SIM800 Series_MQTT _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
www.simcom.com

Document Title:	SIM800 Series _MQTT_Application Note
Version:	1.03
Date:	2020.10.16
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

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

About Document

Version History

Version	Date	Owner	What is new
1.00	2017-10-11	Dingfen.Zhu	Original
1.01	2019-12-10	Xiaohui.Xu	Chapter 2.2, Add AT+SAZURECONF Chapter 3.2, Add MQTT Connecting to Azure IoT
1.02	2020-06-15	Yizhe.Tan /Wenjie.Lai	Modify format and style
1.03	2020-10-16	Wenjie.Lai	Chapter 4.1,Modidy examples and format

Scope

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

Contents

1 Introduction

1.1 Purpose of the document

Based on module AT command manual, this document will introduce MQTT 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

Abbreviations	Description
GPRS	General Packet Radio Service
PDP	Packet Data Protocol
MQTT	Message Queuing Telemetry Transport

2 MQTT Introduction

MQTT (Message Queue Telemetry Transport) is a messaging protocol based on the publish/subscribe paradigm under the ISO standard (ISO/IEC PRF 20922).

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:

1) Use the publish/subscribe message mode to provide one-to-many message publishing and uncouple application.

2) Message transmission blocked by the payload content.

3) Provide network connectivity using TCP/IP.

4) There are three types of message publishing service quality:

- "At most once", the message is completely dependent 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.

5) Small transmission, low overhead (fixed length header is 2 bytes), protocol exchange is minimized to reduce network traffic.

6) Use Last Will and Testament features to notify the mechanism of abnormal client interruption.

3 AT Commands for MQTT

SIM800 series modules provide MQTT AT command is as follows:

AT Command	Description
AT+SMCONF	Set MQTT Parameter
AT+SAZURECONF	Set Azure IoT Parameter
AT+SMCONN	MQTT Connection
AT+SMSUB	Subscribe Packet
AT+SMUNSUB	Unsubscribe Packet
AT+SMPUB	Publish Packet
AT+SMSTATE	Inquire MQTT Connection Status
AT+SMDISC	Disconnect MQTT
AT+SMSSL	Set MQTT to Use SSL Function
+SMPUBLISH	Received Data

3.1 AT+SMCONF Set MQTT Parameter

AT+SMCONF Set MQTT Parameter	
Test Command AT+SMCONF=?	Response **+SMCONF: "MQTTParamTag","MQTTParamValue"** **OK**
Read Command AT+SMCONF?	Response **+SMCONF: <MQTTParamTag>,<MQTTParamValue>** **OK**.
Write Command AT+SMCONF=<MQTTParamTag>,<MQTTParamValue>	Response **OK** or **+CME ERROR: <err>** Parameters **<MQTTParamTag>** MQTT Parameter **"CID"** Bearer profile identifier **"URL"** MQTT Server URL

	"serve:tcpPort" "server": FQDN or IP-address "tcpPort": default value is 1883 **"CLIENTID"** Client connection id. Default is NULL. **"KEEPALIVE"** Hold connect time.default:60,Range: (60-3600) **"CLEANSS"** Clean session flag，default:0，Range:(0-1) **"USERNAME"** User name. default null **"PASSWORD"** Password，default null **"TIMEOUT"** MQTT response timeout value **<MQTTarmValue>** MQTT Parameter value
Reference	Note

3.2 AT+SAZURECONF Set Azure IoT Parameter

AT+SAZURECONF Set Azure IoT Parameter	
Test Command **AT+SAZURECONF=?**	Response **+SAZURECONF: "url","deviceId","deviceKey",<expiry_time>** **OK**
Write Command **AT+SAZURECONF=<URL>,<DeviceId>,<DeviceKey>,<Expiry_time>**	Response **OK** or **+CME ERROR: <err>** Parameters **< URL>** Azure IoT HostName "serve:tcpPort" "server": HostName "tcpPort": Port need to configure to 8883. **<DeviceId>** The Id of the Azure IoT Device. **<DeviceKey>** The "PrimaryKey" of the Azure IoT Device. **<expiry_time>** Expiration time,Rang:0~3600*24*365(second).
Reference	Note

3.3 AT+SMCONN MQTT Connection

AT+SMCONN MQTT Connection

Test Command **AT+SMCONN=?**	Response **OK**
Execution Command **AT+SMCONN**	Response **OK** or **+CME ERROR: <err>**
Reference	Note

3.4 AT+SMSUB Subscribe Packet

AT+SMSUB Subscribe Packet	
Test Command **AT+SMSUB=?**	Response **+SMSUB: "topic",<qos>** **OK**
Write Command **AT+SMSUB=<topic>,<qos>**	Response **OK** or **+CME ERROR: <err>** Unsolicited Result Code **+SMSUB: <packet_id>,<status>** Parameters **<topic>** Topic name **<qos>** Qos level, range:(0-1) **<packet_id>** Packet id **<status>** Subscribe status 0 Success 1 Time Out 2 Other Error
Reference	Note

3.5 AT+SMPUB Publish Packet

AT+SMPUB Publish Packet	
Test Command **AT+SMPUB=?**	Response **+SMPUB: "topic",<qos>,"message"**

	OK
Write Command **AT+SMPUB=<topic>,<qos>,<retain>,<message>**	Response **OK** or **+CME ERROR: <err>** Unsolicited Result Code **+SMPUB: <packet_id>,<status>** Parameters **<topic>** Topic name **<qos>** Qos level，rang: (0-1) **<retain>** Retain flag，default 0, range:（0-1） **<message>** Message content,, range: (0-1024) **<packet_id>** Packet id **<status>** Publish status 0 Success 1 Time Out 2 Other Error
Reference	Note

3.6 AT+SMUNSUB Unsubscribe Packet

AT+SMUNSUB Unsubscribe Packet	
Test Command **AT+SMUNSUB=?**	Response **+SMUNSUB: "topic"** **OK**
Write Command **AT+SMUNSUB=<topic>**	Response **OK** or **+CME ERROR: <err>** Unsolicited Result Code **+SMUNSUB: <packet_id>,<status>** Parameters **<topic>** Topic name **<packet_id>** Packet id **<status>** Unsubscribe status 0 Success

	1 Time Out 2 Other Error
Reference	Note

3.7 AT+SMSTATE Inquire MQTT Connection Status

AT+SMSTATE Inquire MQTT Connection Status	
Test Command **AT+SMSTATE=?**	Response **OK**
Read Command **AT+SMSTATE?**	Response **+SMSTATE: <status>** **OK** Parameters **<status>** 0 Disconnect status 1 Connect status
Reference	Note

3.8 AT+SMDISC Disconnect MQTT

AT+SMDISC Disconnect MQTT	
Test Command **AT+SMDISC=?**	Response **OK**
Read Command **AT+SMDISC?**	Response **OK** or **+CME ERROR: <err>**
Reference	Note

3.9 AT+SMSSL Set MQTT to Use SSL Function

AT+SMSSL Set MQTT to Use SSL Function	
Test Command **AT+SMSSL=?**	Response **+SMSSL: <Enable>** **OK**
Read Command **AT+SMSSL?**	Response **+SMSSL: <Enable>** **OK**
Write Command **AT+SMSSL=<Enable>**	Response **OK** or **+CME ERROR: <err>** Parameters **<Enable>** 0 Disable SSL function 1 Enable SSL function
Reference	Note

3.10 +SMPUBLISH Received Data

+SMPUBLISH Received Data	
	Unsolicited Result Code **+SMPUBLISH: <packet_id>,"topic",<msgLen>,"message"**
	Parameters **<packet_id>** Packet id **<topic>** Topic name **<messageLen>** Message length **<message>** message content
Reference	Note

4 MQTT Examples

4.1 Standard MQTT

The following table provides some using method of the MQTT function.

```
//Example of Standard MQTT.
AT+SAPBR=3,1,"Contype","GPRS"          //Configure bearer profile 1
OK
AT+SAPBR=3,1,"APN","CMNET"
OK
AT+SAPBR=1,1                           //To open a GPRS context
OK
AT+SAPBR=2,1                           //To query the GPRS context
+SAPBR: 1,1,"10.89.193.1"

OK
AT+SMCONF="URL","117.131.85.139:6000"  //Set parameters for MQTT
OK
AT+SMCONF="CLEANSS",1
OK
AT+SMCONN                              //MQTT Connection
OK
AT+SMSUB="Topic1",1                    //Subscribe topic
OK                                     //Subscribe response

+SMSUB: 1,0

AT+SMPUB="Topic1",1,0,"hello world"    //Publish message
OK                                     //Publish response

+SMPUB: 2,0

+SMPUBLISH: 1,"Topic1",11,"hello world"  //Received published data
AT+SMUNSUB="Topic1"                    //Unsubscribe topic
OK                                     //Unsubscribe response
```

```
+SMUNSUB: 3,0
AT+SMDISC                                   //Disconnect MQTT
OK
AT+SAPBR=0,1                                //To close a GPRS context.
OK
```

4.2 MQTT Connecting to Azure IoT

The following table provides the example of MQTT Connecting to Azure IoT.

```
//Example of MQTT Connecting to Azure IoT.
AT+CLTS=1                                   //Synchronize local time when registering base station
OK
AT&W
OK
AT+CFUN=0
OK
AT+CFUN=1
+CPIN: READY

OK

AT+CCLK?                                    //Get local Time
+CCLK: "19/09/04,16:05:01+32"

OK
AT+SAPBR=3,1,"Contype","GPRS"               //Configure bearer profile 1
OK
AT+SAPBR=3,1,"APN","CMNET"                  //Enable wireless connection, this parameter needs to set different APN values according to different cards.
OK
AT+SAPBR=1,1                                //To open a GPRS context
OK
AT+SAPBR=2,1                                //To query the GPRS context
+SAPBR: 1,1,"10.156.5.253"

OK
```

AT+FSCREATE=C:\USER\HENRY_SSL.CRT	//Import the Azure Root Certificate File
OK	
AT+FSWRITE=C:\USER\HENRY_SSL.CRT,0,947,10>	
OK	
AT+SSLSETROOT="C:\USER\HENRY_SSL.CRT",947	
OK	
AT+SSLOPT=0,0	
OK	
AT+SSLOPT=1,1	
OK	
AT+SMSSL=1	//Set MQTT to Use SSL Function
OK	
AT+SAZURECONF="9AMIoTHub-HW.azure-devices.cn:8883","simcomdevice","n1AoqKmG6ItXWtNX1HL4zPAih/ug50D7P4rCv6pc/3c=",86400	//Set parameters for Azure IoT <DeviceID> is “simcomdevice”.
OK	
AT+SMCONN	//MQTT Connection
OK	
AT+SMSUB="devices/simcomdevice/messages/devicebound/#",1	//Subscribe topic（The format of the subscribe topic is as below, "devices/deviceID/messages/devicebound/#"）. Just replace "deviceID" with <DeviceID> in AT+SAZURECONF command.
OK	//Subscribe response
+SMSUB: 1,0	
+SMPUBLISH: 3,"devices/simcomdevice/messages/devicebound/%24.mid=17a3c1ba-9aed-437a-9923-4426ba711167&%24.to=%2Fdevices%2Fsimcomdevice%2Fmessag",6,"123213"	//Receive the message from the Azure IoT
AT+SMPUB="devices/simcomdevice/messages/events/",1,0,"hello world"	//Publish message（The format of the publish topic is as below,"devices/deviceID/messages/events/" ） . Just replace "deviceID" with <DeviceID> in AT+SAZURECONF command.
OK	//Publish response
+SMPUB: 2,0	
	//If successful, you can receive the published message “hello world” from Azure IoT.
AT+SMDISC	//Disconnect MQTT

```
OK
AT+SAPBR=0,1                    //To close a GPRS context.
OK
```