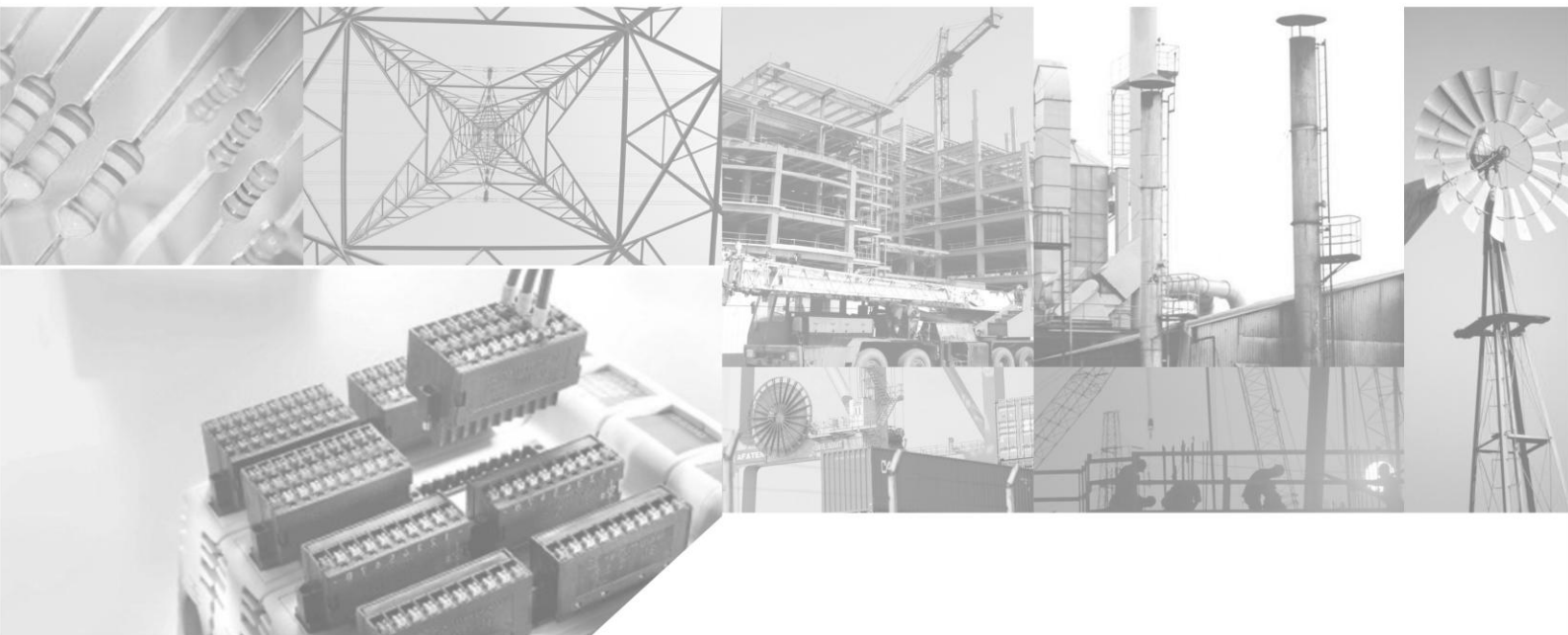


Neoway M660 CMUX User Guide

V1.4



Copyright © 2008 Neoway Technology

All rights reserved.



is a trade mark of Neoway Technology Co. Ltd.

Notice

This document is intended for the customer engineers.

This document is subject to change without any notice.

No responsibility is assumed by Neoway Technology for the use of this document.

Neoway Technology always provides the best supports.

For pricing, delivery, and ordering information please contact:

Sales@neoway.com.cn

For any technical support requests please contact:

Support@neowat.com.cn

For further information please visit:

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

Contents

1. Introduction	5
1.1 Audience	5
1.2 Related Documents.....	5
2. What's serial port MUX?	5
3. Software architecture of MUX	6
4. Multiplexing mechanism.....	6
5. CMUX frame structures	7
6. About control frame	7
7. Reference design.....	8
8. Procedure of CMUX mode data transfer	9
9. CMUX tool introduction.....	10

Revision history		
Revision	Content	Effective Date
V1.0	Initial	201109
V1.1	modified	201204
V1.2	modified	201205
V1.3	modified	201208
V1.4	modified	201209

NEOWAY CONFIDENTIAL

1. Introduction

This document tends to provide the reader with detailed information about developing a Multiplexing Protocol running on his equipment in order to use CMUX of Neoway modules, including the description of Neoway Multiplexing Protocol, using guide of Neoway Serial Port MUX tool.

1.1 Audience

This document is intended for developers and testing engineers to learn about Multiplexing Protocol and how to use it.

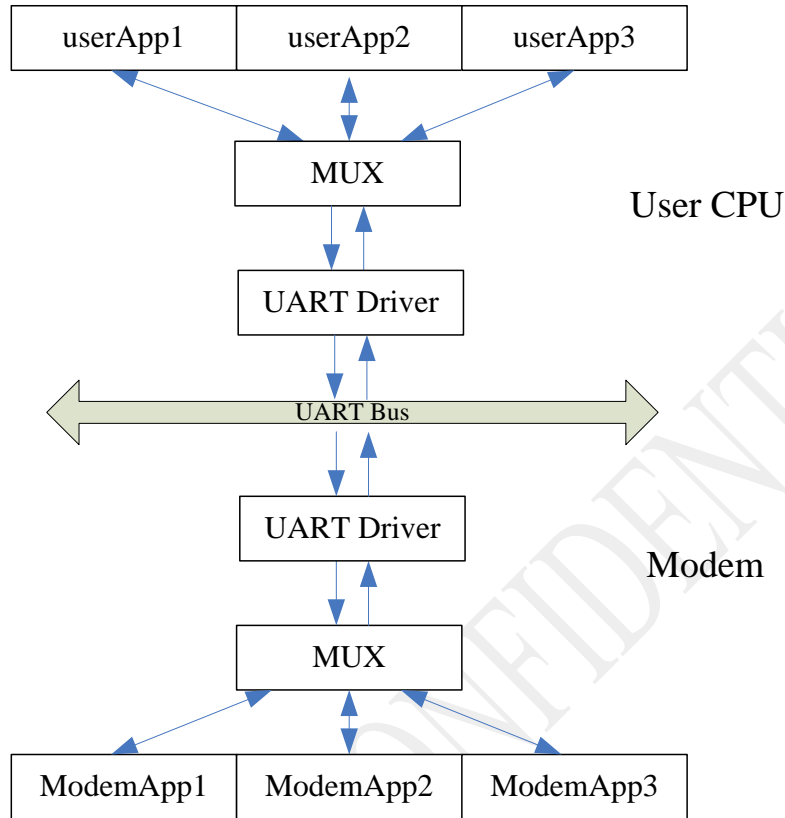
1.2 Related Documents

[1] GSM 07.10 Version 7.1.0 Release 1998

2. What's serial port MUX?

The multiplexer can provide several virtual connections between TE and MS, the multiplexing method is to add a header to every package to identify which multiplex channel this package belongs to.

3. Software architecture of MUX



Picture1 Multiplexing software frame

From the above picture, we can see that a MUX layer is added to the MUX application in comparison with traditional serial port application. The MUX layer of user CPU must be compatible with Neoway module MUX layer. Each user app should deliver the data to be sent to MUX layer, the MUX player will package the data and send them to UART driver.

Generally, the multiplexer can provide a virtual data channel and a virtual AT commands channel in module applications. Therefore, when module is on GPRS data mode via the data channel, AT commands can be used at the same time via AT commands channel. And that can avoid switching from data mode to command mode back and forth.

4. Multiplexing mechanism

Four steps are included in multiplexing, parameter negotiation, DLC establishment, data transmission, DLC releasing.

DLC0 is multiplexer Control Channel, only control messages are conveyed on this channel. The two MUX entities (Multiplexer of the user side and Multiplexer of of the module side) use DLC0 to

negotiate and set parameters, establish other virtual channels and report status.

To enter MUX mode, AT+CMUX=0 is entered, after that DLC0 can be established. Other channels can be established then. Three Data Control channels can be established. DLC0 is for control channel, DLC1 is for data channel, DLC2 is for AT command channel.

SAMB frame is used to establish DLC. Refer to section 5.4.1 Establishment of GSM7.10 for detailed establish information.

5. CMUX frame structures

All the messages transferred between the module and the user application is in the frame format that defined in the following structure.

1 octet	1 octet			1 octet			1 – 2 octets		0-32768 octets	1 octet	1 octet	
START	ADDRESS			CONTROL			LENGTH		INFORMATION	FCS	STOP	
0xF9	6 bits	1 bit	1 bit	3 bits	1 bit	4 bits	7-15 octets	1 bit				0xF9
							Length value	(*)				
	DLCI (0 - 63)	C/R	EA=1		P/F			EA				

Picture2 MUX frame structure

Both start and stop octet is defined as 0xF9 in hexadecimal format. There are five fields, ADDRESS, CONTROL, LENGTH, INFORMATION, FCS. The important fields are explained as below,

- DLCI in ADDRESS: the id of this frame, it indicates which virtual channel this frame belongs to.
- CONTROL field: indicates the type of this frame. Values are as below,

Frame Type	1	2	3	4	5	6	7	8	Notes
SABM (Set Asynchronous Balanced Mode)	1	1	1	1	P/F	1	0	0	
UA (Unnumbered Acknowledgement)	1	1	0	0	P/F	1	1	0	
DM (Disconnected Mode)	1	1	1	1	P/F	0	0	0	
DISC (Disconnect)	1	1	0	0	P/F	0	1	0	
UIH (Unnumbered Information with Header check)	1	1	1	1	P/F	1	1	1	
UI (Unnumbered Information)	1	1	0	0	P/F	0	0	0	Optional

Among the above types, SABM frame is used to establish DLC channel, UA frame is used to respond, DISC frame is used to close DLC, UIH frame is used for data transmission.

- FCS field: FCS only calculate address, control, length checksum for UIH frame, but calculate the whole frame (including INFORMATION field) checksum for UI frame. This algorithm is implemented in the code example.

6. About control frame

Data frames(UIH frame) transferred via DLC0 are all interpreted as control frame. The data field of control frame includes all kinds of monitoring message. Refer to section 5.4.6 Multiplexer Control

Channel in GSM7.10 for the specified format.

In addition, establish DLC0, DLC1, DLC2 is the main work of MUX layer software, including interaction of SABM, UA, DISC and MSC frames.

7. Reference design

Only after the establishing of 3 data channel and 1 control channel, can you send AT commands.

When sending AT commands or other data, please note that the convergence layer of Neoway M660 is type2, the control byte is v.24, one more byte needs to be added before information, or the interpretation will be incorrect, this byte should also be judged when receiving.

M660 Multiplexing frame is based on GSM 0710-720 standard SPEC, it uses Basic option frame(refer to GSM07.10 chapter 5.2), for UIH frame, there's information field that stands for the real data to be transmitted. Convergence layer is put before information according to chapter 5.5

Example: F9 0F EF 09 **01** 41 54 0D 7C F9

Frame	Meaning
F9	Flag
0F	Address
EF	UIH
09	Length
01 41 54 0D	Information
01	Convergence layer
41 54 0D	AT command : AT
7C	FCS
F9	Flag

The fifth byte, convergence layer type2, or v.24, usually use 01 or 05, or 85

Messages of Multiplexing data channel: CPU sends ATD*99# to module at first

Module replies as below

- f9 05 ef 19 **0d** 0d 0a 43 4f 4e 4e 45 43 54 0d 0a 44 f9

Module outputs, when AT commands, the output is 0d

- f9 05 ef 03 **8d** b5 f9

Module outputs 8d, which indicates data connection is OK.

- f9 05 ef 59 **8d** 7e ff 7d 23 c0 21 7d 21 7d ...

Module outputs valid network data.

- f9 07 ef xx **8d** xx xx xx xx ...

Module uart receiving, CPU dialup connection responds.

8. Procedure of CMUX mode data transfer

1) Enter CMUX mode

```
>at+cmux=0,0,,1509,254,5,255
<at+cmux=0,0,,1509,254,5,255
<OK
```

Note: Belowing output will be in hexadecimal format.

2) Establish control channel DLC0

```
>F9 03 3F 01 1C F9
<F9 03 73 01 D7 F9
```

3) Establish data channel DLC1,DLC2,DLC3

+ Establish DLC1

```
>F9 07 3F 01 DE F9
<F9 07 73 01 15 F9
```

>**F9 01 EF 0B E1 07 07 0C 01 79 F9** (data sent to module, should be sent in time, it's better to send this data as soon as receiving UA frame)

+ Establish DLC2

```
>F9 0B 3F 01 59 F9
<F9 0B 73 01 92 F9
```

>**F9 01 EF 0B E1 07 0B 0C 01 79 F9**(data sent to module, should be sent in time, it's better to send this data as soon as receiving UA frame)

+ Establish DLC3

```
>F9 0F 3F 01 9B F9
<F9 0F 73 01 50 F9
<F9 09 EF 1F 0D 0D 0A 2B 45 49 4E 44 3A 20 31 32 38 0D 0A 27 F9
```

>**F9 01 EF 0B E1 07 0F 0C 01 79 F9**(data sent to module, should be sent in time, it's better to send this data as soon as receiving UA frame)

4) Data transfer

+ Send "AT\r" via DLC1

```
>F9 07 FF 09 01 41 54 0D 2C F9
<F9 05 EF 05 0D 41 51 F9
<F9 05 EF 05 0D 54 51 F9
<F9 05 EF 05 0D 0D 51 F9
<F9 05 EF 0F 0D 0D 0A 4F 4B 0D 0A BC F9
```

+ Send "AT\r" via DLC2

```
>F9 0B EF 09 01 41 54 0D BE F9
<F9 09 EF 05 0D 41 D6 F9
```

<F9 09 EF 05 0D 54 **D6** F9
 <F9 09 EF 05 0D 0D **D6** F9
 <F9 09 EF 0F 0D 0D 0A 4F 4B 0D 0A **3B** F9



Send “AT\r” via DLC2

>F9 0F EF 09 **01 41 54 0D 7C** F9
 <F9 0D EF 05 0D 41 **14** F9
 <F9 0D EF 05 0D 54 **14** F9
 <F9 0D EF 05 0D 0D **14** F9
 <F9 0D EF 0F 0D 0D 0A 4F 4B 0D 0A **F9** F9

5) **Release the channels and exit MUX mode**

Function	Sending (CPU->M660)	Receiving(M660->CPU)
Release DLC1	F9 07 53 01 3F F9	F9 07 73 01 15 F9
Release DLC2	F9 0B 53 01 B8 F9	F9 0B 73 01 92 F9
Release DLC3	F9 0F 53 01 7A F9	F9 0F 73 01 50 F9
Release DLC0 If DLC0 is released successfully, then MUX mode is exited .	F9 03 53 01 FD F9	F9 03 73 01 D7 F9 ✧ <F9 07 73 01 15 F9 (If DLC1 is establish, then release DLC1 before releasing DLC0, the reply is UA acknowledge frame of releasing DLC1) ✧ <F9 0B 73 01 92 F9 (If DLC2 is establish, then release DLC2 before releasing DLC0, the reply is UA acknowledge frame of releasing DLC1). ✧ < F9 0F 73 01 50 F9 (If DLC3 is establish, then release DLC3 before releasing DLC0, the reply is UA acknowledge frame of releasing DLC1).

9. CMUX tool introduction

Neoway has developed a CMUX tool on PC(windows OS), the operation steps are diagrammatized as below ,

Open Neo_CMUX.exe at directory /Release. And config as following,

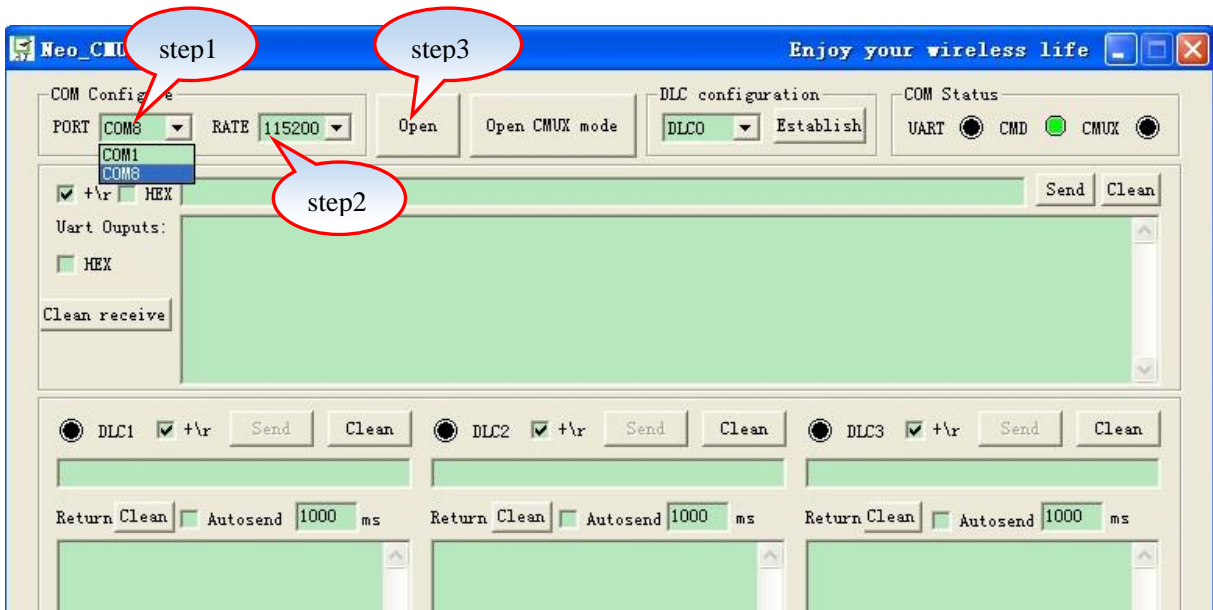
Step1: choose **PORT**

Step2: choose **RATE**(115200)

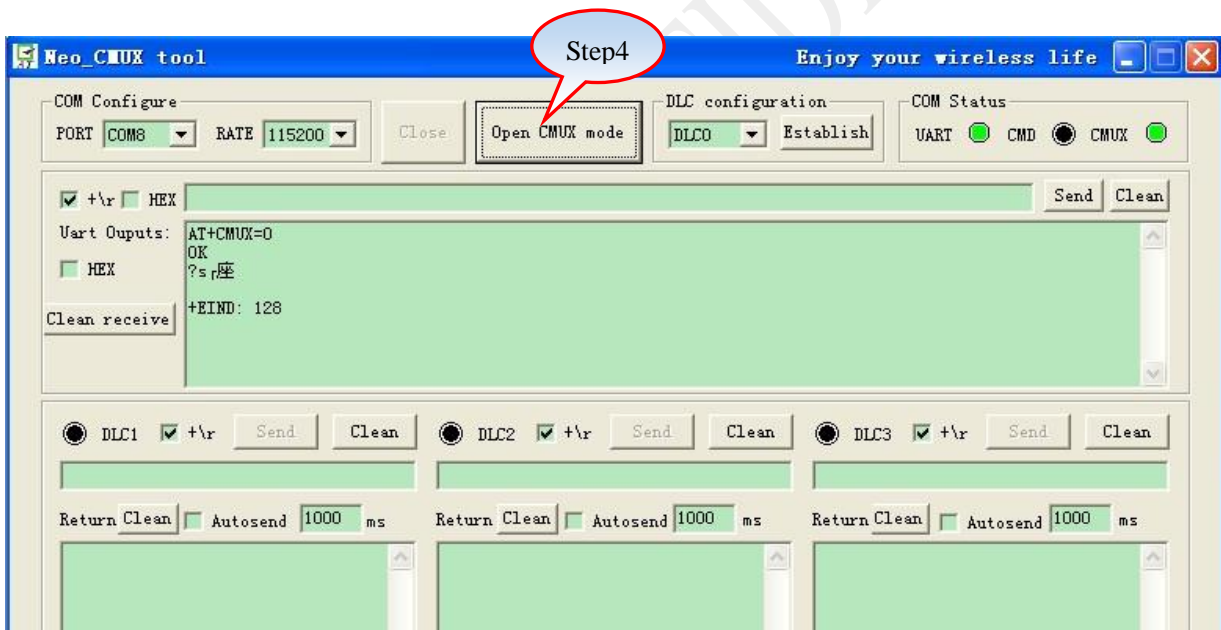
Step3: click **Open**, then the com port is opened

Step4: **Open CMUX mode**

Step5: establish virtual channels, click **Establish**, DLC0 will be created, DLC0 is the control channel, then DLC1, DLC2, DLC3. If all the channels are created successfully, you can see green icon status indication of DLC as shown in picture4. And at commands can be entered via virtual channels DLC1~DLC3



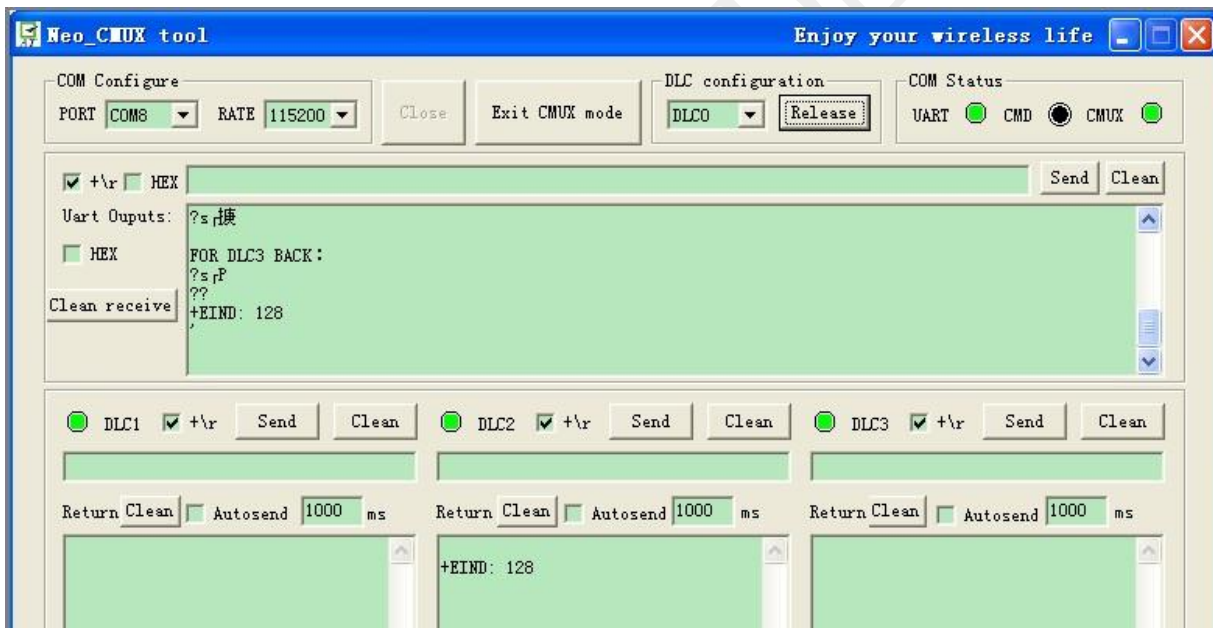
Pic1



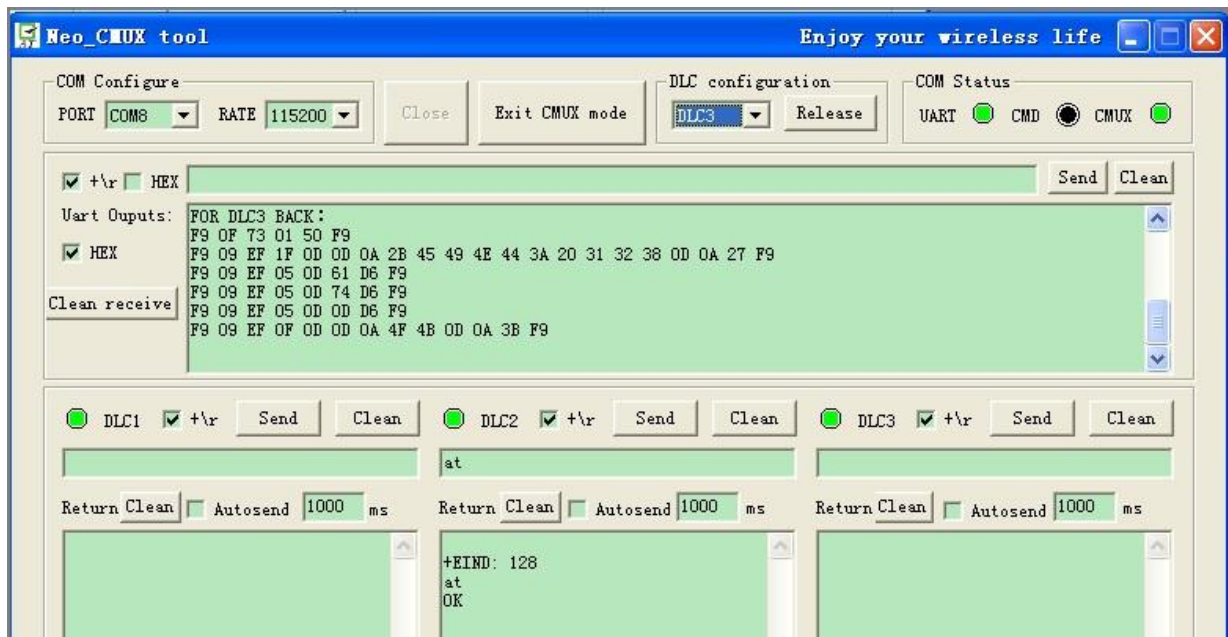
Pic2



Pic3



Pic4



Pic5