SIM7022 Series_MQTT(S) _Application Note

LPWA Module

Document Title:	SIM7022 Series_MQTT(S)_Application Note
Version:	1.02
Date:	2022.07.08
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 © 2022 SIMCom Wireless Solutions Limited All Rights Reserved.

About Document

Version History

Revision	Date	Chapter	Description
V1.00	2022.5.12	All	New version
V1.01	2022.05.31	All	Update file
V1.02	2022.07.08	All	Update some description

Scope

This document could be applied to following modules.

Name	Type	Size(mm)	Comments
SIM7022	NB2	17.6*15.7	Band 1/2/3/4/5/8/12/13/14/17/18/19/20/25/26/28/66/70/85

Contents

1   Introduction

1.1   Purpose of the document

Based on module AT command manual, this document will introduce MQTTS application process on SIM7022 series of module,developers could understand and develop application quickly and efficiently based on this document.

1.2   Related documents

[1] SIM7022 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;

Other Conventions:
- MQTT(Message Queuing Telemetry Transport);
- SSL(Secure Sockets Layer);
- PDP(Packet Data Protocol);
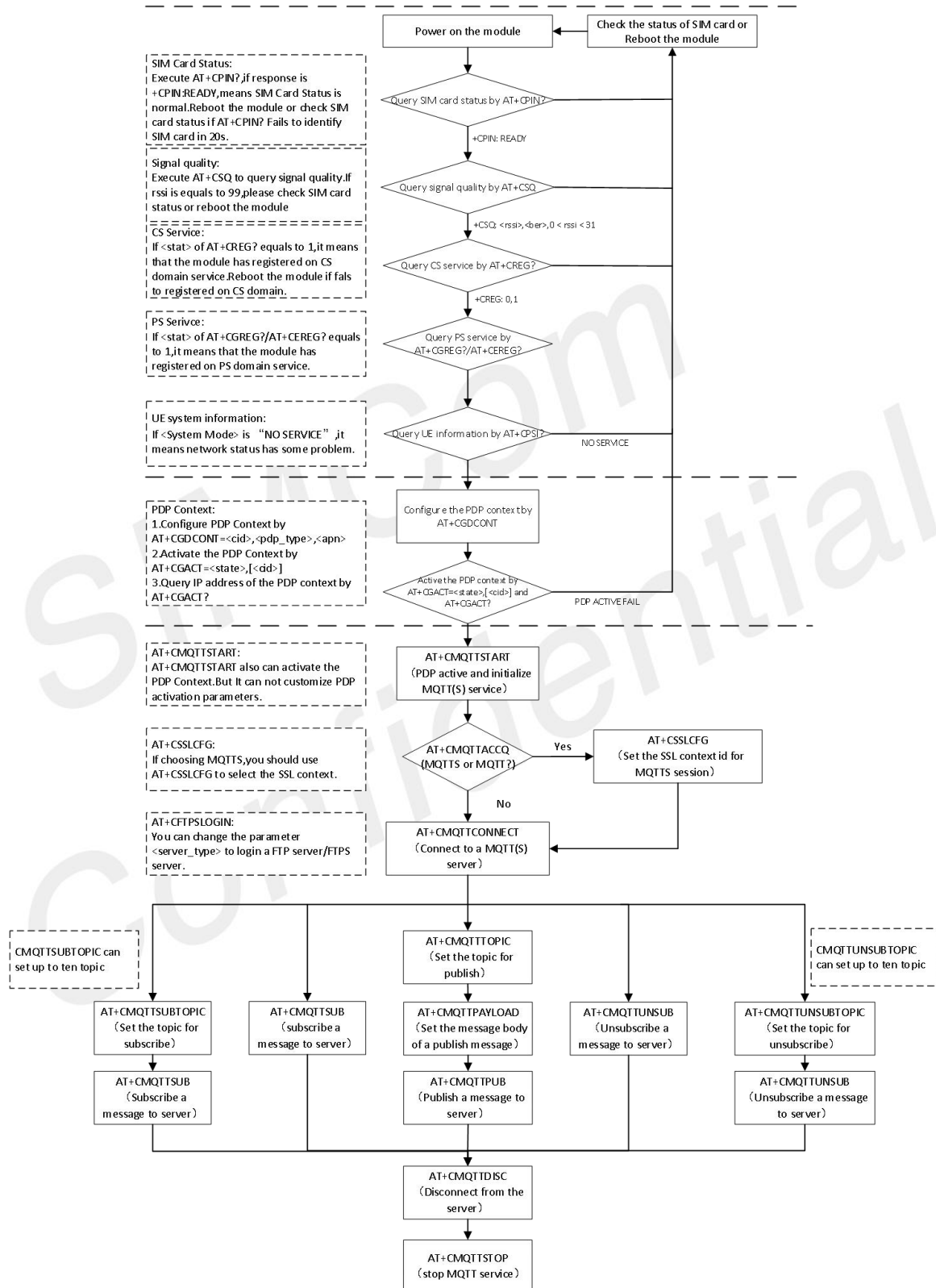
2   MQTT(S) Introduction

2.1   MQTT(S) Introduction

MQTT (Message Queue Telemetry Transport) is a messaging protocol based on the publish/subscribe paradigm under the ISO standard (ISO/IEC PRF 20922). It works on the TCP/IP protocol suite and is a publish/subscribe messaging protocol designed for remote devices with poor hardware performance and poor network conditions.

The MQTT protocol is a protocol designed for the communication of remote sensors and control devices with limited computing power and working on low-bandwidth, unreliable networks. It has the following main features:

➢   Use the publish/subscribe message mode to provide one-to-many message publishing and uncouple the application;

➢   Message transmission for shielding the payload content;

➢   Provide network connection using TCP/IP;

➢   There are three types of message publishing service quality:

✧   "At most once," message publishing relies entirely on the underlying TCP/IP network. Message loss or duplication can occur. This level can be used in the following situations, environmental sensor data, loss of a read record does not matter, because there will be a second transmission in the near future.

✧   "At least once" to ensure that the message arrives, but message duplication may occur.

✧   "Only once" to ensure that the message arrives once. This level can be used in situations where repeated or missing messages can result in incorrect results.

➢   small transmission, low overhead (fixed length of the head is 2 bytes), protocol exchange is minimized to reduce network traffic;

➢   Use the Last Will and Testament features to notify the parties about the mechanism of client abort.

2.2 The process of Using MQTT(S) AT Command

3   AT Commands for MQTT(S)

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

For detail information, please refer to "SIM7022 Series_AT Command Manual".

4 MQTT(S) Examples

Before all MQTT(S) related operations, we should ensure the following:
Ensure network is available:

AT+CSQ	
+CSQ: 23,0	
OK	
AT+CGREG?	Need to check network registeration state until get
+CGREG: 0,1	1(home register) or 5(roaming register)
OK	
AT+CGDCONT=1,"IP","apn"	Customer need to set IP type(IP or IPV6) and
OK	correct apn name

4.1 Connect to MQTT broker without SSL/TLS

Following commands shows how to communicate with an MQTT broker.

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 without SSL/TLS
AT+CMQTTWILLTOPIC=0,10	//Set the will topic for the CONNECT message
>	
OK	
AT+CMQTTWILLMSG=0,6,1	//Set the will message for the CONNECT message
>	
OK	
AT+CMQTTCONNECT=0,"tcp://test.mosquitto.	//Connect to an MQTT broker

org:1883",60,1
OK

+CMQTTCONNECT: 0,0
AT+CMQTTSUB=0,9,1   //Subscribe one topic from the broker
>

OK

+CMQTTSUB: 0,0
AT+CMQTTTOPIC=0,9   //Set the topic for the PUBLISH message
>

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

OK
AT+CMQTTPUB=0,1,60   //Publish a message
OK

+CMQTTPUB: 0,0
+CMQTTRXSTART: 0,9,60   //Receive publish message from broker
+CMQTTRXTOPIC: 0,9
simcommsg
+CMQTTRXPAYLOAD: 0,60
012345678901234567890123456789012345678
901234567890123456789
+CMQTTRXEND: 0
AT+CMQTTSUBTOPIC=0,9,1   //Set one topic for the SUBSCRIBE message
>

OK
AT+CMQTTSUB=0   //Subscribe a message
OK

+CMQTTSUB: 0,0
AT+CMQTTUNSUB=0,9,0   //Unsubscribe one topic from the broker
>

OK

+CMQTTUNSUB: 0,0
AT+CMQTTDISC=0,120   //Disconnect from broker
OK

```
+CMQTTDISC: 0,0
AT+CMQTTREL=0                              //Release the client
OK
AT+CMQTTSTOP                               //Stop MQTT Service
OK

+CMQTTSTOP: 0
```

4.2 Connect to SSL/TLS MQTT broker(not verify server)

Following commands shows how to access to an MQTT broker without verifying the server. It needs to configure the authentication mode to 0, and then it will connect to the server successfully.

```
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
OK                                        SSL/TLS MQTT broker
AT+CMQTTWILLTOPIC=0,10                    //Set the will topic for the CONNECT message
>

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

OK
AT+CMQTTCONNECT=0,"tcp://test.mosquitto.  //Connect to a MQTT broker
org:8883",60,1
OK

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

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

OK
AT+CMQTTPUB=0,1,60                        //Publish a message
OK
```

```
+CMQTTPUB: 0,0
AT+CMQTTSUBTOPIC=0,9,1              //Set one topic for the SUBSCRIBE message
>

OK
AT+CMQTTSUB=0                        //Subscribe a message
OK

+CMQTTSUB: 0,0
AT+CMQTTSUB=0,9,1                    //Subscribe one topic from the broker
>

OK

+CMQTTSUB: 0,0
AT+CMQTTUNSUB=0,9,0                  //Unsubscribe one topic from the broker
>

OK

+CMQTTUNSUB: 0,0
AT+CMQTTDISC=0,120                   //Disconnect from broker
OK

+CMQTTDISC: 0,0
AT+CMQTTREL=0                        //Release the client
OK
AT+CMQTTSTOP                         //Stop MQTT Service
OK

+CMQTTSTOP: 0
```

4.3 Access to SSL/TLS MQTT broker(only verify the server)

Following commands shows how to access to a SSL/TLS MQTT broker with verifying the server. It needs to configure the authentication mode to 1 and the right server root CA, and then it will connect to the server successfully.

```
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
>	
OK	
AT+CMQTTWILLMSG=0,6,1	//Set the will message for the CONNECT message
>	
OK	
AT+CMQTTCONNECT=0,"tcp://mqtts_server:port",60,1	//Connect to a MQTT broker, input the right broker and port
OK	
+CMQTTCONNECT: 0,0	
AT+CMQTTTOPIC=0,13	//Set the topic for the PUBLISH message
>	
OK	
AT+CMQTTPAYLOAD=0,60	//Set the payload for the PUBLISH message
>	
OK	
AT+CMQTTPUB=0,1,60	//Publish a message
OK	
+CMQTTPUB: 0,0	
AT+CMQTTSUBTOPIC=0,9,1	//Set one topic for the SUBSCRIBE message
>	
OK	
AT+CMQTTSUB=0	//Subscribe a message
OK	
+CMQTTSUB: 0,0	
AT+CMQTTSUB=0,9,1	//Subscribe one topic from the broker

```
>

OK

+CMQTTSUB: 0,0
AT+CMQTTUNSUB=0,9,0                     //Unsubscribe one topic from the broker
>

OK

+CMQTTUNSUB: 0,0
AT+CMQTTDISC=0,120                      //Disconnect from broker
OK

+CMQTTDISC: 0,0
AT+CMQTTREL=0                           //Release the client
OK
AT+CMQTTSTOP                            //Stop MQTT Service
OK

+CMQTTSTOP: 0
```

4.4 Access to SSL/TLS MQTT broker(verify server and client)

Following commands shows how to access to a SSL/TLS MQTT broker with verifying the server and client. It needs to configure the authentication mode to 2, the right server root CA, the right client certificate and key, and then it will connect to the server successfully.

```
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 MQTT broker
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
>	
OK	
AT+CMQTTWILLMSG=0,6,1	//Set the will message for the CONNECT message
>	
OK	
AT+CMQTTCONNECT=0,"tcp://hooleeping.com:8883",60,1	//Connect to a MQTT broker
OK	
+CMQTTCONNECT: 0,0	
AT+CMQTTTOPIC=0,13	//Set the topic for the PUBLISH message
>	
OK	
AT+CMQTTPAYLOAD=0,60	//Set the payload for the PUBLISH message
>	
OK	
AT+CMQTTPUB=0,1,60	//Publish a message
OK	
+CMQTTPUB: 0,0	
AT+CMQTTSUBTOPIC=0,9,1	//Set one topic for the SUBSCRIBE message
>	
OK	
AT+CMQTTSUB=0	//Subscribe a message
OK	
+CMQTTSUB: 0,0	
AT+CMQTTSUB=0,9,1	//Subscribe one topic from the broker
>	
OK	
+CMQTTSUB: 0,0	
AT+CMQTTUNSUB=0,9,0	//Unsubscribe one topic from the broker
>	

OK

+CMQTTUNSUB: 0,0
AT+CMQTTDISC=0,120   //Disconnect from broker
OK

+CMQTTDISC: 0,0
AT+CMQTTREL=0   //Release the client
OK
AT+CMQTTSTOP   //Stop MQTT Service
OK

+CMQTTSTOP: 0

4.5   Access to MQTT broker without checking UTF8 coding

Following commands shows how to communicate with a MQTT broker 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
OK   server not SSL/TLS
AT+CMQTTCFG="checkUTF8",0,0   //Configure not checking UTF8 coding
OK
AT+CMQTTCONNECT=0,"tcp://198.41.30.241:1   //Connect to a MQTT broker
883",60,1
OK

+CMQTTCONNECT: 0,0
AT+CMQTTSUB=0,9,1   //Subscribe one topic which is not UTF8 coding
>   string.
                                                 //The data can input by hexadecimal format.
OK

+CMQTTSUB: 0,0
AT+CMQTTTOPIC=0,9   //Set the topic for the PUBLISH message
>

OK
AT+CMQTTPUB=0,1,60   //Publish a message

OK

+CMQTTPUB: 0,0
+CMQTTRXSTART: 0,9,0   //Receive publish message from broker
+CMQTTRXTOPIC: 0,9
xxxxxx

+CMQTTRXEND: 0
AT+CMQTTDISC=0,120   //Disconnect from broker
OK

+CMQTTDISC: 0,0
AT+CMQTTREL=0   //Release the client
OK
AT+CMQTTSTOP   //Stop MQTT Service
OK

+CMQTTSTOP: 0