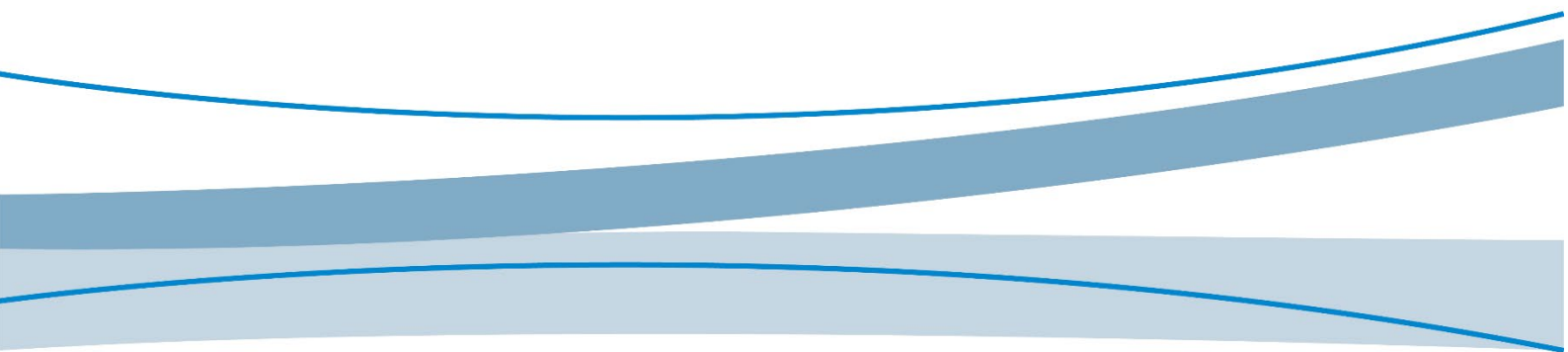




MTC

Application Guide_Standard MQTT

V1.2



Disclaimer

Any action you take in the course of using this document is at your own risk, and Fibocom shall not be liable for any damages or losses under any circumstances. Due to product version upgrade or other reasons, Fibocom reserves the right to modify any information in this document at any time without prior notice and any responsibility. Unless otherwise agreed, all statements, information and suggestions in this document do not constitute any express or implied guarantee.

This document may include the third-party information covering products, services, software, data, and so on. Fibocom does not control and assumes no responsibility for the third-party content, including but not limited to the accuracy, compatibility, reliability, availability, legitimacy, appropriateness, performance, non-infringement, and status update, unless otherwise specified in this document. Fibocom does not provide any guarantee or authorization for the third-party content mentioned or referenced in this document. If you need a third-party license, obtain it in an authorized or legal way, unless otherwise specified in this document.

Copyright Notice

Copyright © 2025 Fibocom Wireless Inc. All rights reserved.

Unless specially authorized by Fibocom, the recipient of the documents shall keep the documents and information received confidential, and shall not use them for any purpose other than the implementation and development of this project. Without the written permission of Fibocom, no unit or individual shall extract or copy part or all of the contents of this document without authorization, or transmit them in any form. Fibocom has the right to investigate legal liabilities for any offense and tort in connection with violation of confidentiality obligations, or unauthorized use or malicious use of the said documents and information in other illegal forms.

Trademark Statement

 The trademark is registered and owned by Fibocom Wireless Inc.

Other trademarks, product names, service names and company names appearing in this document are owned by their respective owners.

Contact Information

Website: <https://www.fibocom.com>

Address: 10/F-14/F, Block A, Building 6, Shenzhen International Innovation Valley, Dashi First Road, Xili Community, Xili Subdistrict, Nanshan District, Shenzhen

Tel: 0755-26733555

Contents

Applicable Model.....	2
Change History.....	3
1 Foreword	4
2 Reference Documents	5
3 MQTT Overview	6
3.1 Internal Dial-up.....	6
3.2 MQTT Data Interaction	7
3.3 AT Commands for MQTT Service.....	8
4 Application Examples.....	9
4.1 Non-encrypted Connection Examples.....	9
4.1.1 Creating a MQTT Connection.....	9
4.1.2 Subscribing to MQTT Topic	10
4.1.3 Publishing the MQTT Message.....	10
4.1.4 Canceling the Subscription of MQTT Topic	11
4.1.5 Disconnecting the MQTT Connection.....	11
4.2 Encrypted Connection Examples	11
4.2.1 TLS Certificate Configuration	11
4.2.2 Creating a MQTT Connection.....	12
4.2.3 Subscribing to MQTT Topic	13
4.2.4 Publishing the MQTT Message.....	13
4.2.5 Canceling the Subscription of MQTT Topic	14
4.2.6 Disconnecting the MQTT Connection.....	14
5 MQTT Acronyms and Abbreviations	15
5.1 Keep Alive.....	15
5.2 Persistence Message and Retain Flag	15
5.3 Will Message.....	15
5.4 CleanSession Flag	15
6 The MQTT Encrypted Connection Failed But Non-encrypted Connection Succeeded.....	17
Appendix A Terms and Abbreviations	18

Applicable Model

No.	Applicability Model	Description
1	All MTC products	NA

Change History

V1.2 (2025-01-06)	Correct text errors in the document.
V1.1 (2024-11-12)	Correct text errors in the document.
V1.0 (2023-11-22)	Initial version

1 Foreword

Fibocom module integrates the standard MQTT protocol, and the MCU implements the MQTT transmission function by sending AT commands to the module.

2 Reference Documents

Refer to TCP&UDP, SSL, MQTT and universal AT command manuals of the corresponding platform.

3 MQTT Overview

3.1 Internal Dial-up

Before using MQTT services, the module needs to initiate internal dial-up and check the network status synchronously to ensure that the module can access the network normally.

1. AT+CSQ?

It is used to query the current signal value. The larger the first value is within 0-31, the better the signal quality is. A value smaller than or equal to 12 indicates a weak signal, and a value greater than or equal to 21 indicates a strong signal. The value 99 indicates unknown or unavailable network.

2. AT+COPS?

It is used to query the carrier and whether the module is registered with the network.

3. AT+CPIN?

It is used to check the current status of the SIM card. Generally, a delay of 3 seconds is recommended.

4. AT+CGREG?

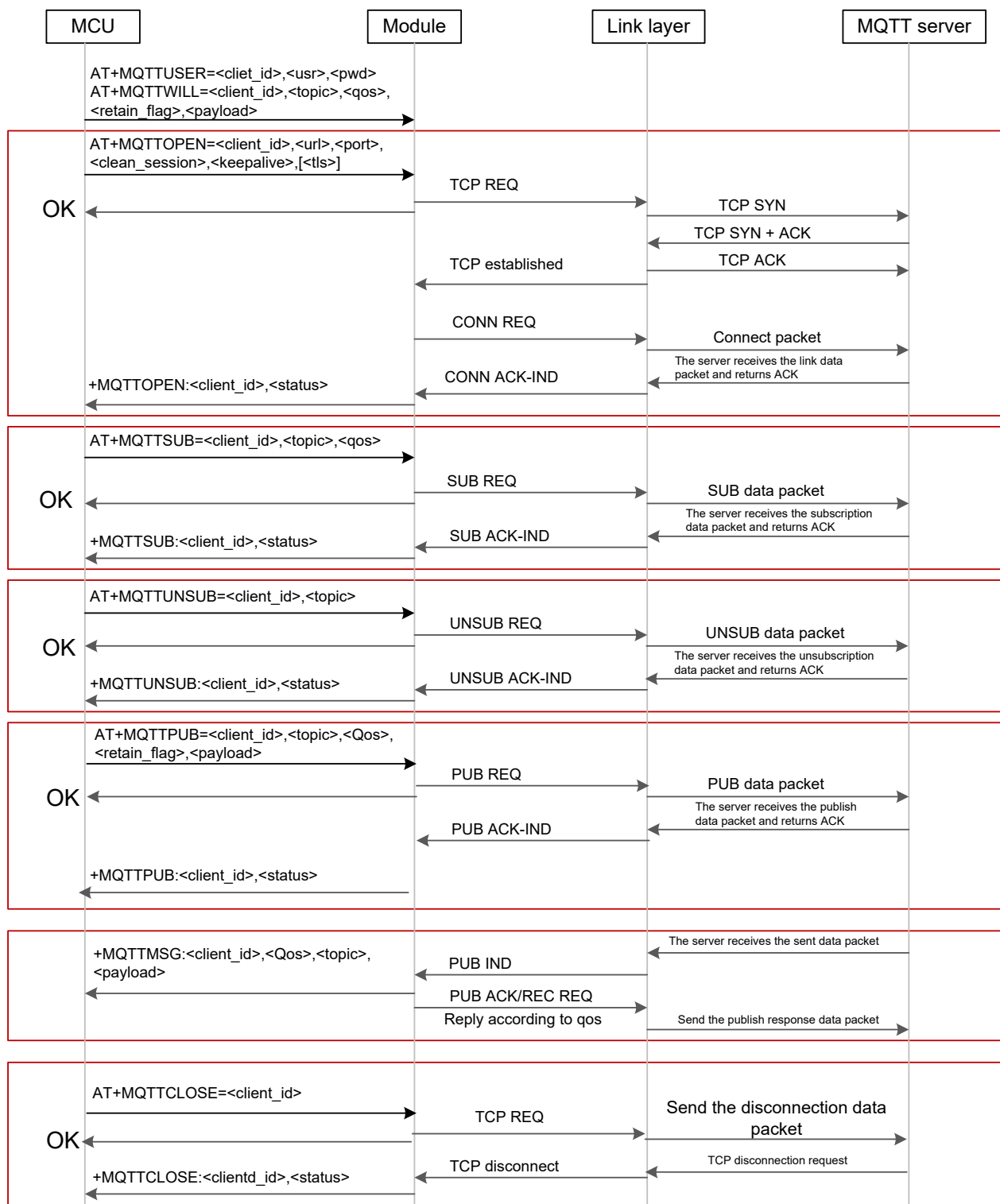
If +CGREG: 0,1 is returned, the local network data service is available. If +CGREG: 0,5 is returned, the data service in carrier roaming condition is available. In domestic scenarios except Hong Kong, Macao and Taiwan, false roaming may occur due to card configuration issues.

The network registration time is strongly related to the SIM card and the current network environment. For example, when the MC660x&MG66x module is used in countries other than China, the maximum network registration time may be 7 minutes.

5. AT+MIPCALL=<operation>,<apn>/<cid>

It is used to perform dial-up, activate the network, and request the carrier to allocate an IP address. If you need to configure the APN of the SIM card, consult the carrier. Generally, the APN of China Mobile network is "CMNET" or "CMNET", and the APN of China Unicom network is "3GNET". If other configurations are required, such as setting a username and password, refer to the corresponding AT command manuals.

3.2 MQTT Data Interaction



3.3 AT Commands for MQTT Service

AT Commands	Function
AT+MIPCALL	Request the operator for an IP address. Fill the APN of the SIM card in the position of CMNET, which can be obtained from the SIM card supplier. Generally, China Mobile uses CMNET or CMIOT as its APN, and China Unicom 3gnet as its APN. If you need to configure a user name and password, use the MIPCALL command in AT manual.
AT+MQTTUSER	Set MQTT authentication information.
AT+MQTTOPEN	Establish MQTT connection.
AT+MQTTSUB	Subscribe to a topic.
AT+MQTTPUB	Publish a message to a topic.
AT+MQTTUNSUB	Unsubscribe from a topic.
AT+MQTTCLOSE	Close MQTT connection.
AT+MQTTWILL	Set the will message; the setting command must be sent before MQTT connection is established.
AT+GTSSLFILE	Load TLS certificate and key.
AT+GTSSLVER	Query or set TLS version.
AT+GTSSLMODE	Set whether to verify the server certificate.
AT+GTSSLERR	Get SSL error code.
AT+CCLK	Query the current time of the module.

4 Application Examples

4.1 Non-encrypted Connection Examples

4.1.1 Creating a MQTT Connection

```
AT command ready //The module is started up successful

+SIM READY      //Ensure that the SIM card is ready
AT              //Query whether AT commands are available
OK
AT+CPIN?        //Check the SIM card status
+CPIN: READY    //Ensure that the SIM card is ready

OK
AT+CSQ?         //Query the signal strength
+CSQ: 20,99

OK
AT+CGREG?       //Query whether the data service is available
+CGREG: 0,1     //The data service is available

OK
AT+MIPCALL=1,"CMNET" //Request for an IP address
OK

+MIPCALL: 10.217.202.45

AT+MQTTUSER=1,"MC66x/test1","LeKH43ojsvwz1NS15EGImf5JIuPkCv9cPnu1DQUxGOk="
OK              //Set MQTT authentication information with the parameters of Client_id,
                user name and password in sequence, which are subject to regulations of the platform.
```

```
AT+MQTTWILL=1,"test1",0,0,"disconnect unexpectedly"
```

OK //Set MQTT will message with the parameters of Client_id, topic name, QoS level, reserved flag and message content in sequence.

```
AT+MQTTOPEN=1,"MC66x.mqtt.iot.bj.baidubce.com",1883,0,60 //Establish the MQTT connection with the parameters of Client_id, Remote IP address (domain name), remote port, clear flag and keep-alive connection value in sequence
```

OK

+MQTTOPEN: 1,1 //MQTT connection established successfully. Subsequent operation can proceed only after this value is returned.

4.1.2 Subscribing to MQTT Topic

```
AT+MQTTSUB=1,"test1",0 //Subscribe to topic with the parameters of Client_id, topic name and QoS level (0-2, some platforms do not support QoS Level 2) in sequence.
```

OK

+MQTTSUB: 1,1 //Subscribed successfully.

4.1.3 Publishing the MQTT Message

```
AT+MQTTPUB=1,"test1",0,0,"123" //Publish a message to the corresponding topic with the parameters of Client_id, topic name, QoS level (0-2, some platforms do not support QoS Level 2), reserved flag and message content
```

OK

+MQTTPUB: 1,1 //Published successfully.

+MQTTMSG: 1,0,"test1","123" //Receive a message from subscribed topic with the parameters of Client_id, QoS level, topic and message content in sequence.

```
AT+MQTTPUB=1,"test1",0,0,3 //Publish a message to the corresponding topic (ODM) with the parameters of Client_id, topic name, QoS level (0-2, some platforms do not support QoS Level 2), reserved flag and message data length in sequence. Enter 123 after receiving >.
```

>

OK

```
+MQTTPUB: 1,1 //Published successfully.
```

```
+MQTTMSG: 1,0,"test1","123" //Receive a message from subscribed topic with the  
parameters of Client_id, QoS level, topic and message content in sequence.
```

4.1.4 Canceling the Subscription of MQTT Topic

```
AT+MQTTUNSUB=1,"test1" //Unsubscribe from topic with the parameters of  
Client_id and topic name in sequence.
```

OK

```
+MQTTUNSUB: 1,1 //Unsubscribed successfully
```

4.1.5 Disconnecting the MQTT Connection

```
AT+MQTTCLOSE=1 //Close MQTT connection
```

OK

```
+MQTTCLOSE: 1,1 //MQTT connection closed successfully
```

4.2 Encrypted Connection Examples

4.2.1 TLS Certificate Configuration

//The following is the connection mode verified by the certificate:

```
AT+GTSSLVER? //Query TLS version
```

```
+GTSSLVER: 2 //TLS version: 1=SSL3.0; 2=TLS1.0; 3=TLS1.1; 4=TLS1.2
```

OK

```
AT+GTSSLVER=4          //Set TLS1.2 version (the TLS version may vary with different
cloud platforms, please contact with the relevant cloud platform)

OK

AT+GTSSLVER?            //Confirm TLS version

+GTSSLVER: 4

OK

AT+GTSSLFILE="CERTFILE",1220 //Load client certificate

>

OK

AT+GTSSLFILE="KEYFILE",1679 //Load client key

>

OK

AT+GTSSLFILE="TRUSTFILE",1758//Load root CA certificate

>

OK
```

4.2.2 Creating a MQTT Connection

```
AT+MQTTOPEN=1,"MC66x.mqtt.iot.bj.baidubce.com",8883,0,60,2 //Establish the MQTT
connection with the parameters of Client_id, remote IP address (domain name), remote
port, clear flag, keep-alive connection value and enabled TLS in sequence

OK

+MQTTOPEN: 1,1 //MQTT connection established successfully. Subsequent operation can
proceed only after this value is returned.
```

4.2.3 Subscribing to MQTT Topic

```
AT+MQTTSUB=1,"test1",0           //Subscribe to topic with the parameters of Client_id,  
topic name and QoS level (0-2, some platforms do not support QoS Level 2) in sequence.
```

OK

```
+MQTTSUB: 1,1                     //Subscribed successfully.
```

4.2.4 Publishing the MQTT Message

```
AT+MQTTPUB=1,"test1",0,0,"123"    //Publish a message to the corresponding topic  
with the parameters of Client_id, topic name, QoS level (0-2, some platforms do not  
support QoS Level 2), reserved flag and message content
```

OK

```
+MQTTPUB: 1,1                     //Published successfully.
```

```
+MQTTMSG: 1,0,"test1","123"       //Receive a message from subscribed topic with the  
parameters of Client_id, QoS level, topic and message content in sequence.
```

```
AT+MQTTPUB=1,"test1",0,0,3        //Publish a message to the corresponding topic (ODM)  
with the parameters of Client_id, topic name, QoS level (0-2, some platforms do not  
support QoS Level 2), reserved flag and message data length in sequence. Enter 123  
after receiving >.
```

>

OK

```
+MQTTPUB: 1,1                     //Published successfully.
```

```
+MQTTMSG: 1,0,"test1","123"      //Receive a message from subscribed topic with the  
parameters of Client_id, QoS level, topic and message content in sequence.
```

4.2.5 Canceling the Subscription of MQTT Topic

```
AT+MQTTUNSUB=1,"test1"          //Unsubscribe from topic with the parameters of  
Client_id and topic name in sequence.
```

OK

```
+MQTTUNSUB: 1,1                  //Unsubscribed successfully
```

4.2.6 Disconnecting the MQTT Connection

```
AT+MQTTCLOSE=1                  //Close MQTT connection
```

OK

```
+MQTTCLOSE: 1,1                  //MQTT connection closed successfully
```


5 MQTT Acronyms and Abbreviations

5.1 Keep Alive

Keep Alive is a time interval in the unit of seconds, expressed as a 16-bit word. It refers to the time when the client transmits a control message to the time when the next message is sent. The maximum time interval allowed to be idle (after the MQTT connection is established, the control message is automatically sent by the internal program of the module, and the user only needs to set the Keep Alive value when establishing the connection). If the Keep Alive value is non-zero, and the server does not receive a control packet from the client within 1.5 times the Keep Alive time, it must disconnect the client's network connection, and consider the network connection disconnected. If Keep Alive value is 0, it means that the Keep Alive function is disabled. In this situation, the server does not need to disconnect the connection of the client because the client is not active.



Regardless of the Keep Alive value, the client can be disconnected at any time as long as the server considers the client to be inactive or unresponsive.

5.2 Persistence Message and Retain Flag

If the Retain flag of a PUBLISH message sent by the client to the server is set to 1, the server must store the application message and its Quality of Service (QoS) so that it can be distributed to future subscriber matching the topic name.

5.3 Will Message

A Will message must be stored on the server and associated with this network connection once the connection request with a configured Will message is accepted. After the network connection is closed, the server must publish the Will message, unless the server deletes the Will message when it receives the DISCONNECT message.

Conditions for publishing the Will message include but are not limited to the following:

- The server detected an I/O error or a network failure.
- The client fails to communicate within the Keep Alive period.
- The client directly closes the network connection without sending a DISCONNECT message.
- The server closes the network connection due to a protocol error.

5.4 CleanSession Flag

If the CleanSession flag is set to 0, the server must resume communication with the client based on the state of the current session (identified using the client identifier). If there is no session associated with this client identifier, the server must create a new session. After the connection is disconnected, the client and server must save the session information. After a session with CleanSession flag 0 is disconnected, the server must save subsequent QoS 1 and QoS 2 messages as part of the session state,

if these messages match any subscriptions of the client at the time of disconnection. The server can also save QoS 0 messages that meet the same conditions. If the CleanSession flag is set to 1, the client and server must discard any previous session and start a new one. The session only lasts as long as the network connection. State data associated with this session cannot be reused by any subsequent sessions. Where:

The client's session state includes:

- QoS 1 and QoS 2 messages that have been sent to the server but have not yet been acknowledged.
- QoS 2 messages that have been received from the server but have not yet been acknowledged.

The server's session state includes:

- Whether the session exists, even if the rest of the session state is empty.
- Client's subscription information.
- QoS 1 and QoS 2 messages that have been sent to the client but have not yet been acknowledged.
- QoS 1 and QoS 2 messages to be transmitted to the client.
- QoS 2 messages that have been received from the client but have not yet been acknowledged.
- Optionally, the server decides whether to save QoS 0 messages to be sent to the client.

6 The MQTT Encrypted Connection Failed But Non-encrypted Connection Succeeded

Under normal circumstances, during the actual use of the communication module, in addition to the necessary network environment, NTP time synchronization is also required. In MQTT encrypted connection mode, the client and server need to perform a TLS handshake. When the client and server handshake without NTP time synchronization, the handshake packet contains an incorrect timestamp, which will cause the connection error. The correct approach is to perform NTP time synchronization before establishing an MQTT connection.

```
AT+MIPNTP="ntp.ntsc.ac.cn",123          //Use NTP server "ntp.ntsc.ac.cn:123"
to synchronize time

OK

+MIPNTP: 1                               //Time synchronization is successful

AT+CCLK?                                //Use the CCLK command to query the
current time

+CCLK: "23/09/04,15:17:41+32"

OK
```

Appendix A Terms and Abbreviations

Table 1. Term and acronyms

Acronyms	Full Spelling
APN	Access Point Name
IP	Internet Protocol
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol