is similar to TCP.

```
//Example of UDP connection to remote server.
AT+CGATT? //GPRS Service's status
+CGATT: 1

OK
AT+CSTT="CMNET" //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection (GPRS
OK //or CSD)
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTART="UDP","116.228.221.51",9 //Start up the "UDP" connection
600"
OK
CONNECT OK //The UDP connection has been established
successfully
AT+CIPSEND //Send data to remote server, CTRL+Z (0x1a) to send.
> SIM800 UDP test
SEND OK //Data has been sent out from the serial port, but it is
unknown if the data reaches the UDP server. for UDP,
"SEND OK" just means data has been sent out from
the serial port of module, not meaning data reaching
the server, due to the UDP simpler message-based
connectionless protocol.

UDP test //Receive data from remote server
AT+CIPCLOSE //Close the UDP connection
OK
CLOSED OK //Connection is closed
```

### 3.1.3 How to Establish a TCP Server Connection

In single connection mode, when configured as TCP server, SIM800 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 establish a TCP server connection
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
10.78.33.227
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 //Remote client receives data
AT+CIPSERVER=0 //Just close the listening status, the current connection
OK is still active.
SERVER CLOSE //TCP server is closed
hello SIM800 //Receive data from remote client
AT+CIPCLOSE //Close TCP connection
CLOSE OK
```

### 3.1.4 UDP Extended Mode

In UDP extended mode, SIM800 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+CGATT? //GPRS Service's status
+CGATT: 1

OK
AT+CSTT="CMNET" //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection (GPRS
OK or CSD)
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" //Start up UDP connection to remote server
600"
OK
CONNECT OK //UDP connection has been established successfully.
AT+CIPUDPMODE? //Check UDP mode's status
+CIPUDPMODE: 1,"116.228.221.51",9600

OK
RECV FROM: 116.228.221.51:9600 //Receiving data from 116.228.221.51:9600

+IPD,5:test1
RECV FROM: 116.228.221.51:1234 //Receiving data from 116.228.221.51: 1234

+IPD,5:test2
RECV FROM: 10.78.103.220: 5678 //Receiving data from 10.78.103.220: 5678

+IPD,5:test3

AT+CIPSEND //Send data to 116.228.221.51:9600
> Hello 9600
SEND OK
AT+CIPUDPMODE=2,"211.136.131.65",4500 //Re-set UDP address to be sent data to.
0
OK
AT+CIPUDPMODE? //The destination UDP address has been
+CIPUDPMODE: 1,"211.136.131.65",4500 updated, and UDP extended mode is
still on.

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

```

## 3.2 Transparent Mode

### 3.2.1 What is Transparent Mode

SIM800 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.

**Note:** *The flow control is default off. It is better to open the hardware flow control for using transparent mode, by setting AT+IFC=2,2.*

### 3.2.2 How to Configure Transparent Mode

To enable transparent mode, the command AT+CIPMODE should be set to 1. In transparent mode, the command AT+CIPCCFG is used for configuring transfer mode, which has 7 parameters NmRetry, WaitTm, SendSz, Esc, Rxmode, RxSize, Rxtimer.

- NmRetry: Number of retries to be made for an IP packet.
- WaitTm: Number of 200ms intervals to wait for serial input before sending the packet
- SendSz: Size in bytes of data block to be received from serial port before sending.
- Esc: Whether turn on the escape sequence, default is TRUE.
- Rxmode: Whether to set time interval during output data from serial port.
- RxSize: Output data length for each time, default value is 1460.
- Rxtimer: Time interval (ms) to wait for serial port to output data again. Default value: 50ms

### 3.2.3 How to Establish Connection under Transparent Mode

The transparent mode is only supported under single connection. In transparent mode, SIM800 series can work as three modes too: TCP client, UDP client and TCP server. Once the connection is established, CONNECT will be returned in the serial port and the module will be in data mode. At the same time, DCD pin will be active low. Below is an example of TCP client connection.

```
//Example of TCP client connection
```

```
AT+CGATT?
```

```
//GPRS Service's status
```

```
+CGATT: 1
```

```
OK
```

```
AT+CIPMODE=1
```

```
//Enable transparent mode
```

```
OK
```

```

AT+CSTT="CMNET" //Start task and set APN.
OK
AT+CIICR //Bring up wireless connection (GPRS
OK or CSD)
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSTART="TCP","116.228.221.51","8 //Start up TCP connection to remote server
500"
OK //Connected successfully, enter into data mode
CONNECT
..... //Input data to serial port, no echo, so can't see input
OK data
//Quit data mode by pulling DTR or using "+++"
ATO //Return to data mode
CONNECT //Data received from remote server
SIM800 channel test //Closed by remote server, quit data mode
CLOSED

```

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

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

- (1) The escape sequence can be used if the fourth parameter of AT+CIPCCFG is TRUE. 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) DTR line of serial port can also be used. To use this method, AT&D1 should be set firstly. Pull DTR line to ground for at least 1 second and then pull up, the module will be switched from data mode to command mode and OK will be returned indicating the module is in command mode.
- (3) For TCP client connection, if the remote server closes the connection, the module will be switched back to command mode automatically.
- (4) For TCP server connection, if the remote client closes the connection, the module will be switched back to command mode automatically.
- (5) 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 if the connection is active, and CONNECT will be returned again.

### 3.2.5 How to handle incoming call and short message in data Mode

RI pin of serial port will give a 50ms low pulse when a call is coming during data mode. RI pin will give a 120ms low pulse when a short message is received.

To handle the incoming call or SMS, the module need to enter command mode first (using DTR or +++), and then the module will give the normal unsolicited responses as following:

```
//Example of handle incoming call and short message in data mode
RING //Incoming call
+CMTI: "SM", 10 //New short message
```

### 3.3 Fix Local Port of TCP/UDP Client

Normally, if client SIM800 series establishes TCP/UDP connection to remote server, the TCP/UDP connection will be dynamically allocated a local port when AT+CLPORT is set to 0. If AT+CLPORT is set to a nonzero value, the local port will be fixed at this value. The default configuration of this command is 0. Be aware that this command will be effective only in single connection mode and when the module is as a client.

SIMCom  
Confidential

## 4 Multi Connection

To enable multi connection mode, the command AT+CIPMUX must be set to 1. Be aware that in the multi connection mode, SIM800 series only supports non-transparent mode. Due to this matter, the command AT+CIPMODE is invalid under multi connection mode.

In multi connection mode, SIM800 series can work as a TCP/UDP client or a TCP server. As a client, it can establish 6 connections to remote server in total, both TCP and UDP.

As a TCP server, it allows remote clients to connect in; at the same time, it can establish TCP/UDP connections to upper remote servers. Total 5 available connections are supported, with the server itself occupying one connection.

### 4.1 As a Client

Being a client, SIM800 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. User can use command AT+CIPCLOSE=<n> to close one specific connection with number <n>. Also user can send AT+CIPSHUT to close all connections.

```
//Example of client
AT+CGATT? //GPRS Service's status
+CGATT: 1

OK
AT+CIPMUX=1 //Enable multi connection
OK
AT+CSTT="CMNET" //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=0,"TCP","116.228.221.51", //Establish a TCP connection, connection number 0
"8500"
```

OK

0, CONNECT OK

**AT+CIPSTART=1,"UDP","116.228.221.51",** //Establish a UDP connection, connection number 1  
**"9600"**

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,0,17:** //Received data from connection 0, data length 17

**SIM800 TCP test** //Received data from connection 1, data length 17

**+RECEIVE,1,17:** //Connection 0 is closed by remote server

**SIM800 UDP test**

0, CLOSED

**AT+CIPSTATUS** //Query the current connection status

OK

STATE: IP PROCESSING

C: 0,0,"TCP","116.228.221.51","8500","  
CLOSED "

C: 1,0,"UDP","116.228.221.51","9600","  
CONNECTED "

C: 2,,"","","INITIAL "

C: 3,,"","","INITIAL "

C: 4,,"","","INITIAL "

C: 5,,"","","INITIAL "

## 4.2 As a TCP Server

Being a TCP server, SIM800 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 TCP server
AT+CGATT? //GPRS Service's status
+CGATT: 1

OK
AT+CIPMUX=1
OK
AT+CSTT="CMNET"
OK
AT+CIICR //Bring up wireless connection (GPRS
or CSD)
OK
AT+CIFSR //Get local IP address
10.78.245.128
AT+CIPSERVER=1,8888 //Start server;listening port:8888
OK

SERVER OK
0, REMOTE IP: 10.76.40.73 //Remote client connect in, connection number 0
allocated
+RECEIVE,0,39: //Received data from remote client, data length 39
SIM800 multi connection TCP server test
AT+CIPSEND=0 //Send data to remote client
> hello client
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+CIPSEND=1

//Send TCP data to remote server

> data from connection 1

1, SEND OK

+RECEIVE,1,20:

//Received TCP data from remote serve

data to connection 1

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"

//TCP server is in the process of listening, occupying one connection

C:

//A client is accepted

0,0,"TCP","10.76.40.73","2020","CONNEC

//TCP connection is connected

TED"

C:

1,0,"TCP","116.228.221.51","8500","CONN

ECTED"

C:

2,0,"UDP","116.228.221.51","9600","CLOS

//UDP connction is closed

ED"

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

## 5 Two GPRS Context

SIM800 series supports the use of two different APN, that is , two different GPRS context for data connection. This mode must be used in multi connection. Customer activated GPRS in first Context , do some connection. When need second Context to data connection, use “at+cipsgtxt=1” to switch, after this, active the second Context and do data connexion.

//Example of two GPRS Context

```

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

OK
AT+CIPMUX=1 //Set multi connection mode
OK
AT+CSTT="CMNET" //Start task and set APN."CMNET"
OK
AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)
AT+CIFSR //Get local IP address
10.126.186.232
AT+CIPSTART=0,"TCP","116.236.221.75", //Establish TCP connection at 0 channel in first Context
5555
OK

0, CONNECT OK
AT+CIPSGTXT=1 //Switch to second Context
OK
AT+CSTT="CMWAP" //Start task and set APN." CMWAP"
OK
AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)
AT+CIFSR //Get local IP address
10.204.70.11
AT+CIPSTART=1,"TCP","116.236.221.75", //Establish TCP connection at 1 channel in second
5555 Context
OK

1, CONNECT OK
+RECEIVE,0,15: //Receive two channel's data as normal multi
Hello world 000 connection
+RECEIVE,1,15:

```

```
Hello world 111
AT+CIPSEND=0 //Send data at two channel as normal multi connection
> test 0
0, SEND OK
AT+CIPSEND=1
>test 2
1, SEND OK
AT+CIPCLOSE=0 //Close data connection as normal multi connection.
0, CLOSE OK
AT+CIPSTATUS //Query Connection's status
OK

STATE: IP PROCESSING

C: //0 channel's connection is closed
0,0,"TCP","116.236.221.75","5555","CLOSED" //1 channel's connection is connected
C:
1,1,"TCP","116.236.221.75","5555","CONNECTED"
C: 2, "", "", "", "INITIAL"
C: 3, "", "", "", "INITIAL"
C: 4, "", "", "", "INITIAL"
C: 5, "", "", "", "INITIAL"
C: 6, "", "", "", "INITIAL"
C: 7, "", "", "", "INITIAL"
AT+CIPSHUT //Shut up second Context, but first Context is active..
SHUT OK
AT+CIPSGTXT=0 //Switch to first Context
OK
AT+CIPSHUT //Shut up first Context
SHUT OK
```

## 6 DNS Function

SIM800 series supports DNS auto parsing, so user can establish TCP/UDP connection to remote sever using domain name directly by the command "AT+CIPSTART=<mode>, <domain name>, <port>". And then user can send data to remote server by AT+CIPSEND. In multi connection, the connection number <n> is needed.

User also can use command AT+CDNSGIP=<domain name > to query the IP address of given domain name.

//Example of DNS Function

```

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

OK
AT+CSTT="CMNET" //Start up task and set APN
OK
AT+CIICR //Bring up wireless connection
OK (GPRS or CSD)
AT+CIFSR //Get local IP address
10.126.186.232
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","119.7
5.217.56"
AT+CDNSGIP="abctest" //Query "abctest"
OK
+CDNSGIP: 0,8 //Parsing error.
AT+CIPSTART="TCP","WWW.SIM.COM",8 //Establish TCP connection.
0
OK

CONNECT OK
AT+CIPSNED //Send data
> DNS test
SEND OK

```

## 7 Data Sending Related

SIM800 series provides three ways to send data: changeable data length sending, fixed data length sending and timed sending. SIM800 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.

### 7.1 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 do not need the terminal symbol CTRL+Z (0x1a) in this case. For multi connection mode, the command is “AT+CIPSEND=<n>,<LENGTH>”.

### 7.2 Timed Sending

There is another way to send data automatically. First, set the timer of sending data automatically by the command “AT+CIPATS=<mode>,<time>”, then issue “AT+CIPSEND” (“AT+CIPSEND=<n>” for multi connection) and lastly input the data after getting promoting mark “>”. Data will be sent automatically when the set timer expires.

### 7.3 Query Available Data Length to be Sent

User can use the command “AT+CIPSEND?” to query the current maximum data length available to be sent, which is determined by the network. In fact, the maximum data length is variable, depending on the actual network. The maximum data length can be up to 1460 bytes. In multi connection, the command “AT+CIPSEND?” will tell current available maximum data length for all active connections.

```
//Example of query available data length to be sent
```

```
AT+CIPSEND?
```

```
//Current maximum value 1460
```

```
+CIPSEND: 1460
```

```
OK
```

```
AT+CIPSEND=1461 //The fixed length exceeds maximum value
ERROR
AT+CIPSEND //The input data exceeds maximum value
>123....999
ERROR
```

## 7.4 Select Data Transmitting Mode

SIM800 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
> hello

DATA ACCEPT:5 //Data has been sent, not sure whether to be accepted
```

### Multi connection:

```
//Example of multi connection
AT+CIPSTART=0,"TCP","116.236.221.75", //Establish TCP connection
5107
OK

0, CONNECT OK
AT+CIPQSEND=1 //Enable quick sending mode
OK
AT+CIPSEND=0
> 1234567890

DATA ACCEPT:0,10 //Data has been sent, not sure whether to be accepted
```

## 7.5 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 first parameter <txlen> is the data amount which has been sent; the second parameter <acklen> is the data amount confirmed successfully by the server; and the third 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",51 //Establish TCP connection
07
OK

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

STATE: CONNECT OK
AT+CIPSEND
> 012345678912
DATA ACCEPT:12
AT+CIPACK //Query status of data send
+CIPACK: 12,12,0 //12 be Send , 12 be confirmed

OK
```

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

```
// Example of query data transmitting amount in multi connection
AT+CIPSTATUS //Query connection's status
OK

STATE: IP PROCESSING

C: 0,"","","","INITIAL"
C:
1,0,"TCP","116.228.221.51","8500","CONNECTED"
C:
```

```
2,0,"UDP","116.228.221.51","9600","CONN
ECTED"
C: 3,"","","","INITIAL"
C: 4,"","","","INITIAL"
C: 5,"","","","INITIAL"
AT+CIPQSEND=1 //Enable quick sending mode
OK
AT+CIPSEND=1 //Send data at 1 channel
> TCP
DATA ACCEPT:1,3
AT+CIPACK=1 //Query status about 1 channel
+CIPACK: 3, 3, 0 //3 be send, 3 be confirm

OK
AT+CIPSEND=1 //Send data at 1 channel
> TCP again
DATA ACCEPT:1,9
AT+CIPACK=1
+CIPACK: 12, 12, 0

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

OK
```

## 8 Data Sending Related

### 8.1 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 these 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.

### 8.2 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” enables getting data from network manually, which should be set before connection. If it is set to 0 (default value if not set), data will be pushed to TE directly.

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

```
//Example of receive data manually
AT+CIPRXGET=1 //Enables getting data from network manually
OK
AT+CIPSTART="TCP","116.228.221.51",55
55
OK

CONNECT OK
+CIPRXGET:1 //Data incoming from server
```

**AT+CIPRXGET=2,1460**

**+CIPRXGET:2,11,0**

**HELLO WORLD**

**OK**

**+CIPRXGET:1**

**AT+CIPRXGET=3,730**

**+CIPRXGET:3,11,0**

**48454C4C4F20574F524C44**

**OK**

//The mode is set to 2, the output data will be in normal mode, not exceeding 1460 bytes at a time.

//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.

SIMCom  
Confidential

## 9 GPRS States Exchange Related

For single connection, there are 10 GPRS states in total; for multi connection, there are 8 GPRS states. After some AT commands are executed, the corresponding state will be changed. User can get a general idea from the following diagrams:

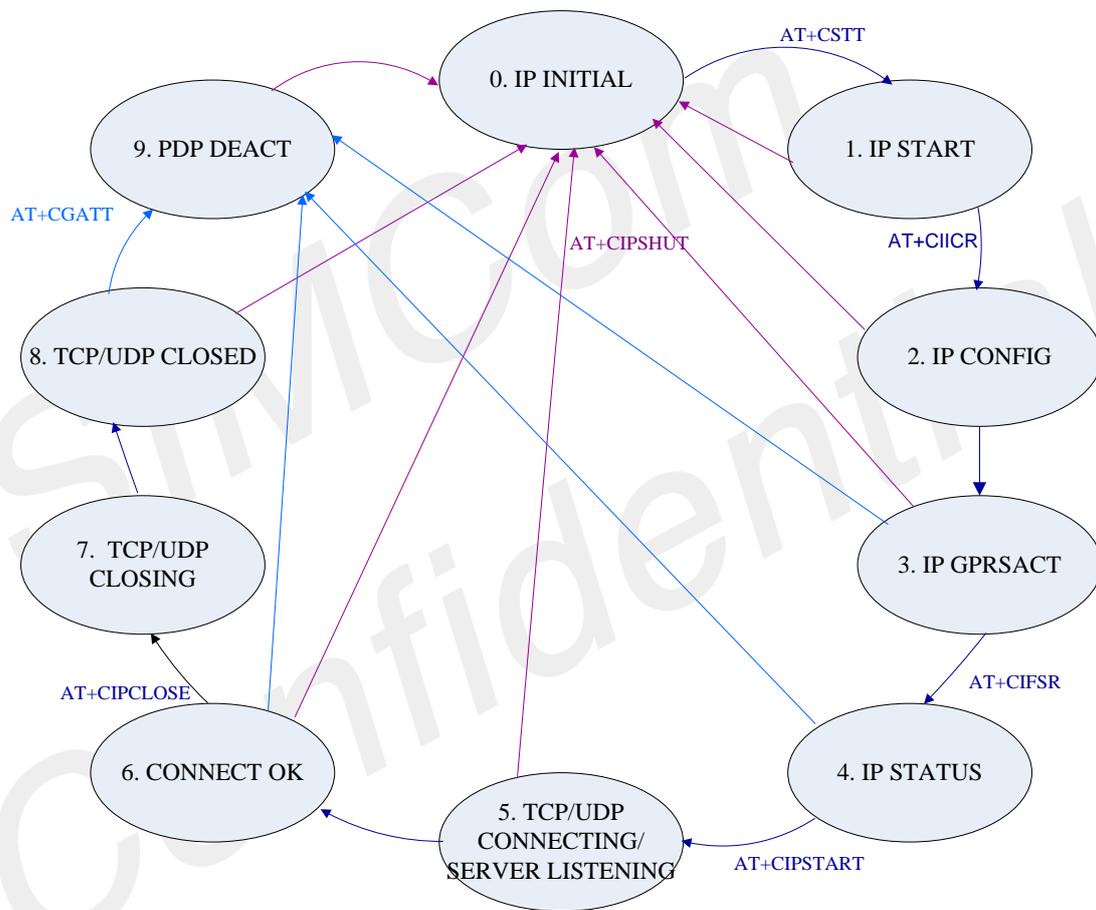
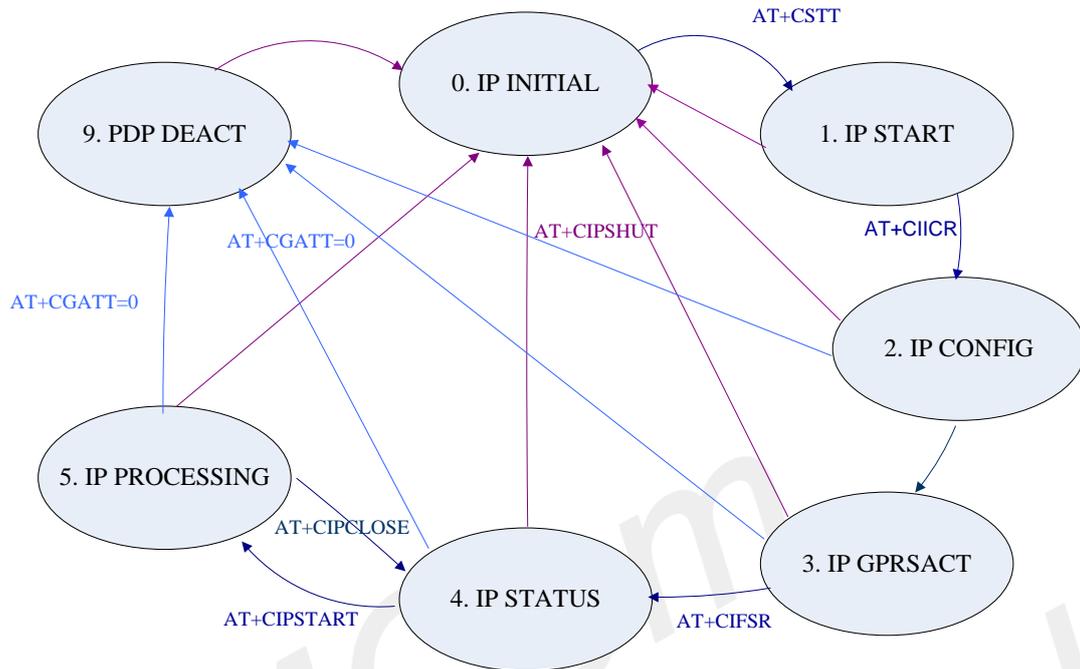


Figure2: GPRS States Diagram for single connection



**Figure3: 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

## 10 GPRS States Exchange 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.

SIMCom  
Confidential

## 11 Connection Activity Checking Related

User can use the command AT+CIPSTATUS to query current connection status.

In transparent mode, DCD pin can be used for this purpose. If TCP/UDP connection exists, the DCD pin will be active (low). At any time if the connection is dropped, DCD pin will go inactive (high).

SIMCom  
Confidential

## 12 Power Consumption with Existing Connection

The module can enter sleep mode to save power consumption with existing connection, in which the current consumption is reduced to only several mA. The sleep mode function is enabled by setting "AT+CSCLK=1". With this setting, if the DTR maintains HIGH and there is no interruption on either H/W or S/W or event over-the-air, module will go into sleep. In the sleep mode, the serial port stops working, which means no AT command can be issued any more. During the sleep mode with existing connection, the incoming data from the remote end can wake up the module. Any event over-the-air, such as incoming call or SMS etc, will wake up the module too. But At command is not be respond now, If need input AT command, must pull low DTR 50ms or more, after this, the serial port can receive data again. More detail please refer to HD document.

SIMCom  
Confidential

## 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, please use AT+CIPSHUT 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.

SIMCom  
Confidential