



# SIM7500\_SIM7600\_SIM7800 Series MQTT(S)\_Application Note

LTE 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>	SIM7500_SIM7600_SIM7800 Series_MQTT(S)_Application Note
<b>Version:</b>	2.00
<b>Date:</b>	2020.8.6
<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](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)

# About Document

## Version History

Version	Date	Owner	What is new
V2.00	2020.8.6	Ning.lv	Update the format

SIMCom  
Confidential

# Contents

<b>About Document.....</b>	<b>3</b>
Version History.....	3
<b>Contents.....</b>	<b>4</b>
<b>1. Introduction.....</b>	<b>5</b>
1.1 Purpose of the document.....	5
1.2 Related documents.....	5
1.3 Conventions and abbreviations.....	5
<b>2. MQTT Introduction.....</b>	<b>6</b>
2.1 Characteristic.....	6
2.2 Request Method.....	6
<b>3. AT Commands for MQTT(S).....</b>	<b>8</b>
<b>4. Bearer Configuration.....</b>	<b>9</b>
4.1 PDN Auto-activation.....	9
<b>5. MQTT(S) Samples.....</b>	<b>10</b>
5.1 MQTT Function.....	10
5.1.1 Access to MQTT server without SSL/TLS.....	10
5.1.2 Access to MQTT server without checking UTF8 coding.....	11
5.2 MQTTS Function.....	12
5.2.1 Connect to SSL/TLS MQTT server (not verify server).....	12
5.2.2 Access to SSL/TLS MQTT server (only verify the server).....	14
5.2.3 Access to SSL/TLS MQTT server (verify server and client).....	16

# 1.Introduction

## 1.1 Purpose of the document

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

## 1.2 Related documents

[1] SIM7600 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. MQTT Introduction

MQTT (Message Queuing Telemetry Transport) is a lightweight broker-based publish/subscribe messaging protocol. It is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport. It is useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium.

### 2.1 Characteristic

- **Support client/server mode;**
  - ✧ **The publish/subscribe message pattern to provide one-to-many message distribution and decoupling of applications**
  - ✧ **A messaging transport that is agnostic to the content of the payload**
  - ✧ **The use of TCP/IP to provide basic network connectivity**
  - ✧ **Three qualities of service for message delivery**
  - ✧ **A small transport overhead (the fixed-length header is just 2 bytes), and protocol exchanges minimised to reduce network traffic**
  - ✧ **A mechanism to notify interested parties to an abnormal disconnection of a client using the Last Will and Testament feature**

### 2.2 Request Method

According to the MQTT standard, MQTT provides a variety of request methods. CONNECT, SUBSCRIBE, PUBLISH, UNSUBSCRIBE, DISCONNECT, PINGREQ

No	Method	Description
1	CONNECT	When a TCP/IP socket connection is established from a client to a server, a protocol level session must be created using a CONNECT flow.
2	SUBSCRIBE	The SUBSCRIBE message allows a client to register an interest in one or more topic names with the server. Messages published to these topics are delivered from the server to the client as PUBLISH messages. The SUBSCRIBE message also specifies the QoS level at which the subscriber wants to receive published

		messages.
3	PUBLISH	A PUBLISH message is sent by a client to a server for distribution to interested subscribers. Each PUBLISH message is associated with a topic name (also known as the Subject or Channel). This is a hierarchical name space that defines taxonomy of information sources for which subscribers can register an interest. A message that is published to a specific topic name is delivered to connected subscribers for that topic.
4	UNSUBSCRIBE	An UNSUBSCRIBE message is sent by the client to the server to unsubscribe from named topics.
5	DISCONNECT	The DISCONNECT message is sent from the client to the server to indicate that it is about to close its TCP/IP connection. This allows for a clean disconnection, rather than just dropping the line.
6	PINGREQ	The PINGREQ message is an "are you alive?" message that is sent from a connected client to the server.

The SIM7600 series supports several methods: CONNECT, SUBSCRIBE, PUBLISH, UNSUBSCRIBE, DISCONNECT.

SIMCom  
Confidential

## 3. AT Commands for MQTT(S)

Command	Description
<b>AT+CMQTTSTART</b>	Start MQTT service
<b>AT+CMQTTSTOP</b>	Stop MQTT service
<b>AT+CMQTTACCQ</b>	Acquire a MQTT client
<b>AT+CMQTTREL</b>	Release a MQTT client
<b>AT+CMQTTSSLCFG</b>	Set the SSL context
<b>AT+CMQTTWILLTOPIC</b>	Input the topic of will message
<b>AT+CMQTTWILLMSG</b>	Input the will message
<b>AT+CMQTTCONNECT</b>	Connect to a MQTT server
<b>AT+CMQTTDISC</b>	Disconnect from server
<b>AT+CMQTTTOPIC</b>	Input the publish message topic
<b>AT+CMQTTPAYLOAD</b>	Input the publish message body
<b>AT+CMQTTTPUB</b>	Publish a message to server
<b>AT+CMQTTSUBTOPIC</b>	Input a subscribe message topic
<b>AT+CMQTTSUB</b>	Subscribe a message to server
<b>AT+CMQTTUNSUBTOPIC</b>	Input a unsubscribe message topic
<b>AT+CMQTTUNSUB</b>	Unsubscribe a message to server
<b>AT+CMQTTCFG</b>	Configure the MQTT Context

For detail information, please refer to "SIM7600 Series\_AT Command Manual ".



## 4. Bearer Configuration

Usually module will register PS service automatically.

### 4.1 PDN Auto-activation

//Example of PDN Auto-activation.

**AT+CPIN?**

+CPIN: READY

Check SIM card status

OK

**AT+CSQ**

+CSQ: 27,99

Check RF signal

OK

**AT+CGREG?**

+CGREG: 0,1

Check PS service

OK

**AT+COPS?**

+COPS: 0,0," CHINA MOBILE",7

Query Network information, operator and network mode 7, LTE network

OK

## 5. MQTT(S) Samples

### 5.1 MQTT Function

#### 5.1.1 Access to MQTT server without SSL/TLS

//Example of Access to MQTT server without SSL/TLS.

```
AT+CMQTTSTART           Start MQTT service, activate PDP context
OK

+CMQTTSTART: 0
AT+CMQTTACCQ=0, "client test0"  Acquire one client which will connect to a MQTT
OK                               server not SSL/TLS
AT+CMQTTWILLTOPIC=0,10         Set the will topic for the CONNECT message
>0123456789
OK
AT+CMQTTWILLMSG=0,6,1         Set the will message for the CONNECT message
>qwerty
OK
AT+CMQTTCONNECT=0,"tcp://test.mosquitto.  Connect to a MQTT server
org:1883",60,1
OK

+CMQTTCONNECT: 0,0
AT+CMQTTSUB=0,10,1           Subscribe one topic from the server
>simcomtest
OK

+CMQTTSUB: 0,0
AT+CMQTTTOPIC=0,10         Set the topic for the PUBLISH message
> simcomtest

OK
AT+CMQTTPAYLOAD=0,9         Set the payload for the PUBLISH message
>mqtt_test
OK
AT+CMQTTTPUB=0,1,60         Publish a message
```

OK

+CMQTTPUB: 0,0

+CMQTTRXSTART: 0,10,9

Receive publish message from server

+CMQTTRXTOPIC: 0,10

simcomtest

+CMQTTRXPAYLOAD: 0,9

mqtt\_test

+CMQTTRXEND: 0

**AT+CMQTTSUBTOPIC=0,9,1**

Set one topic for the SUBSCRIBE message

>123456789

OK

**AT+CMQTTSUB=0**

Subscribe a message

OK

+CMQTTSUB: 0,0

**AT+CMQTTUNSUB=0,9,0**

Unsubscribe one topic from the server

>simcommmsg

OK

+CMQTTUNSUB: 0,0

**AT+CMQTTDISC=0,120**

Disconnect from server

OK

+CMQTTDISC: 0,0

**AT+CMQTTREL=0**

Release the client

OK

**AT+CMQTTSTOP**

Stop MQTT Service

OK

+CMQTTSTOP: 0

### 5.1.2 Access to MQTT server without checking UTF8 coding

//Example of Access to MQTT server without checking UTF8 coding.

**AT+CMQTTSTART**

Start MQTT service, activate PDP context

OK

+CMQTTSTART: 0

**AT+CMQTTACCQ=0, "client test0"**

Acquire one client which will connect to a MQTT server not SSL/TLS

OK

**AT+CMQTTCFG="checkUTF8",0,0**

Configure not checking UTF8 coding

OK

<b>AT+CMQTTCONNECT=0,"tcp://test.mosquitto.org:1883",60,1</b>	Connect to a MQTT server
OK	
<b>+CMQTTCONNECT: 0,0</b>	
<b>AT+CMQTTSUB=0,9,1</b>	Subscribe one topic which is not UTF8 coding string, The data can input by hexadecimal format
> 00000000?	
OK	
<b>+CMQTTSUB: 0,0</b>	
<b>AT+CMQTTTOPIC=0,9</b>	Set the topic for the PUBLISH message
> 00000000?	
OK	
<b>AT+CMQTTTPUB=0,1,60</b>	Publish a message
OK	
<b>+CMQTTTPUB: 0,0</b>	
<b>+CMQTTTRXSTART: 0,9,0</b>	Receive publish message from server
<b>+CMQTTTRXTOPIC: 0,9</b>	
00000000?	
<b>+CMQTTTRXEND: 0</b>	
<b>AT+CMQTTDISC=0,120</b>	Disconnect from server
OK	
<b>+CMQTTDISC: 0,0</b>	
<b>AT+CMQTTREL=0</b>	Release the client
OK	
<b>AT+CMQTTSTOP</b>	Stop MQTT Service
OK	
<b>+CMQTTSTOP: 0</b>	

## 5.2 MQTTS Function

### 5.2.1 Connect to SSL/TLS MQTT server (not verify server)

// Example of Access to a MQTT server without verifying the server

<b>AT+CMQTTSTART</b>	Start MQTT service, activate PDP context
----------------------	--



OK

+CMQTTUNSUB: 0,0

AT+CMQTTDISC=0,120

Disconnect from server

OK

+CMQTTDISC: 0,0

AT+CMQTTREL=0

Release the client

OK

AT+CMQTTSTOP

Stop MQTT Service

OK

+CMQTTSTOP: 0

## 5.2.2 Access to SSL/TLS MQTT server (only verify the server)

//Example of Access to SSL/TLS MQTT server (only verify the server)

AT+CSSLCFG="sslversion",0,4

Set the SSL version of the first SSL context

OK

AT+CSSLCFG="authmode",0,1

Set the authentication mode(verify server) of the first SSL context

OK

AT+CSSLCFG="cacert",0,"server\_ca.pem"

Set the server root CA of the first SSL context

OK

AT+CMQTTSTART

Start MQTT service, activate PDP context

OK

+CMQTTSTART: 0

AT+CMQTTACCQ=0,"client test0",1

Acquire one client which will connect to a SSL/TLS MQTT server

OK

AT+CMQTTSSLCFG=0,0

Set the first SSL context to be used in the SSL connection

OK

AT+CMQTTWILLTOPIC=0,10

Set the will topic for the CONNECT message

>0123456789

OK

AT+CMQTTWILLMSG=0,6,1

Set the will message for the CONNECT message

>qwerty

OK

AT+CMQTTCONNECT=0,"tcp://mqttps\_server:port",60,1

Connect to a MQTT server, input the right server and port

OK

<b>+CMQTTCONNECT: 0,0</b> <b>AT+CMQTTTOPIC=0,13</b> >ddrrrrggghhk	Set the topic for the PUBLISH message
OK	
<b>AT+CMQTTPAYLOAD=0,60</b> >01234567890123456789012345678901234567 8901234567890123456789	Set the payload for the PUBLISH message
OK	
<b>AT+CMQTTTTPUB=0,1,60</b> OK	Publish a message
<b>+CMQTTTTPUB: 0,0</b> <b>AT+CMQTTSUBTOPIC=0,9,1</b> >123456789	Set one topic for the SUBSCRIBE message
OK	
<b>AT+CMQTTTTSUB=0</b> OK	Subscribe a message
<b>+CMQTTTTSUB: 0,0</b> <b>AT+CMQTTTTSUB=0,9,1</b> >simcommsg OK	Subscribe one topic from the server
<b>+CMQTTTTSUB: 0,0</b> <b>AT+CMQTTTUNSUB=0,9,0</b> >simcommsg OK	Unsubscribe one topic from the server
<b>+CMQTTTUNSUB: 0,0</b> <b>AT+CMQTTDISC=0,120</b> OK	Disconnect from server
<b>+CMQTTDISC: 0,0</b> <b>AT+CMQTTREL=0</b> OK	Release the client
<b>AT+CMQTTSTOP</b> OK	Stop MQTT Service
<b>+CMQTTSTOP: 0</b>	

### 5.2.3 Access to SSL/TLS MQTT server (verify server and client)

```
//Access to SSL/TLS MQTT server (verify server and client).
AT+CSSLCFG="sslversion",0,4           Set the SSL version of the first SSL context
OK
AT+CSSLCFG="authmode",0,2           Set the authentication mode(verify server and
OK                                     client) of the first SSL context
AT+CSSLCFG="cacert",0,"ca_cert.pem"  Set the server root CA of the first SSL context
OK
AT+CSSLCFG="clientcert",0,"cert.pem"  Set the client certificate of the first SSL context
OK
AT+CSSLCFG="clientkey",0,"key_cert.pem" Set the client key of the first SSL context
OK
AT+CMQTTSTART                         start MQTT service, activate PDP context
OK

+CMQTTSTART: 0
AT+CMQTTACCQ=0,"client test0",1      Acquire one client which will connect to a SSL/TLS
OK                                     MQTT server
AT+CMQTTSSLCFG=0,0                   Set the first SSL context to be used in the SSL
OK                                     connection
AT+CMQTTWILLTOPIC=0,10               Set the will topic for the CONNECT message
>0123456789

OK
AT+CMQTTWILLMSG=0,6,1                Set the will message for the CONNECT message
>qwerty

OK
AT+CMQTTCONNECT=0,"tcp://hooleeping.co  Connect to a MQTT server
m:8883",60,1
OK

+CMQTTCONNECT: 0,0
AT+CMQTTTOPIC=0,13                   Set the topic for the PUBLISH message
>ddrrrrggghhk

OK
AT+CMQTTPAYLOAD=0,60                 Set the payload for the PUBLISH message
>01234567890123456789012345678901234567
8901234567890123456789

OK
AT+CMQTT PUB=0,1,60                  Publish a message
OK
```



<b>+CMQTTPUB: 0,0</b> <b>AT+CMQTTSUBTOPIC=0,9,1</b> >123456789	Set one topic for the SUBSCRIBE message
OK <b>AT+CMQTTSUB=0</b> OK	Subscribe a message
<b>+CMQTTSUB: 0,0</b> <b>AT+CMQTTSUB=0,9,1</b> >simcommmsg OK	Subscribe one topic from the server
<b>+CMQTTSUB: 0,0</b> <b>AT+CMQTTUNSUB=0,9,0</b> >simcommmsg OK	Unsubscribe one topic from the server
<b>+CMQTTUNSUB: 0,0</b> <b>AT+CMQTTDISC=0,120</b> OK	Disconnect from server
<b>+CMQTTDISC: 0,0</b> <b>AT+CMQTTREL=0</b> OK	Release the client
<b>AT+CMQTTSTOP</b> OK	Stop MQTT Service
<b>+CMQTTSTOP: 0</b>	