



Perfect Wireless Experience  
完美无线体验

---

# FIBOCOM AT Commands User Manual\_MQTT

Version: V1.2.6

Date: 2019-10-14



### Applicability Type

No.	Type	Note
1	NL668-EAU-00	NA
2	NL668-EU-00/01/03	NA
3	M910-GL-00	NA
4	NL668-CN-00/01/02/03/04/10	NA
5	NL668-AM-00/01	NA
6	NL668-JP-00/01	NA
7	NL668-LA-00	NA
8	NL661-EU-00	NA

FIBOCOM  
 Confidential

## Copyright

Copyright ©2019 Fibocom Wireless Inc. All rights reserved.

Without the prior written permission of the copyright holder, any company or individual is prohibited to excerpt, copy any part of or the entire document, or transmit the document in any form.

## Attention

The document is subject to update from time to time owing to the product version upgrade or other reasons. Unless otherwise specified, the document only serves as the user guide. All the statements, information and suggestions contained in the document do not constitute any explicit or implicit guarantee.

## Trademark



The trademark is registered and owned by Fibocom Wireless Inc.

## Versions

Version	Date	Remarks
V1.0.0	2016-12-22	Initial version
V1.0.1	2017-03-13	Add MQTT Error chapter Add sample
V1.0.2	2017-09-15	Modify AT command format according to the normalization document Add some AT command about reading message
V1.1.0	2018-05-18	Add MQTTPUB parameter <Dataseize>
V1.2.0	2018-08-29	Add MQTTBREAK and MQTTUSER parameter <"ClientIDStr">
V1.2.1	2019-01-10	Add Application type NL668-EAU-00/NL668-EU-00/M910-GL-00
V1.2.2	2019-01-29	Modify + MQTTOPEN
V1.2.3	2019-03-06	Add Application type NL668-CN-00/01/02
V1.2.4	2019-03-19	Add Application type NL668-CN-04; Add MQTT Manual Usage Example chapter;
V1.2.5	2019-04-17	List all the specific application types
V1.2.6	2019-10-14	Add Application type NL668-CN-10

Reproduction forbidden without Fibocom Wireless Inc. written authorization – All Rights Reserved.

# Contents

1	MQTT Commands .....	5
1.1	+MQTTUSER, Set user name and password .....	5
1.2	+MQTTOPEN, Establish a connection via MQTT protocol .....	6
1.3	+MQTTCLOSE, Close the MQTT connection .....	7
1.4	+MQTTBREAK, MQTT disconnect report .....	8
1.5	+MQTTSUB, Subscribe A topic .....	9
1.6	+MQTTUNSUB, Unsubscribe A topic .....	10
1.7	+MQTTPUB, Publish A topic.....	10
1.8	+MQTTCONF, Config the form of receive message .....	12
1.9	+MQTTMSG, Receive a message from server .....	13
1.10	+MQTTMSGI, Receive a message's length from server.....	13
1.11	+MQTTREAD, Read the received message.....	14
1.12	+MQTTWILL, Set MQTT will.....	15
2	MQTT Error.....	17
3	MQTT Manual Usage Example .....	17
3.1	MQTT Connection Established.....	17
3.2	MQTT Message Process .....	18
3.3	MQTT Disconnect.....	18

FIBOCOM  
Confidential

# 1 MQTT Commands

## 1.1 +MQTTUSER, Set user name and password

### 1.1.1 Description

A connecting client can specify a user name and a password, this command should be set before open a MQTT connection. If the user name and password is incorrect, MQTT server will refuse the connection. This command is essential because of the server need these information to connect.

### 1.1.2 Syntax

Syntax	Possible response(s)
+MQTTUSER=<Client id>,<"Username">,<"Password">[,<"ClientID Str">]	OK or ERROR: <err>
+MQTTUSER?	+MQTTUSER: <Client id>,<"Username">,<"Password">[,<"ClientIDStr">] +MQTTUSER: <Client id>,<"Username">,<"Password">[,<"ClientIDStr">]  OK or +MQTTUSER: <Client id>,<"Username">,<"Password">[,<"ClientIDStr">]  OK or OK
+MQTTUSER=?	+MQTTUSER: (list of supported <Client id>s),<ulength>,<plength>[,<clength>]  OK

### 1.1.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	Yes	Yes	< 1s

## 1.1.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Username>: Registered Username for MQTT server, string type, the max length is 128 bytes.

<Password>: Registered Password for MQTT server, string type, the max length is 128 bytes.

<ClientIDStr>: Registered ClientID for MQTT server, string type, the max length is 23 bytes. It's a optional parameter, if not set, this field would be replaced by the IMEI of the module.

<ulength>: integer type value indicating the maximum length of field <Username>

<plength>: integer type value indicating the maximum length of field <Password>

<clength>: integer type value indicating the maximum length of field < ClientIDStr >

## 1.2 +MQTTOPEN, Establish a connection via MQTT protocol

### 1.2.1 Description

This command is used for opening the connection between the module and MQTT Server.

### 1.2.2 Syntax

Command	Possible response(s)
+MQTTOPEN=<Client id>,<"Remote IP/URL">,<Remote Port>,<Cleansession flag>,<Keepalive time>	OK +MQTTOPEN: <Client id>,<Status> or ERROR: <err>
+MQTTOPEN?	+MQTTOPEN: <Client id>  OK for each client id that can be opened or +MQTTOPEN: 0  OK if there are no free client.
+MQTTOPEN=?	+MPQTTOPEN: (list of supported <Client id>s),(< " Remote IP/URL">),(range of supported <Remote Port>s),(range of supported <Cleansession flag>s),(range of supported <Keepalive Time>s)

Command	Possible response(s)
	OK

### 1.2.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	No	Yes	< 1s

### 1.2.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

When id 1 and 2 can be opened, the Client id in the query response is 1,2

<Remote IP/URL>: IP: IP of the remote site in the format "AAA.BBB.CCC.DDD". The range of each octet is 0-255. Value can be written in 1, 2, or 3 digits. Host name: of remote site. The host-name convention should meet the rules as describe in RFC-1035 section: 2.3 Conventions. Syntax is not validated, except the maximum length (255 characters).

<Remote Port>: Port of remote site.

Port range: 1-65535 (decimal digits) for outgoing connection.

<Cleansession Flag>: integer type and valid value is 0, 1.

Type of protocol stack.

0 information of client will not be cleaned when the connection is closed

1 information of client will be cleaned when the connection is closed

<Keepalive Time>: Keepalive interval time, the range of time is 1-300(s).

<Status>: integer type and range 0-1.

0 fail.

1 success.

## 1.3 +MQTTCLOSE, Close the MQTT connection

### 1.3.1 Description

This command causes the Module to free the socket accumulating buffer and to close the socket.

### 1.3.2 Syntax

Command	Possible response(s)
+MQTTCLOSE=<Client id>	OK  +MQTTCLOSE: <Client id>,<Status>  or

Command	Possible response(s)
	ERROR: <err>
+MQTTCLOSE?	+MQTTCLOSE: <Client id>  OK for each client id that can be closed or +MQTTCLOSE: 0  OK if there are no free client id.
+MQTTCLOSE=?	+MQTTCLOSE: (list of supported <Client id>s)  OK

### 1.3.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	No	Yes	< 1s

### 1.3.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

When id 1 and 2 are both closed, the Client id in the query response is 1, 2

<Status>: integer type and range 0-1.

- 0 fail.
- 1 success.

## 1.4 +MQTTBREAK, MQTT disconnect report

### 1.4.1 Description

This is a unsolicited command, which is reported when the MQTT disconnected unexpected.

### 1.4.2 Syntax

Command	Possible response(s)
Unsolicited Response	+MQTTBREAK: <Client id>,<cause>

### 1.4.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	N/A	N/A	N/A	N/A

### 1.4.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<cause>: The cause of the MQTT disconnected, integer type and the range of the cause is 1-3.

- 1: MQTT disconnected by accident, such as the server initiative to disconnect the current connection or the client disconnect because of the ping request no responding.
- 2: Wireless Link disconnected, such as the valid IP address disappeared owing to the network failure.
- 3: GPRS network not registered, such as the report of the CGREG or CREG is 0.

## 1.5 +MQTTSUB, Subscribe A topic

### 1.5.1 Description

After the client is connected with MQTT server, you could subscribe one or more topic to MQTT server.

### 1.5.2 Syntax

Command	Possible response(s)
+MQTTSUB=<Client id>,<"Topic">,<Qos>	OK +MQTTSUB: <Client id>,<Status> or ERROR: <err>
+MQTTSUB=?	+MQTTSUB: (list of supported <Client id>s),<tlength >,(range of supported <Qos>s)  OK

### 1.5.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	No	Yes	< 1s

### 1.5.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Topic>: The topic should be printable ASCII characters, string type and the range of the topic length is 1-255.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

<Status>: integer type and range 0-1.

- 0 fail.

1 success.

<tlength>: integer type value indicating the maximum length of field <Topic>.

## 1.6 +MQTTUNSUB, Unsubscribe A topic

### 1.6.1 Description

This command will unsubscribe the specific topic from the server.

### 1.6.2 Syntax

Command	Possible response(s)
+MQTTUNSUB=<Client id>,<"Topic">	OK  +MQTTUNSUB: <Client id>,<Status>  or  ERROR: <err>
+MQTTUNSUB=?	+MQTTUNSUB: (list of supported <Client id>s),<tlength>  OK

### 1.6.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	No	Yes	< 1s

### 1.6.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Topic>: The topic should be printable ASCII characters, string type and the range of the topic length is 1-255.

<Status>: integer type and range 0-1

0 unsubscribe fail

1 unsubscribe success

<tlength>: integer type value indicating the maximum length of field <Topic>.

## 1.7 +MQTTPUB, Publish A topic

### 1.7.1 Description

After the client is connected with MQTT server, you can publish a message with a specific topic.

## 1.7.2 Syntax

Syntax	Possible response(s)
+MQTTPUB=<Client id>,<"Topic">,<Qos>,<Retain flag>,<"Payload"/>Datasize>	OK  +MQTTPUB: <Client id>,<Status>  or  ERROR: <err>
+MQTTPUB=?	+MQTTPUB: (list of supported <Client id>s),<length>,(range of supported <Qos>s),(range of supported <Retain flag>s),<plength>  OK

## 1.7.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	No	Yes	< 1s

## 1.7.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type and valid client id is 1, 2.

<Topic>: The topic should be printable ASCII characters, string type and the range of the topic length is 1-255.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

<Retain flag>: Retain flag, integer type, range 0, 1. The Retain flag indicates whether the server should retain the message which is published by the server.

- 0 this message should not be retained by the MQTT server
- 1 the MQTT server should retain a copy of the message

<Payload>: The message should be printable ASCII characters, string type and the range of message length is 0-1024.

<Datasize>: integer type and range 1-1024

This command cause data will be sending in HEX. After command received, Module will respond "><CR><LF>". Send any data in HEX. The data buffer range is 1<=Datasize<=1024 bytes. When

Module receive the corresponding length data, the data will be push automatic and returns to regular AT command mode.

<Status>: integer type and range 0-1

0 publish fail

1 publish success

<tlength>: integer type value indicating the maximum length of field <Topic>

<plength>: integer type value indicating the maximum length of field <Payload>

## 1.8 +MQTTCONF, Config the form of receive message

### 1.8.1 Description

After subscribe a topic, this command is decided to the form of the unsolicited event when any message about the topic is received.

### 1.8.2 Syntax

Command	Possible response(s)
+MQTTCONF=<mode>	OK or ERROR: <err>
+MQTTCONF?	+MQTTCONF: <mode> OK
+MQTTCONF=?	+MQTTCONF: (list of supported <mode>s) OK

### 1.8.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	Yes	Yes	< 1s

### 1.8.4 Defined Values

<mode>: the form of the unsolicited message, integer type, range 0,1.

0 default value, report the content of the topic and payload directly by the MQTTMSG command.

1 report the length of the topic and payload by the MQTTMSGI command.

## 1.9 +MQTTMSG, Receive a message from server

### 1.9.1 Description

After subscribe a topic, this unsolicited event will be sent by the Module to the terminal when any message about the topic is received. This command will show the content of the message.

### 1.9.2 Syntax

Command	Possible response(s)
Unsolicited Response	+MQTTMSG: <Client id>,<Qos>,<"Topic">,<"Payload">

### 1.9.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	N/A	N/A	N/A	N/A

### 1.9.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

## 1.10 +MQTTMSGI, Receive a message's length from server

### 1.10.1 Description

After subscribe a topic, this unsolicited event will be sent by the Module to the terminal when any message about the topic is received. This command will show the length of the topic and payload about the message.

### 1.10.2 Syntax

Command	Possible response(s)
Unsolicited Response	+MQTTMSGI: <Client id>,<Qos>,<length>,<plength>

### 1.10.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	N/A	N/A	N/A	N/A

### 1.10.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

<tlength>: integer type value indicating the received length of the topic.

<plength>: integer type value indicating the received length of the payload.

## 1.11 +MQTTREAD, Read the received message

### 1.11.1 Description

After received a message by the MQTTMSGI command, this command is used to read the content of the message.

### 1.11.2 Syntax

Command	Possible response(s)
+MQTTREAD=<Client id>	+MQTTREAD: <Client id>,0,0,0  OK if the client id is not received message or +MQTTREAD: <Client id>,<Qos>,<tlength>,<plength> <"Topic">,<"Payload">  OK if the client id has received message or ERROR: <err>
+MQTTREAD=?	+MQTTREAD: (list of supported <Client id>s)  OK

### 1.11.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	Yes	Yes	< 1s

### 1.11.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type. Valid client id is 1, 2.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

<length>: integer type value indicating the received length of the topic.

<payload>: integer type value indicating the received length of the payload.

## 1.12 +MQTTWILL, Set MQTT will

### 1.12.1 Description

This command is option, and is needed to set before open a MQTT connection. If set the will message, When the client disconnect abnormally from server, the will message will be sent to the client that subscribe the will topic.

### 1.12.2 Syntax

Command	Possible response(s)
+MQTTWILL=<Client id>,<"Topic">,<Qos>,<Retain flag>,<"Payload">	OK or ERROR: <err>
+MQTTWILL?	+MQTTWILL: <Client id>,<"Topic">,<Qos>,<Retain flag>,<"Payload"> +MQTTWILL: <Client id>,<"Topic">,<Qos>,<Retain flag>,<"Payload">  OK or

Command	Possible response(s)
	+MQTTWILL: <Client id>,<"Topic">,<Qos>,<Retain flag>,<"Payload">  OK or OK
+MQTTWILL=?	+MQTTWILL: (list of supported <Client id>s),<tlength>,(range of supported <Qos>s),(range of supported <Retain flag>s),<plength>  OK

### 1.12.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
Yes	No	Yes	Yes	< 1s

### 1.12.4 Defined Values

<Client id>: A unique number that identifies a connection, integer type and valid client id is 1, 2.

<Topic>: The topic should be printable ASCII characters, string type and the range of the topic length is 1-255.

<Qos>: The quality of service, integer type, range 0-2.

- 0 At most once, the message may not be delivered
- 1 At least once, the message will be delivered, but may be delivered more than once in some circumstances.
- 2 Once and one only, the message will be delivered exactly one.

<Retain flag>: Retain flag, integer type, range 0, 1. The Retain flag indicates whether the server should retain the Will message which is published by the server.

- 0 this message should not be retained by the MQTT server
- 1 the MQTT server should retain a copy of the message

<Payload>: The payload message should be printable ASCII characters, string type and the range of will message length is 0-1024.

<tlength>: integer type value indicating the maximum length of field <Topic>.

<plength>: integer type value indicating the maximum length of field <Payload>.

## 2 MQTT Error

Parameter	Description
<Err>	700 "Unacceptable protocol version "
	701 "Identifier rejected"
	702 "Server unavailable"
	703 "Bad user name or password"
	704 "Not authorized"

## 3 MQTT Manual Usage Example

A complete MQTT business operation consists of connecting the broker server, subscribing and publishing the message of the very topic, disconnecting the connection. The steps listed below for normal use are as follows:

### 3.1 MQTT Connection Established

```

AT+COPS?           //Query network registration status
+COPS: 0,0,"CHN-UNICOM",7 //network information
OK
AT+MIPCALL=1       //Dialup and get local IP
OK
+MIPCALL: 10.2.83.148
AT+MQTTOPEN?      //Query the available connections
+MQTTOPEN: 1,2    //ClientID 1 and 2 are available
OK
AT+MQTTUSER=1,"username","passwd","ClientIDStr" //Set MQTT username,password and ClientIDStr
//that registered ClientID for MQTT server
OK
AT+MQTTOPEN=1,"MQTT BrokerServer",1883,0,80 //MQTT connection start
OK
+MQTTOPEN: 1,1    //ClientID 1 make the connection established with the BrokerServer
    
```

## 3.2 MQTT Message Process

```
AT+MQTTSUB=1,"topic1",0 //Subscribe a topic named topic1
OK
+MQTTSUB: 1,1 //Subscribe topic successfully
+MQTTMSG: 1,0,"topic1","Test" //Receive information of previously published topic
AT+MQTTPUB=1,"topic1",1,0,"9" //Publish a topic named topic1
OK
+MQTTPUB: 1,1 //Publish topic successfully
AT+MQTTUNSUB=1,"topic1" //Unsubscribe the topic in clientID 1
OK
+MQTTUNSUB: 1,1 //Unsubscribe topic successfully
AT+MQTTWILL=1,"topic1",1,1,"Tell you, I'm dead." //Publish last words
OK
```

## 3.3 MQTT Disconnect

```
AT+MQTTCLOSE=1 //Close the socket of clientID 1 normally
```

OK

```
+MQTTCLOSE: 1,1
```

When the MQTT connection disconnected unexpected, the device may which is report as follows:

```
+MQTTBREAK: 1,1
```

```
+MQTTCLOSE: 1,1
```