CMUX protocol learning summary

CMUX protocol learning summary

> When I first started contacting CMUX at the beginning of five months ago, I searched a lot of materials and documents on the Internet. At that time, I thought that I understood the protocol quite well, but now that I have forgotten what I learned before, I take this opportunity to review it. . But the core of CMUX protocol is to learn its frame structure. This article refers to the following documents:
>
> - **CMUX-User-Guide**
> - **CMUX protocol document**
> - **Huawei Serial Multiplexing Reference Manual**
>
> Stop talking nonsense, let's look at CMUX.

---

1. What is CMUX

CMUX refers to serial port multiplexing. The multiplexer mode of the serial port is to enable a serial interface to transmit data to four different client applications.
In actual applications, a physical serial port can only transmit one upper-layer application data stream within a certain period of time. What if there are multiple data streams to be sent at the same time? Except accessing multiple UARTs Is there another way?
The function of the CMUX protocol is to use a physical serial port at the bottom to provide multiple logical serial ports to the upper system, and each logical serial port corresponds to a data link connection (DLC). In this way, multiple sessions can exist simultaneously on a serial interface, such as voice, FAX, data, SMS, GPRS, USSD, etc. This is particularly advantageous when a fax/data/GPRS call is in progress, for example the control module or the use of SMS services can be done through additional channels without disturbing the data flow; no need to access the second UART.

The multiplexing protocol provides the ability to virtualize multiple parallel logical communication channels on a single physical communication channel. It is generally used between TE (Terminal Equipment) and MS (Mobile Station). In the meantime, TE is equivalent to the AP side of a smart phone, and MS is equivalent to the MODEM side of a smart phone. The following figure shows a typical protocol hierarchy:

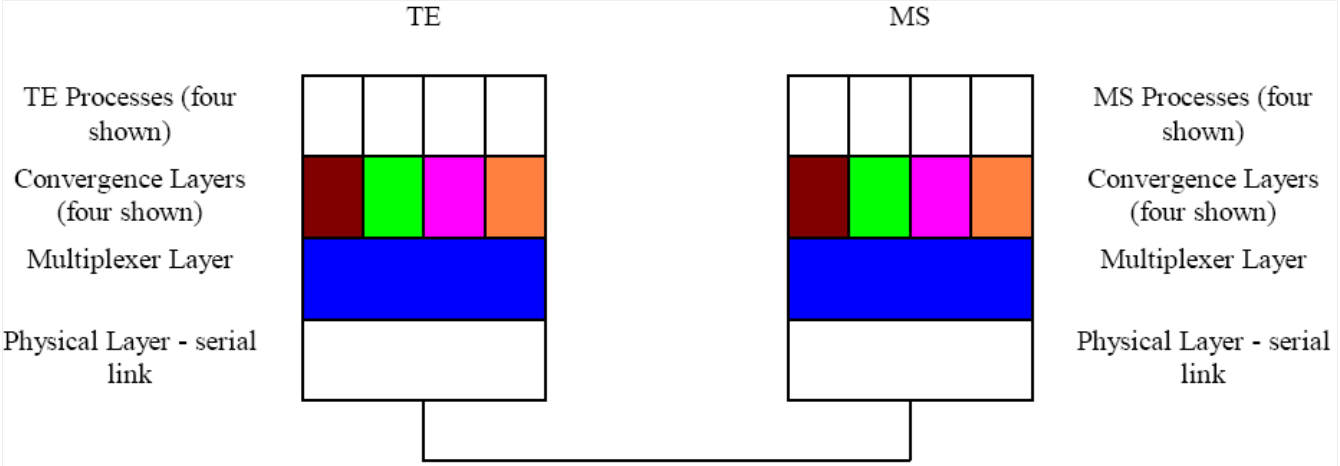


Figure 1: Protocol Stacks

The blue part is the MUX multiplexing layer, which uses the physical serial port link at the bottom to send and receive data, and at the same time provides several logically independent transceiver channels to the upper layer (four logical channels are provided in the figure above, indicated by different colors). Each logical channel is created independently and can have software flow control. In actual use, the MUX on the TE side initiates a channel establishment request to the MUX on the MS side, setting channel parameters, etc., which is the active party; the MUX on the MS side waits for the service request from the TE side and provides corresponding services according to its own capabilities. In other words, the roles of the two are asymmetrical.

Each channel between the TE terminal and the MS terminal is called a DLC link, and they are established independently of each other. The data transmission adopts the 8BIT character Start-Stop method for transmission, and the data interaction between the two parties uses the frame structure defined later.

---

2. Start CMUX mode

Usually send AT+CMUX command to GPRS module to activate multiplexing, the command format is:

```
AT+CMUX=<mode>[,<subset>[,<port_speed>[,<N1>[,<T1>[,<N2> [,<T2>[,<T3>[,<k>]]]]]]]]
```

Parameter Description:

```
mode: Specify the multiplexing mode, the GTM900 module must be set to1。
0 Basic option
1 Advanced option
```

```
subset: Specify frame type, set to GTM900 module0。
0 UIH frames used only
1 UI frames used only
2 I frames used only
 Default value:0
```

```
port_speed: Specify the transmission rate

1 9 600 bit/s
2 19 200 bit/s
3 38 400 bit/s
4 57 600 bit/s
5 115 200 bit/s
6 230 400 bits/s
```

```
N1: Maximum frame size

1- 32768
 Default value: basic mode is 31 (advanced mode is 64)
```

```
T1: Confirmation time, the unit is 10ms.

1-255
 Default value: 10 (ie 100 ms)
```

```
N2: Maximum number of retransmissions

0-100
 Default value: 3
```

```
T2: The response time of the control channel, the unit is 10ms


2-255
 Default value: 30 (ie 300 ms)
 Note: T2 must be greater than T1.
```

```
T3: wake-up response time, in s

1-255
 Default value: 10
```

```
k: Serial port size, used for advanced mode with error recovery.
1-7
 Default value: 2
```

3. CMUX frame structure

There are three operation modes for multiplexing: basic mode, advanced mode with error recovery function, and advanced mode without error recovery function (see GSM 07.10 protocol for details). Different GPRS modules also support different modes.

mark	address	control	length	information	check	mark
1 byte	1 byte	1 byte	1 or 2 bytes	Integer bytes	1 byte	1 byte
0xF9 or 0x7e	DLCI value	Frame type		I frame, UI frame, UIH frame	CRC	0xF9 or 0x7E

Among them, the frame type can be divided into***Control frame***with***Information frame***。
CMUX passed by both parties***Control frame***Negotiation is used to construct and tear down virtual links, and the control frames are as follows:
SABM: Create DLC
UA: Response to SABM frame or DISC frame
DM: When the link is not successfully established, the response to the received DISC command
DISC: Notify the peer to tear down the link and send a DISC frame on DLC0, which is equivalent to exiting the MUX function

Information frame:

UIH\UI\I: These three are information frames, which are frames that carry data to be transmitted.

Each bit in each byte such as tag, address, control, length, etc. represents different information. For details, please refer to the Huawei serial multiplexing manual, which is used for each bit of each Byte in the CMUX frame structure. Detailed introduction.

4. CMUX practical application

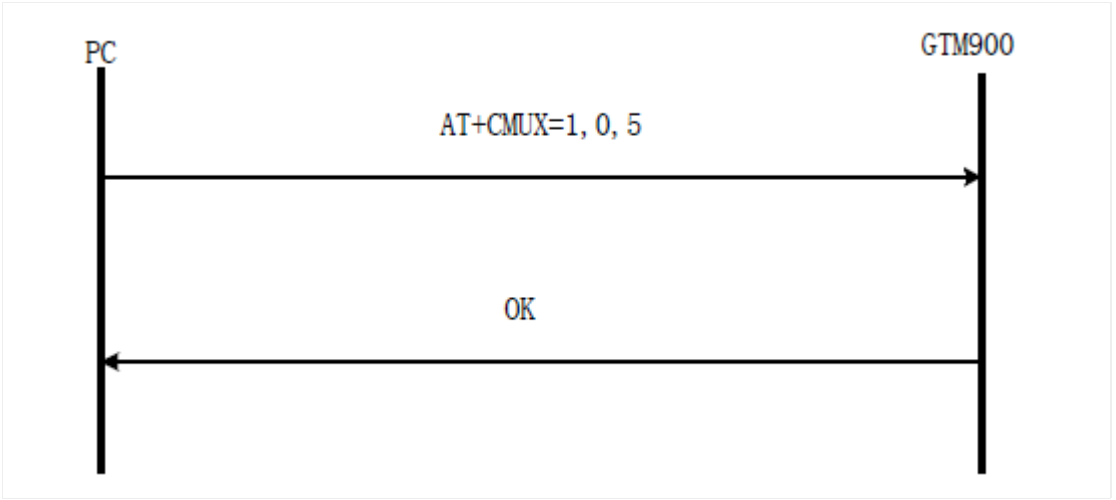
Frame structure of basic mode
The frame structure of the basic mode is as follows:

标记	地址	控制	长度	信息	校验	标记
1字节	1字节	1字节	1或2字节	整数个字节	1字节	1字节

Frame structure in advanced mode
The frame structure of advanced mode is as follows:

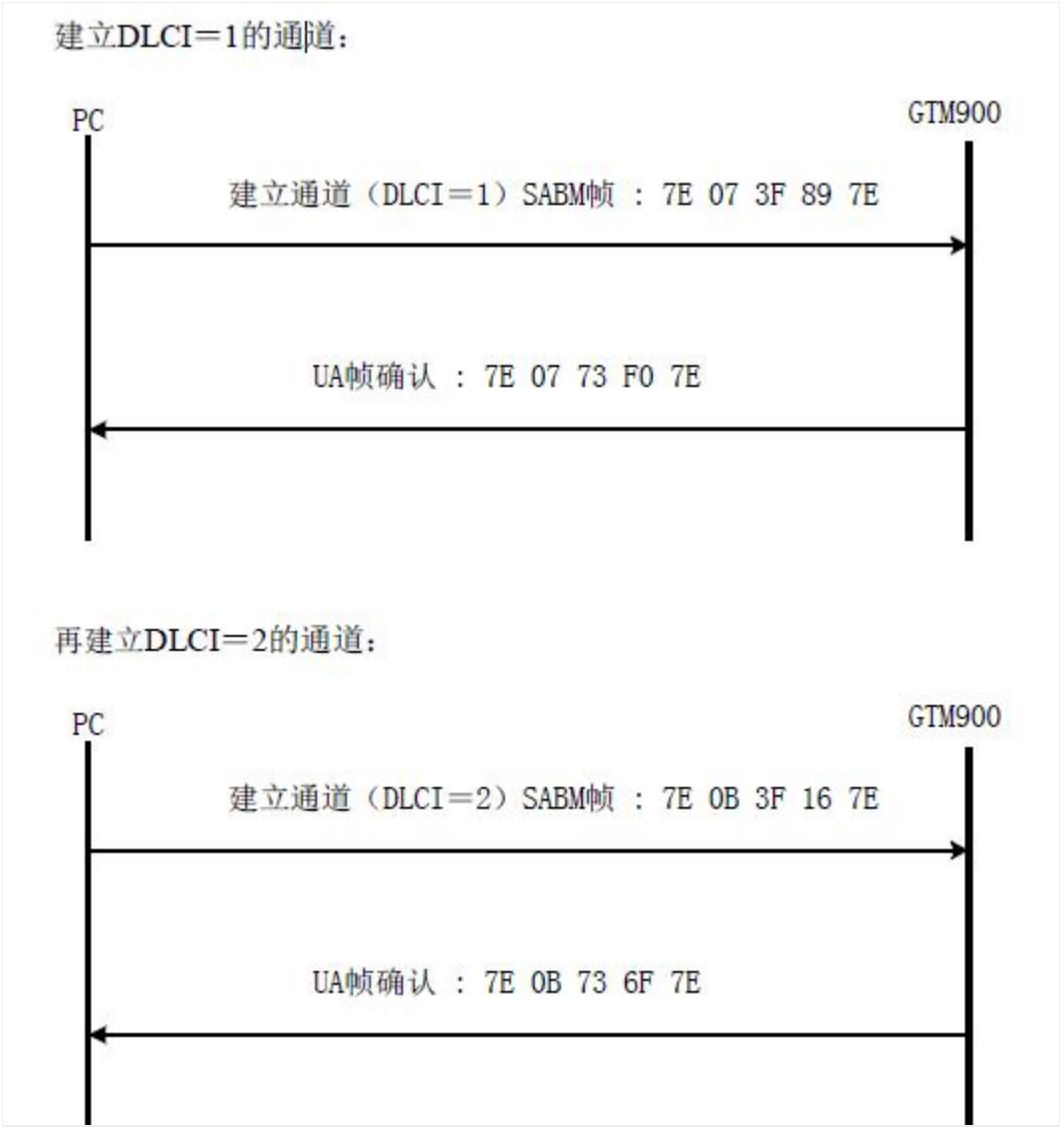
标记	地址	控制	信息	校验	标记
1字节	1字节	1字节	整数个字节	1字节	1字节

Note: The advanced mode has no length field.
①Start business



②Establish a control channel
After starting multiplexing, then a control channel (DLCI=0) needs to be established to control the multiplexing parameters.

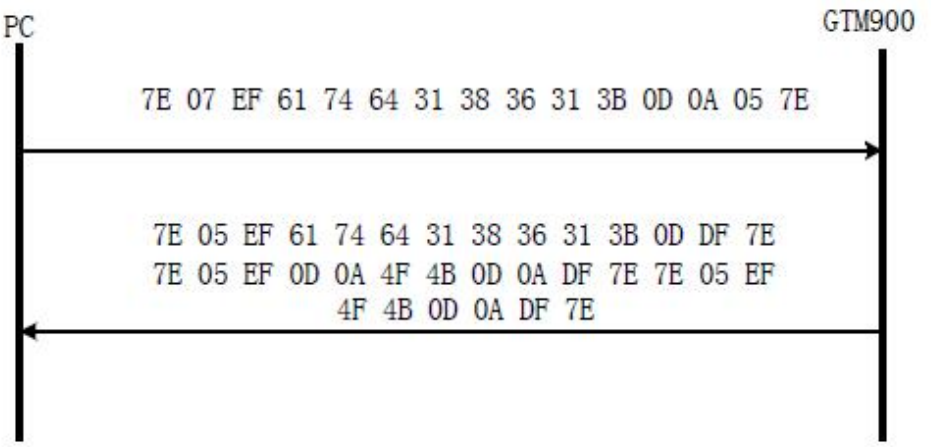
③Establish other channels



④Transfer data

After the logical channel is established, UIH can be used for data transmission. The following describes the data transmission method, using the channel with DLCI=1 to transmit data as an example, other channels are similar.

For example, to transmit AT commands, directly pack the string "atd1861;\r\n" into the information field of the UIH frame, as shown below:



注意：FCS计算只包括地址域和控制域。

传输数据信息时，在PC—》GTM900方向，C/R位置为1；在GTM900—》PC方向，C/R位置为0；两个方向P位都位0。

PC—》GTM900：信息域的字符串“atd1861；\r\n”对应为：61 74 64 31 38 36 31 3B 0D 0A

GTM900—》PC：

GTM900给PC回的信息为：

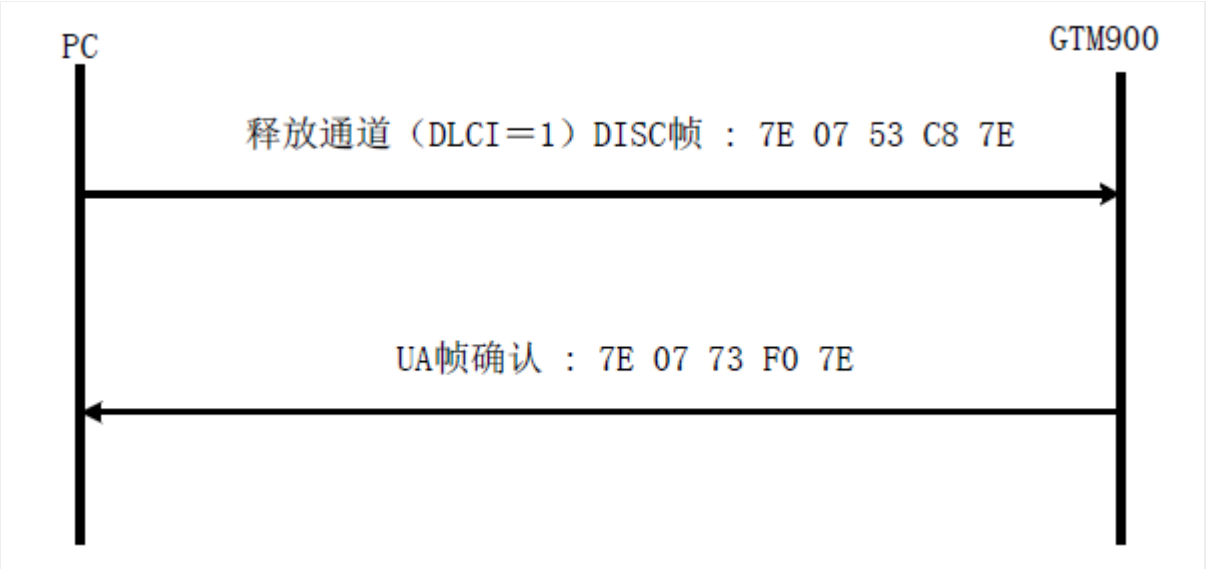
atd1861;

OK

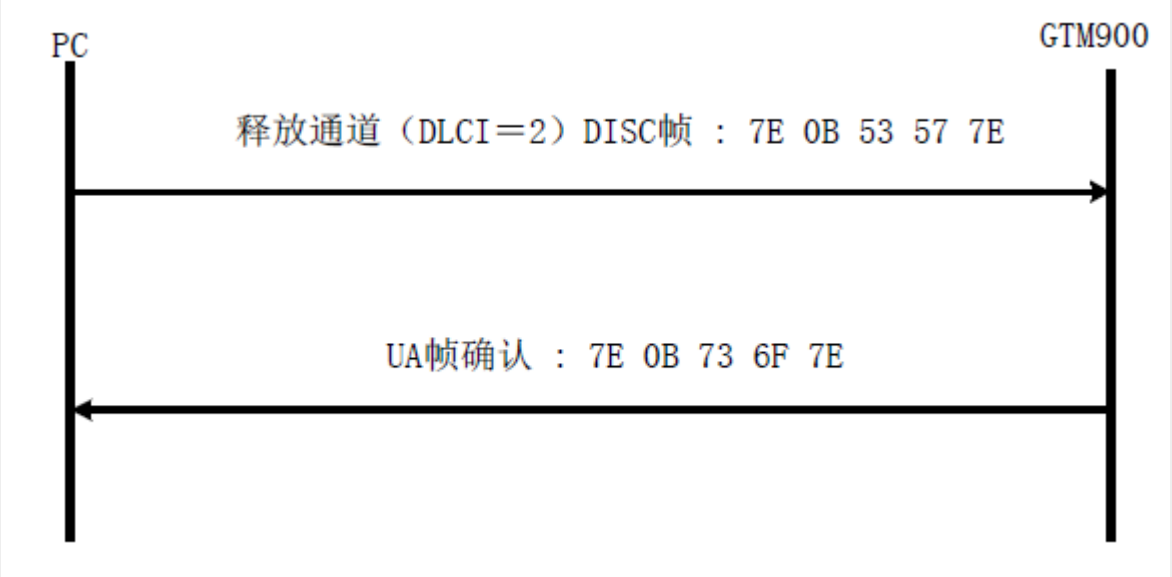
OK

⑤ Release business

Release the channel with DLCI=1:



Release the channel with DLCI=2:



abnormal situation:

⑥Close business

Releasing the channel with DLCI=0 is equivalent to closing the multiplex service and returning to the AT command mode under normal conditions.

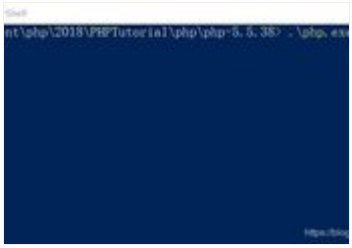
Intelligent Recommendation

Communication protocol DHCP protocol learning summary

1. What does the DHCP protocol mean? DHCP (Dynamic Host Configuration Protocol, dynamic host configuration protocol), formerly BOOTP protocol, is a LAN network protocol, using UDP protocol to work, un...

Java learning summary: HTTP protocol

This article refers to this article：http://blog.51cto.com /13570193/2108347； HyperText Transfer Protocol (HTTP) is the most widely used network protocol on the Internet. All www documents must comply w...



PHP pseudo-protocol learning summary

php:// PHP Provides some miscellaneous input/output (IO) streams, allowing access to PHP's input and output streams, standard input and output and error descriptors, in-memory, temporary file streams ...

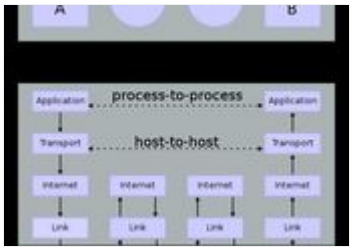
Ftp network protocol learning summary

FTP full English name is File Transfer Protocol In PHP4/5 resource ftp_connect (string $host [, int $port [, int $timeout]]) $host is the server to connect to. #Do you only allow XXX.XXX.X.XX as a ...

Computer network protocol learning summary

Computer network protocol layering Currently, there are two sets of protocols for network communication standards, namely the OSI reference model and the TCP/IP model. The OSI reference model is only ...

More Recommendation

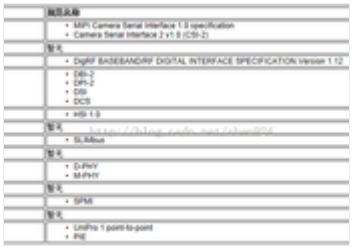


TCP, UDP protocol learning summary

TCP, UDP protocol learning summary TCP / IP protocol suite. Internet protocol suite (English: Internet Protocol Suite, abbreviated IPS) is a network communication model, as well as a whole network tra...

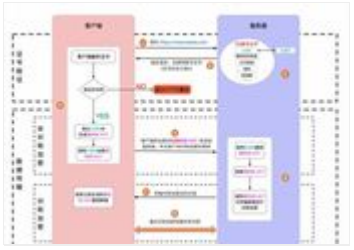
Learning Summary-Vlan Relay Protocol

VTP VTP (VLAN Trunking protocol) is a message protocol used to synchronize VLAN information in the VTP domain without having to configure the same VLAN information on each switch, thereby achieving co...



MIPI protocol learning summary (1)

1. Introduction to MIPI: MIPI (Mobile Industry Processor Interface) isMobile Industry Processor Interfaceabbreviation of. MIPI is an open standard for mobile application processors initiated by the ...



HTTPS protocol personal learning summary

HTTPS What is HTTPS The difference between http and https https features SSL and TSL Symmetric and asymmetric encryption certificate What is CA HTTPS general process What is HTTPS HTTPS refers to Hype...

Summary of basic learning of HTTP protocol

start: August 2, 2020 I. Introduction: Regarding the concept of agreement, I remembered something I did with my friends when I was a kid-for the code, we booked the first and second sentences of the c...

Related Posts

- TCP protocol learning summary (on)
- UDP protocol learning summary
- IP protocol learning summary
- http protocol learning summary
- http protocol learning summary
- WebSocket protocol learning summary
- Http protocol learning summary
- TCP protocol learning summary
- HLS protocol learning summary
- HTTP2.0 protocol learning summary

Popular Posts

- Spring AOP oriented aspect programming
- Digg BT_ How to use Digg API to easily display DIGG posts through XSLT
- Jdk1.8 List deduplication
- Linux Enterprise Interview Questions
- CodeForces Round # 661 (Div. 3) Solution Report
- LOJ2425 NOIP2015 transport plan [+ LCA + binary tree differential] *
- China Service Cloud participated in the 2018th International Internet of Things Expo, empowering equipment on the cloud
- (The JAVA) fetch other data returned by the server
- Refactoring - Optimize your own library in a variety of ways
- September 19 Microservice Architecture Wednesday

Recommended Posts

- Idea common settings

TOP

- Leetcode+combinations DFS classic retrospective
- 123. Best Time to Buy and Sell Stock III
- Module mechanism in Node
- CodeForces - 500A New Year Transportation
- TortoiseGit pushes code to Alibaba Cloud git hosting or gitlab repository error error: failed to push some refs to 'git@code.aliyun.com_XXX.git'
- C#WebForm realizes addition, deletion and modification
- [C] My solution on how to find the 62 power of X
- Cocos2d-js cache
- .NET WebApi combat Lecture

Related Tags

UDP protocol learning summary

WebSocket

websocket

ajax

Network protocol

hls

Multimedia transmission

http server

the Internet

cdn

Advertising Sponsor Link

- I'm Feeling Lucky
- Hard work, too tired, take a break
- Contact us to advertise