

NB-IoT Modules TCP/UDP Application Notes

Issue 1.0

Date 2018-12-06

Neoway Product Document



Copyright © Neoway Technology Co., Ltd 2018. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Neoway Technology Co., Ltd.

Neoway 有方 is the trademark of Neoway Technology Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

This document provides guide for users to use NB-IoT Modules.

This document is intended for system engineers (SEs), development engineers, and test engineers.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE DUE TO PRODUCT VERSION UPDATE OR OTHER REASONS.

EVERY EFFORT HAS BEEN MADE IN PREPARATION OF THIS DOCUMENT TO ENSURE ACCURACY OF THE CONTENTS, BUT ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

Neoway provides customers complete technical support. If you have any question, please contact your account manager or email to the following email addresses:

Sales@neoway.com

Support@neoway.com

Website: <http://www.neoway.com>

Contents

1 Overview.....	1
2 TCP and UDP Services	2
2.1 Initialization Process	2
2.2 Non-Transparent TCP/UDP Transmission	3
2.2.1 Process of Non-Transparent Transmission	3
2.2.2 Example.....	5
2.3 Transparent TCP/UDP Transmission	6
2.3.1 Process of Transparent Transmission	6
2.3.2 Example.....	7
2.4 External Protocol Stack.....	9
2.4.1 External Protocol Stack Process	9
2.4.2 Example.....	9
3 Troubleshooting	10
3.1 Failing to Register with A Network.....	10
3.2 Failing to Obtain IP Address.....	10
3.3 Failing to Set Up Non-Transparent TCP/UDP Connection	10
3.4 Failing to Send TCP/UDP Data	11
3.5 Failing to Receive TCP/UDP Data	12

About This Document

Scope

This document is applicable to NB-IoT Modules and describes how to use the TCP and UDP functions and their AT command processes.




Audience

This document is intended for [system engineers \(SEs\)](#), [development engineers](#), and [test engineers](#).

Change History

Issue	Date	Change	Changed By
1.0	2018-09	Initial draft	Cenny

Conventions

Symbol	Indication
 Warning	This warning symbol means danger. You are in a situation that could cause fatal device damage or even bodily damage.
 Caution	Means reader be careful. In this situation, you might perform an action that could result in module or product damages.
 Note	Means note or tips for readers to use the module

1 Overview

This document describes how to use the TCP and UDP protocols of Neoway NB-IoT modules. Neoway NB-IoT modules embed TCP and UDP protocols and also allows to use external protocols.

Embedded TCP and UDP protocols provides various AT commands. For details, see AT command manuals of NB-IoT modules.

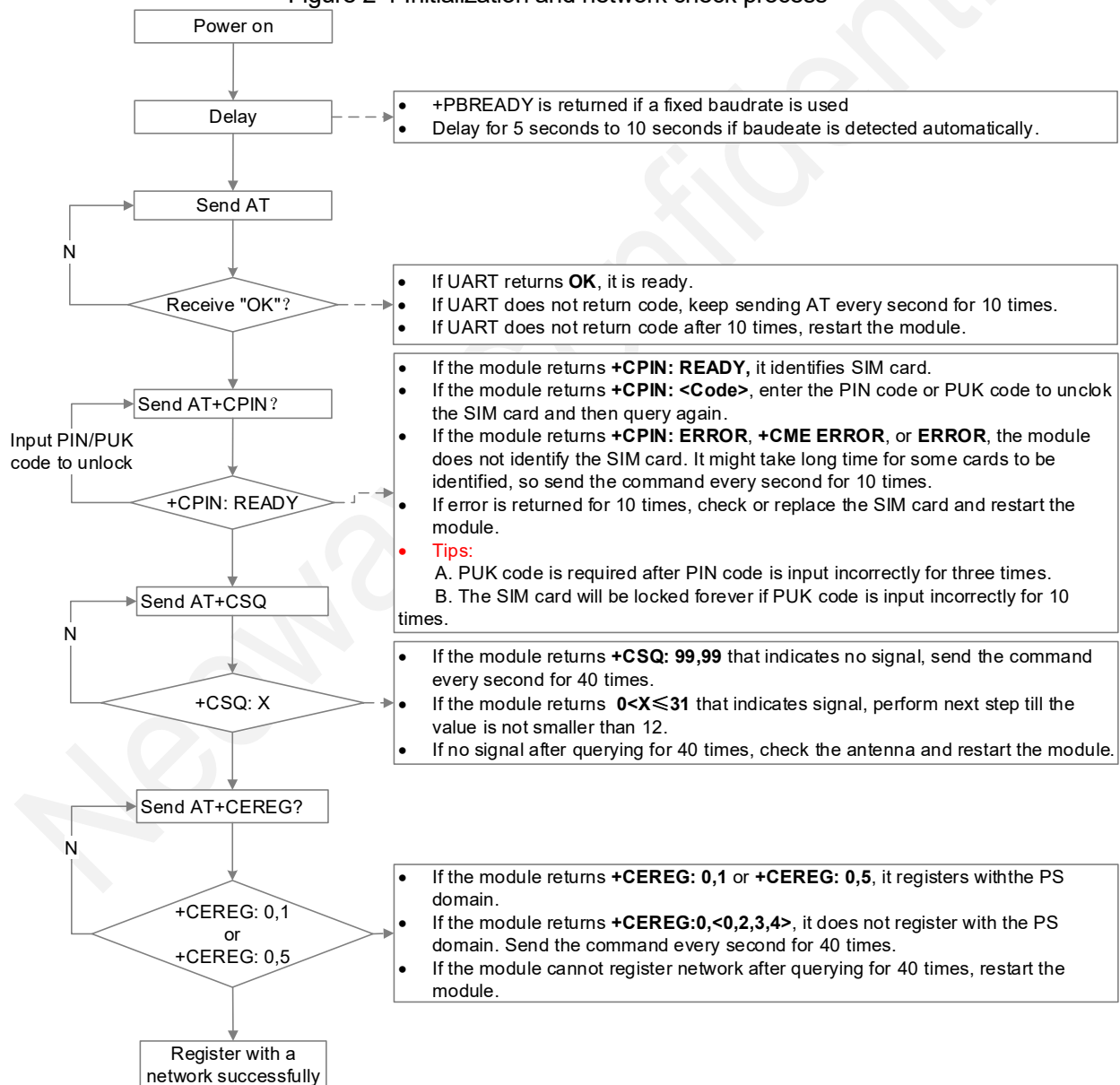
TCP and UDP support transparent and non-transparent transmission.

2 TCP and UDP Services

2.1 Initialization Process

To use the TCP and UDP functions after the module is powered up and started, ensure that the module is initialized and registers with a network.

Figure 2-1 Initialization and network check process



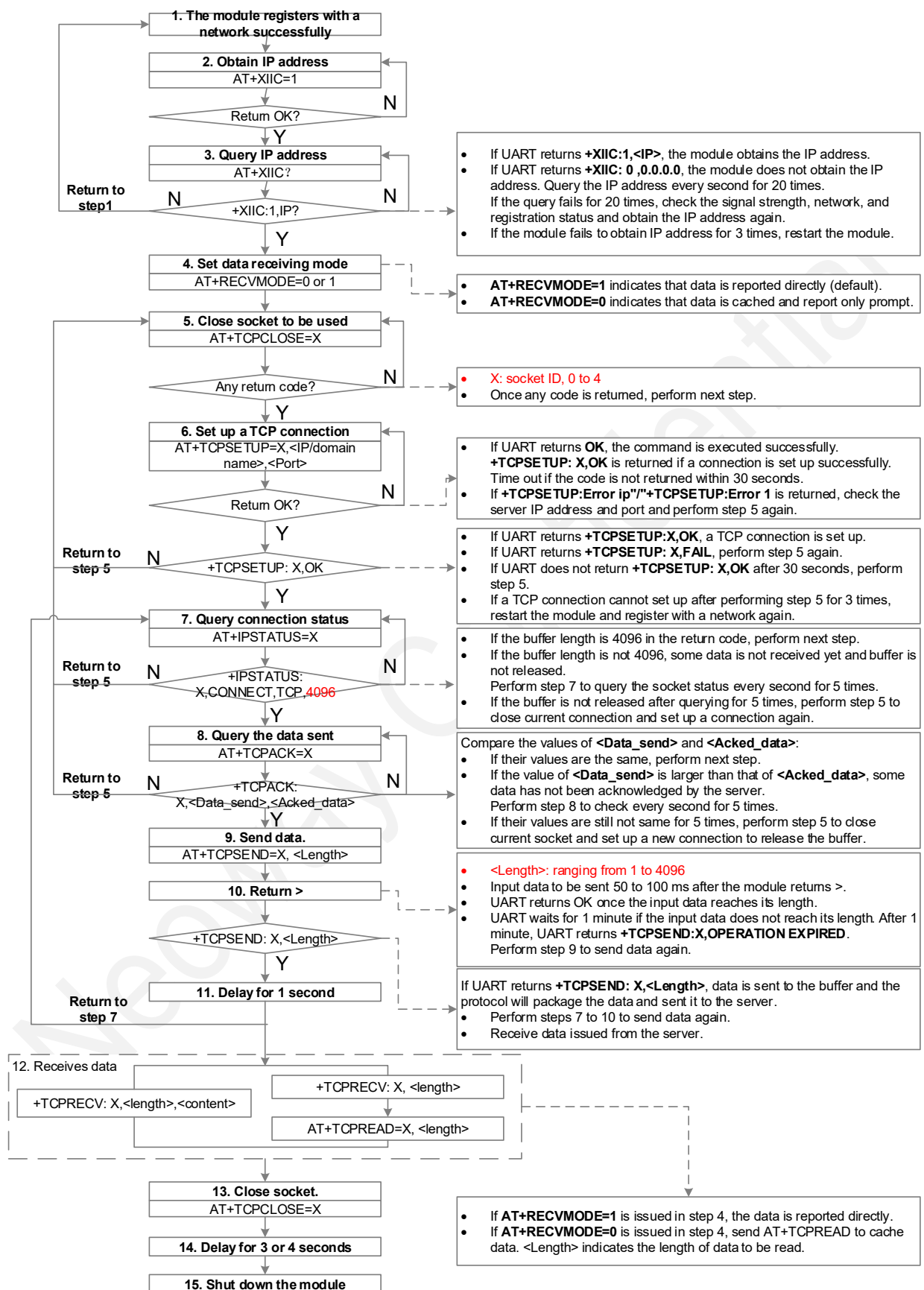
2.2 Non-Transparent TCP/UDP Transmission

2.2.1 Process of Non-Transparent Transmission

Figure 2-2 shows the process of non-transparent TCP transmission.

Neoway Confidential

Figure 2-2 Flowchart of non-transparent transmission



UDP transmission follows the similar procedures.

When you use non-transparent UDP transmission, replace the TCP commands with UDP commands.

AT+TCPCLOSE replaced by **AT+UDPCLOSE**

AT+TCPSETUP replaced by **AT+UDPSETUP**

AT+TCPSEND replaced by **AT+UDPSEND**

AT+TCPREAD replaced by **AT+UDPREAD**

UDP transmission does not include an acknowledgment mechanism. Remove step 8 from the flowchart when you use UDP.

Data can be sent by **AT+UDPSEND=<n>[,<length>][,<content>]<CR>**. For more details, see the latest AT command manuals.

2.2.2 Example

1. Check whether the module registers with a network.

```
AT //Check whether the module responds to AT commands.
OK
AT+CPIN?
+CPIN: READY //Check whether the module identifies the SIM card.
OK
AT+CSQ
+CSQ: 31, 99 //Check the signal quality.
OK
AT+CEREG?
+CEREG: 0,1 //Check whether the module registers with PS domain.
OK
```

2. Obtain an IP address, set up a non-transparent TCP connection, and transmit and receive data.

```
At+XIIC=1 //Obtain the IP address.
OK
AT+XIIC? //Query the IP address.
+XIIC: 1,10.162.64.171
AT+RECVMODE? //Query data receiving mode.
+RECVMODE: 1
OK
AT+RECVMODE=? //Query data receiving mode that the module supports.
+RECVMODE: (0-1)
OK
AT+RECVMODE=0 //Set data receiving mode to buffer mode.
OK
AT+TCPCLOSE=0 //Close TCP connection
+TCPCLOSE: ERROR
AT+TCPSETUP=0,58.60.184.213,10066 //Set up a TCP connection.
OK
+TCPSETUP: 0,OK //A TCP connection is set up successfully.
AT+IPSTATUS=0 //Check the status of the TCP connection.
+IPSTATUS: 0,CONNECT,TCP,4096
AT+TCPACK=0 //Check whether any data is sent.
+TCPACK: 0,0,0
```

```
AT+TCPSSEND=0,15 //Send 15-byte data.
>
AT12345678ATATA
OK
+TCPSSEND: 0,15 //15-byte data is sent successfully.
+TCPRECV: 0,15 //The module buffers 15-byte data that it receives because
transparent response is set on server.

AT+TCPSSEND=0,10 //Send 10-byte data.
>
1234567890
OK
+TCPSSEND: 0,10 //10-byte data is sent successfully.
+TCPRECV: 0,25 //The module buffers 25-byte data in total.

AT+TCPSSEND=0,10 //Send 10-byte data.
>
qqqqqqqqqq
OK
+TCPSSEND: 0,10 //10-byte data is sent successfully.

AT+TCPSSEND=0,10 //Send 10-byte data.
>
qqqqqqqqqq
OK
+TCPSSEND: 0,10 //10-byte data is sent successfully.

AT+TCPSSEND=0,10 //Send 10-byte data.
>
1234567890
OK
+TCPSSEND: 0,10 //10-byte data is sent successfully.
+TCPRECV: 0,35
+TCPRECV: 0,55 //The module buffers 55-byte data in total.

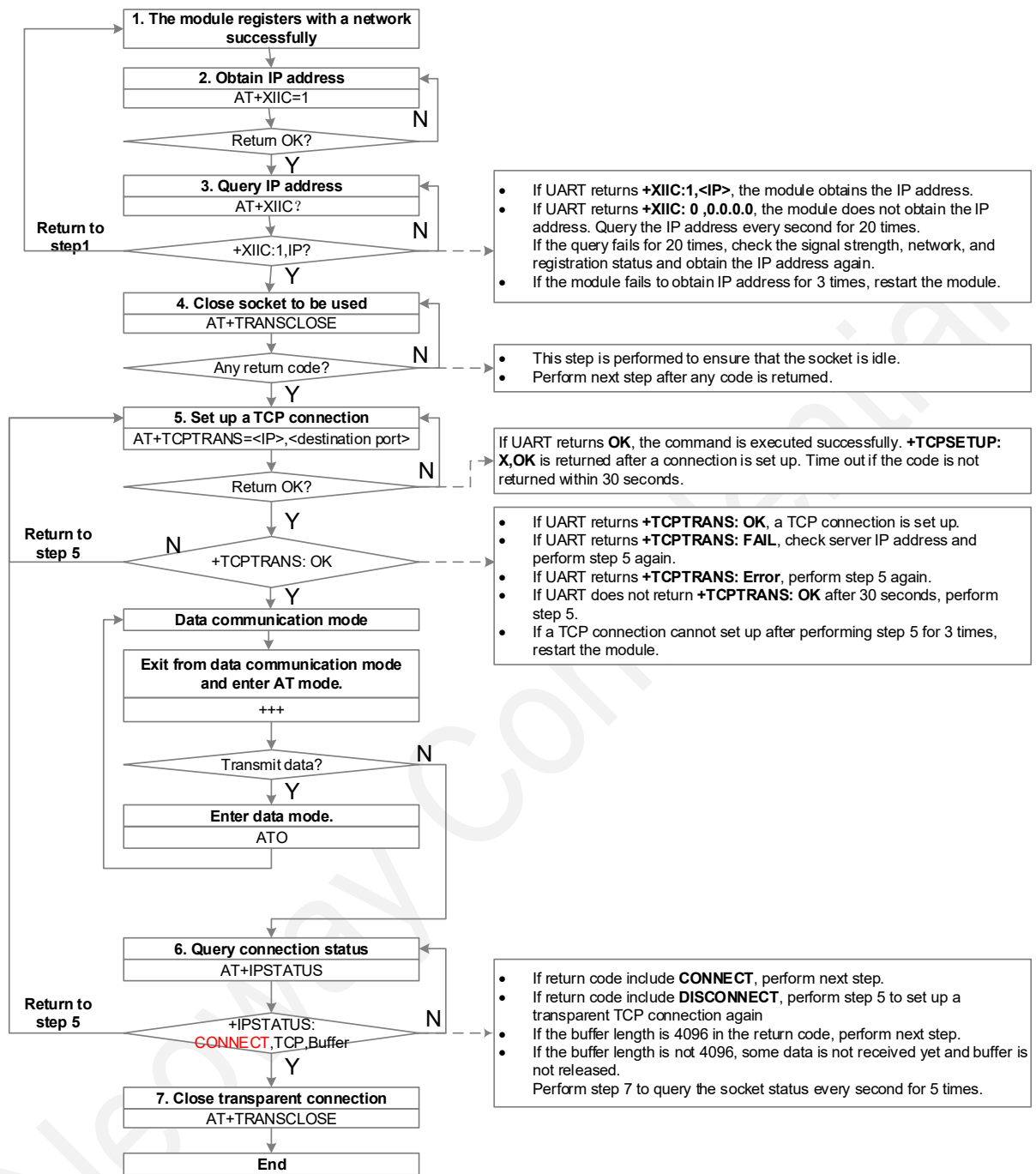
AT+TCPREAD=0,26 //Read 26-byte data from the buffer.
+TCPREAD: 0,26,AT12345678ATATA1234567890q
OK
AT+TCPREAD=0,5 //Read 5-byte data from the buffer.
+TCPREAD: 0,5,qqqqq
OK
AT+TCPREAD=0,30 //Read 30-byte data from the buffer.
+TCPREAD: 0,24,qqqqqqqqqqqqqq1234567890 //24-byte data is read from the buffer.
OK
AT+TCPCLOSE=0
+TCPCLOSE: 0,OK
```

2.3 Transparent TCP/UDP Transmission

2.3.1 Process of Transparent Transmission

Figure 2-3 shows the process of transparent TCP transmission.

Figure 2-3 Flowchart of transparent TCP transmission



UDP transmission follows the similar procedures.

When you use transparent UDP transmission, replace **AT+TCPTRANS** in the flowchart with **AT+UDPTRANS**.

2.3.2 Example

1. Check whether the module registers with a network.

```
AT //Check whether the module responds to AT commands.
```

```
OK
AT+CPIN?
+CPIN: READY          //Check whether the module identifies the SIM card.
OK
AT+CSQ
+CSQ: 31, 99          //Check the signal quality.
OK
AT+CEREG?
+CEREG: 0,1           //Check whether the module registers with a network
successfully.
OK
```

2. Obtain an IP address, set up a transparent TCP connection, and transmit and receive data.

```
At+XIIC=1              //Obtain the IP address.
OK
AT+XIIC?               //Query the IP address.
+XIIC: 1,10.165.202.43
OK
AT+TRANSCLOSE
ERROR

AT+TCPTRANS=58.60.184.213,10066 //Set up a transparent TCP connection.
OK
+TCPTRANS: OK

Send -> 123456          //Enter transparent transmission mode.
Receive-> 123456

Send -> +++            //Send +++ to exit from transparent transmission mode.
Receive->
OK

AT+NVSETPM?           //Enter AT mode to respond to AT commands.
pm is 0
OK

ATO                   //Switch to transparent transmission mode.
CONNECT

Send-> AT+CPSMS?       //Process of sending data
Receive-> AT+CPSMS?

Send -> +++            //Send +++ to exit from transparent transmission mode.
Receive->
OK

AT+IPSTATUS           //Check the status of the TCP connection.
+IPSTATUS: CONNECT,TCP,4096

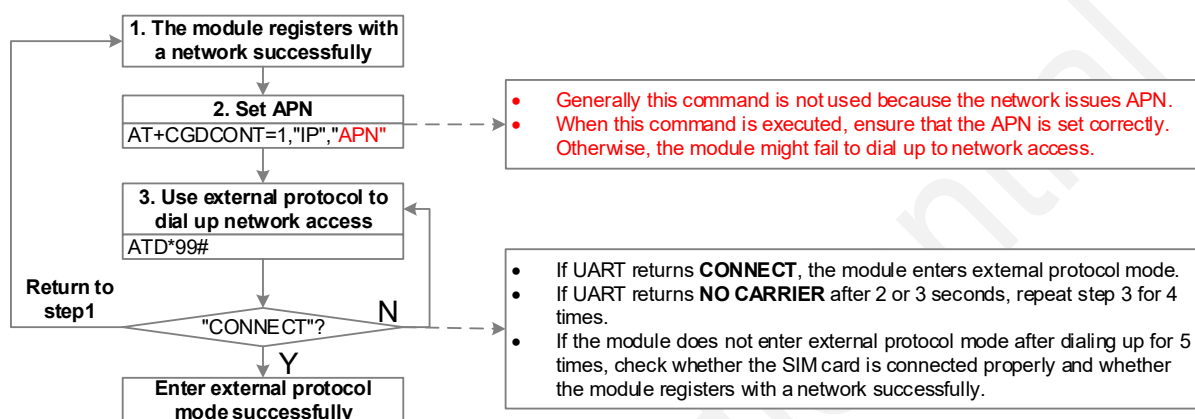
AT+TRANSCLOSE         //Close transparent connection.
+TRANSCLOSE: 0,OK
```

2.4 External Protocol Stack

2.4.1 External Protocol Stack Process

If you use external protocols instead of the embedded protocols and dial up network access through UART, see the following process.

Figure 2-4 Flowchart of external protocol stack



2.4.2 Example

1. Check whether the module registers with a network.

```

AT //Check whether the module responds to AT commands.
OK
AT+CPIN?
+CPIN: READY //Check whether the module identifies the SIM card.
OK
AT+CSQ
+CSQ: 31, 99 //Check the signal quality.
OK
AT+CEREG?
+CEREG: 0,1 //Check whether the module registers with a network.
OK
  
```

2. Set APN and enter external protocol mode.

```

AT+CGDCONT=1,"IP","ctnb" //Set APN based on the SIM card you use.
OK

ATD*99# //Enter external protocol stack.
CONNECT
  
```

3 Troubleshooting

3.1 Failing to Register with A Network

Step 1: Send **AT+CPIN?** to check whether the module identifies the SIM card.

If it does not identify the SIM card, check the following items:

- Whether the SIM card is inserted properly
- Whether the SIM card is damaged
- Whether the SIM card is locked
- Whether the SIM card connector is soldered properly
- Whether the contacts of SIM card and connector are damaged (dirt or oxidation)

If the module still does not identify the SIM card after the above troubleshooting steps are performed, change the SIM card.

Step 2: Check whether the antenna is installed properly and send **AT+CSQ** to check whether signal quality meets communication requirements.

3.2 Failing to Obtain IP Address

Step 1: If the module registers with a network properly but fails to obtain the IP address, check the following items:

- Whether the SIM card is out of credibility
- Whether the APN is set properly (send AT+CGDCONT?)

Step 2: If the module still fails to obtain IP address after eliminating the above issues, contact your operators to help check the SIM card and server configurations.

3.3 Failing to Set Up Non-Transparent TCP/UDP Connection

If the module obtains the IP address but fails to set up non-transparent TCP/UDP connection, troubleshoot in the following procedures:

- Step 1:** Use another client to connect to the server to confirm whether the IP address and port of the server are configured correctly and the server starts to listen at the port.
- Step 2:** Send **AT+PING=<IP/domain name>** to check whether the module can access the server using the IP address that it obtains.
- If the module can access the server through ping command but fails to set up a non-transparent TCP connection, check the server and port configurations.
 - If the module cannot access the server through ping command, check whether the APN is set correctly and whether the SIM card is configured correctly.
- Step 3:** Check whether the SIM card is configured correctly.

Contact your operator to confirm whether the SIM card is allowed to access the server IP address.

3.4 Failing to Send TCP/UDP Data

If the module sets up a non-transparent TCP/UDP connection but fails to send TCP/UDP data, troubleshoot in the following procedures:

- Step 1:** Try to send data to another server to confirm whether the server is functioning properly.
- Step 2:** Contact your operator to check whether the SIM card is out of credibility
- Step 3:** Check whether the connection is closed since no data is sent in long time.

Shorten the heartbeat period or increase the data sending frequency. If data is sent successfully then, the issue is caused by closed connection.

- Step 4:** Send **AT+IPSTATUS=<SocketID>** check the status of the TCP/UDP connection.

- If the UART port returns **CONNECT**, a TCP/UDP connection is set up.
- If the UART port returns **DISCONNECT**, no TCP/UDP connection is set up.

- Step 5:** Check whether the host has sent data to the module.

It is recommended to reserve the UART log saving and printing function for the host.

Output UART logs and check the AT commands and data exchanged between the module and the host in the log.

If the UART port returns **+TCPSEND/+UDPSEND: <SocketID>,<DataLength>** to **AT+TCPSEND/AT+UDPSEND**, the host sends data to the module successfully.

- Step 6:** Send **AT+TCPACK=<SocketID>** to check whether the module receives ACK of TCP protocol from the server.
- If the values of **<data_sent>** and **<acked_rcv>** are same, data is sent to the server and the server acknowledges.

- If the value of **<data_sent>** is larger than that of **<acked_rcv>**, the data is not sent to the server yet or the server does not receive the data because of a delay.

To eliminate this issue, contact Neoway to capture TCP data log for analysis.

3.5 Failing to Receive TCP/UDP Data

If the module sets up a non-transparent TCP/UDP connection but fails to receive TCP/UDP data, check the following items:

Step 1: Try to receive data from another server to confirm whether the server is functioning properly.

Step 2: Contact your operator to check whether the SIM card is out of credibility

Step 3: Send **AT+IPSTATUS=<SocketID>** check the status of the TCP/UDP connection.

- If the UART port returns **CONNECT**, a TCP/UDP connection is set up.
- If the UART port returns **DISCONNECT**, no TCP/UDP connection is set up.

Step 4: Check whether the connection is closed since no data is sent in a long time.

Shorten the heartbeat period or increase the data sending frequency. If data is received successfully then, the issue is caused by closed connection.

If data is still not received, contact Neoway to capture TCP data log for analysis.