



A company of SIM Tech

SIM7500_SIM7600

Series_MQTT_ATC_V1.01



Development Kit



Hardware Design



Reference Design



Software Design

SIMCOM CONFIDENTIAL FILE

Document Title:	SIM7500_SIM7600 Series_MQTT AT Command Set
Version:	1.01
Date:	2017-09-20
Status:	Release
Document ID:	SIM7500_SIM7600 Series_MQTT_ATC_V1.01

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 Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © Shanghai SIMCom Wireless Solutions Ltd. 2017

Contents

Version History	2
1 Introduction.....	3
2 AT commands	3
2.1 AT+CMQTTSTART Open network.....	3
2.2 AT+CMQTTSTOP Close network.....	4
2.3 AT+CMQTTACCQ Acquire a client	5
2.4 AT+CMQTTREL Release a client	6
2.5 AT+CMQTTWILLTOPIC Set the will topic	7
2.6 AT+CMQTTWILLMSG Set will message	8
2.7 AT+CMQTTCNCTTIMEOUT Set retry timeout for connection	9
2.8 AT+CMQTTCONNECT Connect to the server.....	10
2.9 AT+CMQTTDISC Disconnect from the server	11
2.10 AT+CMQTTTOPIC Set publishing topic	13
2.11 AT+CMQTTPAYLOAD Set publishing message.....	14
2.12 AT+CMQTTTPUB Send a PUBLISH message to server	15
2.13 AT+CMQTTSUBTOPIC Set one topic for SUBSCRIBE message.....	16
2.14 AT+CMQTTSUB Send SUBSCRIBE message to server.....	17
2.15 AT+CMQTTUNSUBTOPIC Set one topic for unsubscribe message.....	18
2.16 AT+CMQTTUNSUB Send UNSUBSCRIBE message to server	19
2.17 MQTT URC	21
2.17.1 Disconnect passively.....	21
2.17.2 Receive topic published by MQTT server.....	21
2.18 Return code	23
3 Examples.....	24

Version History

Data	Version	Description of change	Author
2017-05-25	V1.00	New Version	
2017-06-21	V1.01	Modify SIM PING value	Yueying.ding
2017-06-26	V1.01	Modify AT+CMQTTDISC	Yueying.ding

SCOPE

This document describes how to use the MQTT function of SIM7500/SIM7600 series module through AT commands.

The develop software refer to MQTT V3.1.

This document is subject to change without notice at any time.

1 Introduction

MQ Telemetry Transport (MQTT) is a lightweight broker-based publish/subscribe messaging protocol designed to be open, simple, lightweight and easy to implement. This document apply to MQTT V3.1.

2 AT commands

2.1 AT+CMQTTSTART Open network

Description

This command is used to open the network.

SIM PIN	References
YES	Vendor

Syntax

Execution Command	Responses
AT+CMQTTSTART	OK
	+CMQTTSTART: <result>
	+CMQTTSTART: <result>
	OK
	[+CMQTTSTART: <result>]
	ERROR

Defined values

< result >

Refer to "Return code" chapter.

Examples

```
AT+CMQTTSTART
```

```
OK
```

```
+CMQTTSTART: 0
```

```
AT+CMQTTSTART
```

```
OK
+CMQTTSTART: 8
```

2.2 AT+CMQTTSTOP Close network

Description

This command is used to close the network. You should disconnect all of client's connection before executing this command.

SIM PIN	References
YES	Vendor

Syntax

Execution Command	Responses
AT+CMQTTSTOP	+CMQTTSTOP: <result>
	OK
	OK
	+CMQTTSTOP: <result>
	[+CMQTTSTOP: <result>]
	ERROR

Defined values

< result >
Refer to "Return code" chapter.

Examples

```
AT+CMQTTSTOP
+CMQTTSTOP: 0

OK
AT+CMQTTSTOP
+CMQTTSTOP: 9

OK
```

2.3 AT+CMQTTACCQ Acquire a client

Description

This command is used to acquire a client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTACCQ=?	+CMQTTACCQ: (0-<max_client_index>),(1-<max_clientID_len>) OK
Read Command	Responses
AT+CMQTTACCQ?	[+CMQTTACCQ: <client_index>, <clientID><CR><LF> [+CMQTTACCQ: <client_index>, <clientID><CR><LF>]] OK
Write Command	Responses
AT+CMQTTACCQ=<client_index>, <clientID>	OK +CMQTTACCQ: <client_index>,<result> ERROR

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<max_client_index>	a numeric parameter that identifies the max client index.
<clientID>	The UTF-encoded string. It specifies a unique identifier for the client. The string length is from 1 to 23.
<max_clientID_len>	23 - The max length of <clientID> string.
<result>	Refer to "Return code" chapter.

Examples

```
AT+CMQTTACCQ=0,"client c"
```

```
OK
```

2.4 AT+CMQTTREL Release a client

Description

This command is used to release a client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTREL=?	+CMQTTREL: (0-<max_client_index>)
Read Command	Responses
AT+CMQTTREL?	OK
Write Command	Responses
AT+CMQTTREL =<client_index>	OK
	+CMQTTREL: <client_index>,<result>
	ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<result>

Refer to "Return code" chapter.

Examples

```
AT+CMQTTREL=0
```

```
OK
```

```
AT+CMQTTREL=0
```

```
+CMQTTREL: 0,12
```

```
ERROR
```


2.5 AT+CMQTTWILLTOPIC Set the will topic

Description

This command is used to set will topic for the message.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTWILLTOPIC=?	+CMQTTWILLTOPIC: (0-<max_client_index>),(1-<max_topic_len>) OK
Read Command	Responses
AT+CMQTTWILLTOPIC?	OK
Write Command	Responses
AT+CMQTTWILLTOPIC= <client_index>,<req_len gth>	OK +CMQTTWILLTOPIC: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<max_topic_len>

The max length of will topic. The value is 1024

<req_length>

The length of input topic. The will topic should be UTF-encoded string. The range is from 1 to 1024.

<result>

Refer to “Return code” chapter.

Examples

```
AT+CMQTTWILLTOPIC=0,10
>
```

2.6 AT+CMQTTWILLMSG Set will message

Description

This command is used to set will message for the publish message.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTWILLMSG=?	+CMQTTWILLMSG: (0-<max_client_index>),(1-<max_willmsg_len>),(0-2) OK
Read Command	Responses
AT+CMQTTWILLMSG?	OK
Write Command	Responses
AT+CMQTTWILLMSG=<client_index>,<req_length>,<qos>	OK +CMQTTWILLMSG: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<max_willmsg_len>

The max length of will message. The value is 10240

<req_length>

The length of input will message. The will message should be UTF-encoded string. The range is from 1 to 10240.

<qos>

The qos value of the will message. The range is from 0 to 2.

<result>

Refer to “Return code” chapter.

Examples

```
AT+CMQTTWILLMSG=0,56,1
```

>

2.7 AT+CMQTTCNCTTIMEOUT Set retry timeout for connection

Description

This command is used to set the timeout interval value for connection.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTCNCTTIMEOUT =?	+CMQTTRETCNCTTIMEOUT: (0-<max_client_index>),(10-<max_interval>) OK
Read Command	Responses
AT+ CMQTTCNCTTIMEOUT?	+CMQTTCNCTTIMEOUT: <client_index>[,<interval>] [+CMQTTCNCTTIMEOUT: <client_index>[,<interval>]] OK
Write Command	Responses
AT+ CMQTTCNCTTIMEOUT =<client_index>,<interval>	OK ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<max_interval >

The max connect timeout interval value. The value is 180.

<interval >

The timeout interval value for connection. The range is from 10 to 180, 0 is not set the timeout value. The default value is 0.

Examples

```
AT+ CMQTTCNCTTIMEOUT=0,60
```

OK

2.8 AT+CMQTTCONNECT Connect to the server

Description

This command is used to establish connection to the server.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTCONNECT=?	+CMQTTCONNECT: (0-<max_client_index>),(<min_srvaddr_len>-<max_srvaddr_len>) ,(0-<max_keepalive_time>),(0-1) OK
Read Command	Responses
AT+CMQTTCONNECT?	+CMQTTCONNECT: <client_index>[,<server_addr>,<keepalive_time>,<clean_session>][,<user_name>[,<pass_word>]] OK
Write Command	Responses
AT+CMQTTCONNECT=<c lient_index>,<server_addr>, <keepalive_time>,<clean_se ssion>[,<user_name>[,<pass _word>]]	OK +CMQTTCONNECT: <client_index>,<result> +CMQTTCONNECT: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<server_addr>

The string that described the server address and port. The range of the string length is 9 to 256. The string should be like this “tcp://116.247.119.165:5141”, must begin with “tcp://”. If the <server_addr> not include the port, the default port is 1883.

<min_srvaddr_len>

The min length of server address, the value is 9.

<max_srvaddr_len>

The max length of server address, the value is 256.

<keepalive_time>

The Keep Alive timer, measured in seconds, defines the maximum time interval between messages received from a client. The range is from 1s to 64800s (18 hours).

<max_keepalive_time>

The max interval value of the keep alive timer

<clean_session >

The clean session flag.

0 - the server must store the subscriptions of the client after it disconnects. This includes continuing to store QoS 1 and QoS 2 messages for the subscribed topics so that they can be delivered when the client reconnects. The server must also maintain the state of in-flight messages being delivered at the point the connection is lost. This information must be kept until the client reconnects.

1 - the server must discard any previously maintained information about the client and treat the connection as "clean". The server must also discard any state when the client disconnects.

< user_name >

The user name identifies the name of the user who is connecting, which can be used for authentication. The string length is from 1 to 12.

<pass_word>

The password corresponding to the user who is connecting, which can be used for authentication. The string length is from 1 to 12.

<result>

Refer to "Return code" chapter.

Examples

```
AT+CMQTTCONNECT=0,"tcp://116.247.119.165:5141",20,1,"username","password"
```

```
OK
```

```
+CMQTTCONNECT: 0,0
```

2.9 AT+CMQTTDISC Disconnect from the server

Description

This command is used to disconnect from a client to a server.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTDISC=?	+CMQTTDISC: (0-<max_client_index>),(0, 60-180) OK
Read Command	Responses
AT+CMQTTDISC?	[+CMQTTDISC: <client_index>, <disc_state><CR><LF> [+CMQTTDISC: <client_index>, <disc_state><CR><LF>]] OK
Write Command	Responses
AT+CMQTTDISC=<client_index>,<timeout>	OK +CMQTTDISC: <client_index>,<result> +CMQTTDISC: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<timeout>

a numeric parameter that set timeout value for disconnection. The unit is second. The range is 60s to 180s. The default value is 0s (not set the timeout value).

<disc_state>

1 - disconnection

0 - connection

<result>

Refer to "Return code" chapter.

Examples

```
AT+CMQTTDISC=0, 120
```

```
OK
```

```
+CMQTTDISC: 0,0
```

2.10 AT+CMQTTTOPIC Set publishing topic

Description

This command is used to set publishing topic for the client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTTOPIC=?	+CMQTTTOPIC: (0-<max_client_index>),(1-<max_topic_len>) OK
Read Command	Responses
AT+CMQTTTOPIC?	OK
Write Command	Responses
AT+CMQTTTOPIC=<client_index>,<req_length><CR> <i>data for send</i>	OK +CMQTTTOPIC: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<max_topic_len>

a numeric parameter that identifies the max length of topic. The value is 1024.

<req_length>

a numeric parameter that identifies the length of input topic. The topic should be UTF-encoded string. The length is from 1 to 1024.

<result>

Refer to "Return code" chapter.

Examples

```
AT+CMQTTTOPIC=0,13
```

```
>
```

2.11 AT+CMQTTPAYLOAD Set publishing message

Description

This command is used to set publishing message for the client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTPAYLOAD=?	+CMQTTPAYLOAD: (0-<client_index>),(1-<max_pubmsg_len>) OK
Read Command	Responses
AT+CMQTTPAYLOAD?	OK
Write Command	Responses
AT+CMQTTPAYLOAD=<client_index>,<req_length><CR> <i>data for send</i>	OK +CMQTTPAYLOAD: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<max_pubmsg_len >

a numeric parameter that identifies the max length of publish message. The value is 10240.

<req_length>

a numeric parameter that identifies the length of input message. The message should be UTF-encoded string. The length is from 1 to 10240.

<result>

Refer to “Return code” chapter.

Examples

```
AT+CMQTTPAYLOAD=0,62.
```

```
>
```


2.12 AT+CMQTTPUB Send a PUBLISH message to server

Description

This command is used to publish a message. You should set this command after you set the publishing topic and message. If the result of publish topic is fail, the client will be disconnect and user need to connect again.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTPUB=?	+CMQTTPUB: (0-<max_client_index>),(0-<max_qos>),(1-<max_publish_interval>),(0-1),(0-1) OK
Read Command	Responses
AT+CMQTTPUB?	OK
Write Command	Responses
AT+CMQTTPUB=<client_index>,<qos>,<publish_interval>[,<retained>[,<dup>]]	OK +CMQTTPUB: <client_index>,<result> +CMQTTPUB: <client_index>,<result> ERROR

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<max_client_index>

a numeric parameter that identifies the max client index.

<qos>

The publish message's qos. The range is from 0 to 2.

<max_qos>

The max publish message's qos value.

<publish_interval >

The publishing timeout interval value, from the time client send PUBLISH to client receive the first response. The range is from 1s to 180s.

<max_publish_interval >

The max publishing timeout interval value.

< retained>

The Retain flag of the publish message. The value is 0 or 1. The default value is 0. When a client sends a PUBLISH to a server, if the Retain flag is set (1), the server should hold on to the message after it has been delivered to the current subscribers.

< dup>

The dup flag to the message. The value is 0 or 1. The default value is 0. The flag is set when the client or server attempts to re-deliver a message.

<result>

Refer to “Return code” chapter.

Examples

```
AT+CMQTTPUB=0,1,60
```

```
OK
```

```
+CMQTTPUB: 0,0
```

2.13 AT+CMQTTSUBTOPIC Set one topic for SUBSCRIBE message

Description

This command is used to set one topic for subscribing. You would set this command repeatedly for more than one subscribing topic.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTSUBTOPIC=?	+CMQTTSUBTOPIC: (0-<max_client_index>),(1-<max_subtopic_len>),(0-2) OK
Read Command	Responses
AT+CMQTTSUBTOPIC?	OK
Write Command	Responses
AT+CMQTTSUBTOPIC =<client_index>,<reqLength >,<qos><CR> <i>data for send</i>	OK +CMQTTSUBTOPIC: <client_index>,<result> +CMQTTSUBTOPIC: <client_index>,<result> ERROR

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<max_client_index>	a numeric parameter that identifies the max client index.
<qos>	The publish message's qos. The range is from 0 to 2.
<reqLength>	The input length of subscribe message. The range is from 1 to 1024.
<max_subtopic_len >	The max length of subtopic message.
<result>	Refer to "Return code" chapter.

Examples

```
AT+CMQTTSUBTOPIC=0,9,1
>
```

2.14 AT+CMQTTSUB Send SUBSCRIBE message to server

Description

This command is used to subscribe message for the client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTSUB=?	+CMQTTSUB: (0-<max_client_index>),(1-<max_subtopic_len>),(0-2),(0-1) OK
Read Command	Responses
AT+CMQTTSUB?	OK
Write Command	Responses
/* subscribe one topic*/ AT+CMQTTSUB=<client_index>,<reqLength>,<qos>[,	OK +CMQTTSUB: <client_index>,<result>

<pre> <dup>] <CR><i>data for send</i> /* subscribe one or more topics which be set by AT+CMQTTSUBTOPIC*/ AT+CMQTTSUB=<client_i ndex>[,<dup>] </pre>	<pre> +CMQTTSUB: <client_index>,<result> ERROR </pre>
--	--

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<max_client_index>	a numeric parameter that identifies the max client index.
<reqLength>	The input length of subscribe message. The range is from 1 to 1024.
<max_subtopic_len>	The max input length of subscribe topic.
<qos>	The publish message's qos. The range is from 0 to 2.
<dup>	The dup flag to the message. The value is 0 or 1. The default value is 0. The flag is set when the client or server attempts to re-deliver a message.
<result>	Refer to "Return code" chapter.

Examples

AT+CMQTTSUB=0,9,1,1
>
AT+CMQTTSUB=0
OK
+CMQTTSUB: 0,0

2.15 AT+CMQTTUNSUBTOPIC Set one topic for unsubscribe message

Description

This command is used to set one topic for unsubscribing. You would set this command repeatedly for more than one unsubscribing topic.

SIM PIN References

YES	Vendor
-----	--------

Syntax

Test Command AT+CMQTTUNSUBTOPIC =?	Responses +CMQTTUNSUBTOPIC: (0-<max_client_index>),(1-<max_unsubtopic_len>) OK
Read Command AT+CMQTTUNSUBTOPIC ?	Responses OK
Write Command AT+CMQTTUNSUBTOPIC =<client_index>,<reqLength> >><CR> <i>data for send</i>	Responses OK +CMQTTUNSUBTOPIC: <client_index>,<result> +CMQTTUNSUBTOPIC: <client_index>,<result> ERROR

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<max_client_index>	a numeric parameter that identifies the max client index.
<reqLength>	The input length of unsubscribe message. The range is from 1 to 1024.
<max_unsubtopic_len>	The max input length of unsubscribe message.
<result>	Refer to “Return code” chapter.

Examples

```
AT+CMQTTUNSUBTOPIC=0,11
>
```

2.16 AT+CMQTTUNSUB Send UNSUBSCRIBE message to server

Description

This command is used to unsubscribe topic for the client.

SIM PIN	References
YES	Vendor

Syntax

Test Command	Responses
AT+CMQTTUNSUB=?	+CMQTTUNSUB: (0-<max_client_index>),(1-<max_unsubtopic_len>),(0-1) OK
Read Command	Responses
AT+CMQTTUNSUB?	OK
Write Command	Responses
/*unsubscribe one topic*/ AT+CMQTTUNSUB=<client_index>,<reqLength>[,<dup>]<CR> <i>data for send</i>	OK +CMQTTUNSUB: <client_index>,<result> +CMQTTUNSUB: <client_index>,<result>
/*unsubscribe one or more topics which be set by AT+CMQTTUNSUBTOPIC*/ / AT+CMQTTUNSUB=<client_index>[,<dup>]	ERROR

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<max_client_index>	a numeric parameter that identifies the max client index.
<reqLength>	The input length of unsubscribe message. The range is from 1 to 1024.
<max_unsubtopic_len>	The max input length of unsubscribe message.
< dup>	The dup flag to the message. The value is 0 or 1. The default value is 0. The flag is set when the client or server attempts to re-deliver a message.
<result>	Refer to “Return code” chapter.

Examples

```
AT+CMQTTUNSUB=0,11,0
```

```
>
```

```
AT+CMQTTUNSUB=0
```

```
OK
```

```
+CMQTTUNSUB: 0,0
```

2.17 MQTT URC

2.17.1 Disconnect passively

Grammar	Description
+CMQTTCONNLOST: <client_index>,<cause><CR><LF>	When client disconnect passively, URC “+CMQTTCONNLOST” will be reported, then user need to connect MQTT server again.

Defined values

<client_index>

a numeric parameter that identifies a client. The range of permitted values is 0 to 1.

<cause>

The cause of disconnection.

1 – socket is closed passively.

2 – socket is reset.

3 – network is closed.

2.17.2 Receive topic published by MQTT server

Grammar	Description
/* If a client subscribes to one or more topics, any message published to those topics are sent by the server to the client */ <CR><LF>+CMQTTTRXSTART: <client_index>,<topic_total_len>,<payload_total_len> <CR><LF> <CR><LF>+CMQTTTRXTOPIC: <client_index>,<sub_topic_len><CR><LF><sub_topic >	If a client subscribes to one or more topics, any message published to those topics are sent by the server to the client. The following URC is used for transmitting the message published from server to client. 1) +CMQTTTRXSTART: <client_index>,<topic_total_len>,<payload_total_len>

```

/*for long topic, split to multiple packets to report*/
[<CR><LF>+CMQTTRXTOPIC:
<client_index>,<sub_topic_len><CR><LF><sub_topic
>]
<CR><LF>+CMQTTRXPAYLOAD:
<client_index>,<sub_payload_len><CR><LF><sub_pa
yload>
/*for long payload, split to multiple packets to report*/
[<CR><LF>+CMQTTRXPAYLOAD:
<client_index>,<sub_payload_len><CR><LF><sub_pa
yload>]
<CR><LF>+CMQTTRXEND: <client_index>

```

At the beginning of receiving published message, the module will send this command to user, and indicate client index with <client_index>, the topic total length with <topic_total_len> and the payload total length with <payload_total_len>.

2) +CMQTTRXTOPIC:
<client_index>,<sub_topic_len>\r\n<sub_topic>

After the command "+CMQTTRXSTART" received, the module will send second command to user, and indicate client index with <client_index>, the topic packet length with <sub_topic_len> and the topic content with <sub_topic> after "\r\n".

For long topic, it will be split to multiple packets to report and the command "+CMQTTRXTOPIC" will be send more than once with the rest of topic content. The sum of <sub_topic_len> is equal to <topic_total_len>.

3) +CMQTTRXPAYLOAD:
<client_index>,<sub_payload_len>\r\n<sub_payload>

After the command "+CMQTTRXTOPIC" received, the module will send third command to user, and indicate client index with <client_index>, the payload packet length with <sub_payload_len> and the payload content with <sub_payload> after "\r\n".

For long payload, the same as "+CMQTTRXTOPIC".

4) +CMQTTRXEND:
<client_index>

At last, the module will send fourth command to user and indicate the topic and payload have been transmitted completely.

Defined values

<client_index>	a numeric parameter that identifies a client. The range of permitted values is 0 to 1.
<topic_total_len>	The length of topic from MQTT server to client. The range is from 1 to 1024.
<payload_total_len>	the payload total length. The range is from 1 to 10240.
<sub_topic_len>	the topic packet length, The sum of <sub_topic_len> is equal to <topic_total_len>.
<sub_topic>	the topic content.
<sub_payload_len>	the payload packet length. The sum of <sub_payload_len> is equal to <topic_payload_len>.
<sub_payload>	the payload content.

Examples

```

AT+CMQTTSUB=0,13,1
>MQTT Examples
OK

+CMQTTSUB: 0,0

+CMQTRXSTART: 1, 13, 12
+CMQTRXTOPIC: 1, 13
MQTT Examples
+CMQTRXPAYLOAD: 1, 12
Hello World!
+CMQTRXEND: 1

```

2.18 Return code

<result>	The result of cmd.
0	successful.
1	failed.
2	bad UTF-8 string.
3	sock connect fail.
4	sock create fail.
5	sock close fail.
6	message receive fail.
7	network open fail.
8	network close fail.

- 9 – network not opened.
- 10 – client index error.
- 11 – no connection.
- 12 – invalid parameter.
- 13 – not supported operation.
- 14 – client is busy.
- 15 – require connection fail.
- 16 – sock sending fail.
- 17 – timeout.
- 18 – topic is empty.
- 19 – client is used.
- 20 – client not acquire resource.
- 21 – client not release.
- 22 – length out of range.
- 23 – network is opened.
- 24 – packet fail.
- 25 – DNS error.
- 26 – socket is closed by server.
- 27 – connection refused: unaccepted protocol version.
- 28 – connection refused: identifier rejected.
- 29 – connection refused: server unavailable.
- 30 – connection refused: bad user name or password.
- 31 – connection refused: not authorized.

3 Examples

Example: connect

AT+CMQTTSTART

OK

+CMQTTSTART: 0

AT+CMQTTACCQ=0,"client c"

OK

AT+CMQTTCONNECT=0,"tcp://116.247.119.165:5141",20,1

OK

+CMQTTCONNECT: 0,0

AT+CMQTTDISC=0,120

OK

+CMQTTDISC: 0,0

AT+CMQTTREL=0

OK

AT+CMQTTSTOP

+CMQTTSTOP: 0

OK

Example: connect with will topic

AT+CMQTTSTART

OK

+CMQTTSTART: 0

AT+CMQTTACCQ=0,"client c"

OK

AT+CMQTTWILLTOPIC=0,10

>will topic

OK

AT+CMQTTWILLMSG=0,56,1

>you'd better set will topic before setting will message.

OK

AT+CMQTTCONNECT=0,"tcp://116.247.119.165:5141",20,1,"username","password"

OK

+CMQTTCONNECT: 0,0

AT+CMQTTDISC=0,120

OK

+CMQTTDISC: 0,0

AT+CMQTTREL=0

OK

AT+CMQTTSTOP

+CMQTTSTOP: 0

OK

Example: publish topic

AT+CMQTTSTART

OK

+CMQTTSTART: 0

AT+CMQTTACCQ=0,"client c"

OK

AT+CMQTTWILLTOPIC=0,10

>will topic

OK

AT+CMQTTWILLMSG=0,56,1

>you'd better set will topic before setting will message.

```

OK
AT+MQTTCONNECT=0,"tcp://116.247.119.165:5141",20,1,"username","password"
OK

+MQTTCONNECT: 0,0
AT+MQTTPUBTOPIC=0,13
>publish topic
OK
AT+MQTTPUBMSG=0,62
>you'd better set publish topic before setting publish message.
OK
AT+MQTTPUB=0,1,60
OK

+MQTTPUB: 0,0
AT+MQTTDISC=0,60
OK

+MQTTDISC: 0,0
AT+MQTTREL=0
OK
AT+MQTTSTOP
+MQTTSTOP: 0

OK
Example: subscribe and unsubscribe
AT+MQTTSTART
OK

+MQTTSTART: 0
AT+MQTTACCQ=0,"client c"
OK
AT+MQTTWILLTOPIC=0,10
>will topic
OK
AT+MQTTWILLMSG=0,56,1
>you'd better set will topic before setting will message.
OK
AT+MQTTCONNECT=0,"tcp://116.247.119.165:5141",20,1,"username","password"
OK

+MQTTCONNECT: 0,0
AT+MQTTSUB=0,9,1,1

```

```
>subscribe  
OK
```

```
+CMQTTSUB: 0,0  
AT+CMQTTUNSUB=0,9,1
```

```
>subscribe  
OK
```

```
+CMQTTUNSUB: 0,0  
AT+CMQTTDISC=0,60  
OK
```

```
+CMQTTDISC: 0,0  
AT+CMQTTREL=0  
OK
```

```
AT+CMQTTSTOP  
+CMQTTSTOP: 0
```

```
OK
```

Example: receive publish message

```
+CMQTRXSTART: 0,9,6  
+CMQTRXTOPIC: 0,9  
ttttttt  
+CMQTRXPAYLOAD: 0,6  
yyyyyy  
+CMQTRXEND: 0
```

Contact us:

Shanghai SIMCom Wireless Solutions Ltd.

Add: Building A, SIM Technology Building, No.633, Jinzhong Road, Changning District

200335

Tel: +86 21 3252 3300

Fax: +86 21 3252 3301

URL: <http://www.sim.com/wm/>

SIMCOM CONFIDENTIAL FILE