FIBOCOM
PERFECT WIRELESS EXPERIENCE

MTC

Application Guide_Sleep and Wakeup

V1.3

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

Trademark Statement

FIBOCOM 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

Applicability Model

No.	Applicability Model	Description
1	NL668 Series	NA
2	MC116 Series	NA
3	MC116 Series	NA
4	MA510 Series	NA
5	L716 Series	NA
6	LE Series	LE170 series, LE270 series, LE130 series, LE230 series, LE370 series
7	MC66X Series	NA
8	MC61X Series	NA
9	L610 Series	NA
10	Lx61X Series	NA
11	MG66X Series	NA
12	FG132 Series	NA
13	LQ2A0 Series	NA

Change History

V1.3（2025-03-24）	Add LQ2A0 serials
V1.2（2024-05-17）	Modify FG132 USB SUSPEND and debug chapter description
V1.1（2023-12-28）	Add FG132 serials
V1.0（2023-11-30）	Initial version

1 Foreword

The overall power consumption of the system will increase after the Fibocom module is embedded to the host system. Therefore, Fibocom provides several power management solutions for Fibocom modules, so that the host can lower the overall power consumption through the management module. The following introduces the sleep and wake-up mechanisms and use guide of Fibocom module from the user's point of view.

2 Term Definition

Term	Description
WAKEUP_IN/DTR	Pin used to control the sleep and wakeup of UART
RI/WAKEUP_OUT	Pin used to wake up the host
USB VBUS	Pin used to control the sleep and wakeup of USB
USB SUSPEND/RESUME	Pin used to control the sleep and wakeup of USB bus. When the host and the module do not exchange data, a Full Speed J signal, that is, the SUSPEND signal, will be sent. If there is data exchange, a Full Speed K signal, that is, the RESUME signal, will be sent.
EINT	GPIO that supports interrupts

3 Overview

Under the premise of enabling the sleep function, the Fibocom series modules will try to fall asleep when the system is idle. The modules that can be controlled by the HOST to affect the system to enter the idle state are UART and USB, and the system will try to fall asleep only when both UART and USB are idle.

Based on the application scenario of the module on the host computer, it is concluded that the connection mode between the module and the host is mainly UART and USB. The following figure shows the connection mode. In this connection, MODULES refer to Fibocom modules, and HOST refers to host computer.

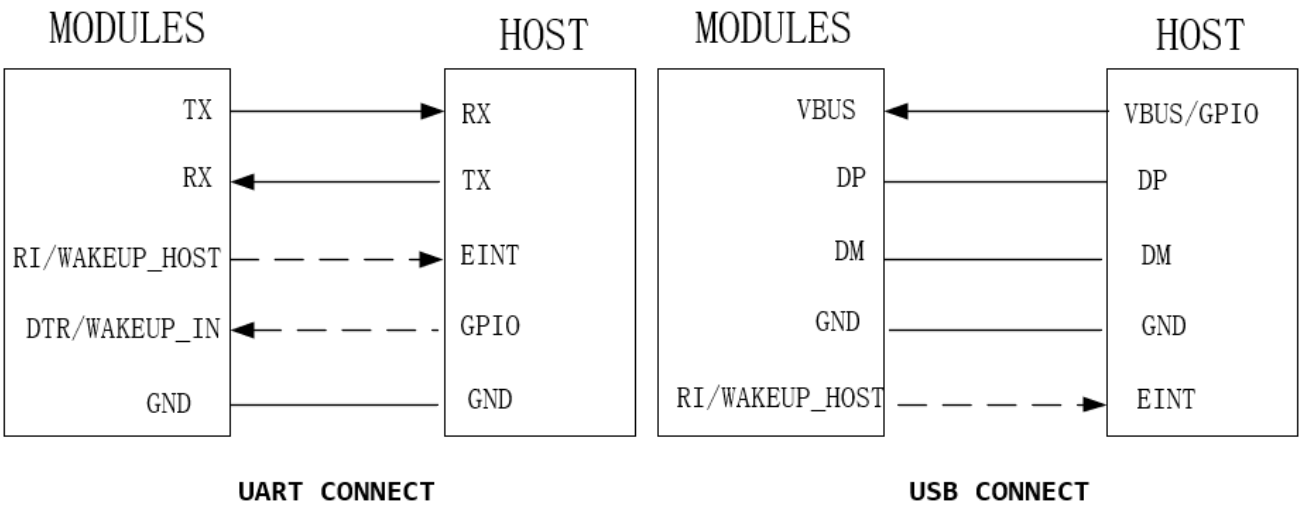

Figure 1. Connection between FIBO MODULES and HOST

For the sleep configuration based on UART connection mode, see section 4.1. For the sleep configuration based on USB connection mode, see section 4.2.

The following table describes the sleep and wakeup solutions supported by Fibocom modules.

Table 1. Overview of sleep mode

Mode	Module	How to Enter Idle State	Description
Sleep	UART	WAKEUP_IN/DTR	When the level of the GPIO pin changes, the UART enters the idle state.
		ATS24	When the ATS24 is being configured and there is no data in the UART, the UART enters the idle state.
	USB	USB VBUS	When the USB VBUS pin is pulled down, the USB enters the idle state.
		USB SUSPEND	When the USB SUSPEND signal is sent, the USB enters the idle state.

Table 2. Wakeup source overview

Wakeup Source	Wakeup Mode		Description
HOST wakeup module	UART	WAKEUP_IN/DTR	The module will wake up when the GPIO pin level changes.
		ATS24	The module will wake up when the UART has data.
	USB	USB VBUS	The module will wake up when the USB VBUS pin is pulled up.
		USB RESUME	The module will wake up when the USB RESUME signal is sent.
Data wakeup module	SMS		The module will automatically wake up when an SMS message is received.
	Tel.		The module will automatically wake up when a call is received.
	Network data		The module will automatically wake up when network data is received.

- No matter how the UART/USB is idle, it needs to be woken up in the corresponding way. For example, if the UART is idle through the GPIO, you need to wake it up through GPIO.
- For details about the module wakeup by HOST, see section 5.1.

Table 3. HOST wakeup by module

Wakeup Mode	Remarks
HOST wakeup by RI/WAKEUP_OUT	When the module receives a call, an SMS message, or TCP/IP data, the RI/WAKEUP_OUT pin outputs a pulse waveform to wake up the HOST.
Remote wakeup by USB	When the USB of the module receives the network data and the URC reports the event, it will initiate a remote wake-up signal to wake up the HOST.

- LE series modules do not support remote wakeup by USB.
- For the HOST wakeup by RI, only a built-in dial-up is implemented. When the TCP/IP data is generated, the RI/WAKEUP_OUT pin outputs the pulse waveform.
- For details about the HOST wakeup, see section 5.2

4 Sleep Solution

The following introduces how the host computer should control the module to sleep based on the UART connection and USB connection.

4.1 Sleep Configuration Based on UART Connection

For the sleep mode based on the UART connection, the system enters the low-power mode as long as the UART enters the idle state.

The system can control the UART to enter the idle state through the GPIO mode or automatic mode, and GPIO mode is preferred, as shown in the following table:

Table 4. Sleep configuration based on the UART connection

Sleep Mode	Configuration Command	Description
GPIO mode	AT+GTLPMMODE=1,1 AT+CSCLK=1 Or AT+GTLPMMODE=2,1 AT+CSCLK=1	The UART enters the idle state when the WAKEUP_IN pin is pulled low. The UART enters the idle state when the DTR pin is pulled low.
Auto mode	AT+GTLPMMODE=0 ATS24=3	The module enters the idle state if the serial port does not exchange data within 3s.

- The WAKEUP_IN/DTR pin(Some models only support one pin, please refer to the hardware manual for details) is preferred to control the UART to sleep or wake up, and the control by other pins needs to be customized.
- The control of NL668, MC116, and MG110 series modules is based on the UART connection mode, and USB is not connected. The USB controller is in the normal operating state. You need to send the AT+GTUSBSLEEPEN=2,0 command to disable USB and then control the module to enter the sleep state by leveraging the above methods.
- The L716 and MA510 series modules do not support CSCLK, and the L716 series modules do not support the control of sleep or wakeup by DTR pin. FG132 series modules do not support ATS24 and CSCLK instructions
- The MA510, NL668, MC116, MG110, and L716 series modules make the UART enter the idle state using the ATS24 mode. The first command sent by the HOST is the wakeup command, and the second command sent is used to respond to the AT command. The MC66x, MG66X, MC61x, L610 and Lx61X series modules make the UART enter the idle state using the ATS24 mode. The first command sent by the HOST is directly used for response. The LE270, LE170, LE130 and LE230 series modules make the UART enter the idle state using the ATS24 mode. The first command sent by the HOST

is used to respond to the AT command when the UART baud rate is 9600 and below. When the UART baud rate is higher than 9600, the first command sent by the HOST is the wakeup command, and the second command sent is used to respond to the AT command.

- Fibocom modules can wake up in any mode. The modules can completely wake up after a certain period of time. It is recommended that the HOST should wait for 500 ms before interacting with the module through AT commands (The MC66x, MG66X, MC61x, L610 and Lx61X series modules make the UART enter the idle state using the ATS24 mode. The first command sent by the HOST is directly used for response. The HOST does not need to wait for 500 ms before sending the next command).
- For details about the AT commands used to control the UART to enter the idle state, see Chapter 6.

4.2 Sleep Configuration Based on USB Connection

Since the UART is not in the idle state by default, the module can sleep only when both UART and USB are idle.

You can enable the USB to enter the idle state using the USB VBUS pin or USB SUSPEND command. The USB VBUS command is preferred. The following table lists the sample commands:

Table 5. Sleep configuration based on the USB connection

Configuration Command	Description
AT+GTLPMMODE=0 ATS24=1 USB VBUS or USB SUSPEND	The UART enters the idle state if there is no data exchange within 1s. The USB can enter the idle state by pulling the USB VBUS pin or using the USB SUSPEND command.

- For details about the USB SUSPEND command, see section 6.6.
- For the NL668, MC116, MG110 series, if you make the USB enter the idle state using the USB SUSPEND command, the ADB port cannot be present in the USB port combination. You can check whether there is an ADB port in the USB port combination using the GTUSBMODE command described in the AT command manual. When you make the USB enter the idle state by pulling down the USB VBUS pin, you also need to send AT+GTUSBDETECTEN=1 to enable the VBUS switch.
- The USB connection-based sleep for the L716 series modules should be configured separately. The host should send AT+GTSET="SLEEPMODE",1 and then pull down the VBUS pin or send the USB SUSPEND command to make the module sleep.
- The NL668 Mini and M.2 packages do not provide the USB VBUS pin.
- The MA510 and FG132 series do not support the USB SUSPEND feature.
- The FG132 series does not support the ATS24 instruction. The AT+GTLPMMODE=1,0 instruction needs to be sent. Make UART idle.

5 Wakeup Solution

5.1 HOST Wakeup Module

5.1.1 UART Wakeup

Use WAKEUP_IN/DTR to make the UART enter the idle state and control the WAKEUP_IN/DTR to wake up the module.

Use ATS24 to make the UART enter the idle state and send data to the serial port to wake up the module. The module automatically goes to sleep when the command execution times out.

Fibocom modules can wake up in any mode. The modules can completely wake up after a certain period of time. It is recommended that the HOST should wait for 500 ms before exchanging AT commands with the module. (The MC66x, MG66X, MC61x, L610 and Lx61X series modules make the UART enter the idle state using the ATS24 mode. The first command sent by the HOST is directly used for response. The HOST does not need to wait for 500 ms before sending the next command).

5.1.2 USB Wakeup

Use USB_VBUS to make the USB enter the idle state and pull up USB_VBUS to wake up the module.

Use Full Speed J of the USB bus to make the USB enter the idle state and make the bus send the Full Speed K signal to wake up the module.

5.2 HOST Wakeup by Module

5.2.1 Wakeup by RI/WAKEUP_OUT

When the module receives an SMS message, an incoming call, or data, the RI pin outputs a pulse waveform to wake up the HOST. The following table lists the example wakeup commands.

Table 6. Commands for wakeup by RI/WAKEUP_OUT

Command	Example	Description
AT+GTWAKE	AT+GTWAKE=0	The RI pin is at the high level by default. When the module receives an SMS message, an incoming call, or data, the RI pin outputs a low-frequency pulse waveform signal. The length of the output waveform is determined by the AT+WRIM command.

Command	Example	Description
	AT+GTWAKE=1,0	The WAKEUP_OUT pin is at the low level by default. When the module receives an SMS message, an incoming call, or data, the WAKEUP_OUT pin outputs a high-frequency pulse waveform signal. The length of the output waveform is determined by the AT+GTPMTIME command.
	AT+GTWAKE=1,1	The WAKEUP_OUT pin is at the high level by default. When the module receives an SMS message, an incoming call, or data, the WAKEUP_OUT pin outputs a low-frequency pulse waveform signal. The length of the output waveform is determined by the AT+GTPMTIME command.
	AT+GTWAKE=2,0	The RI pin is at the low level by default. When the module receives an SMS message, an incoming call, or data, the RI pin outputs a high-frequency pulse waveform signal. The length of the output waveform is determined by the AT+GTPMTIME command.
	AT+GTWAKE=2,1	The RI pin is at the high level by default. When the module receives an SMS message, an incoming call, or data, the RI pin outputs a low-frequency pulse waveform signal. The length of the output waveform is determined by the AT+GTPMTIME command.
AT+GTPMTIME	AT+GTPMTIME=200,200,1000	Set the RI to output a pulse waveform with a width of 1.2s (for details, see the example configuration of GTPMTIME wakeup waveform).
AT+WRIM	AT+WRIM=0,1000	Default value. When there is an incoming call, the RI outputs a pulse waveform with a width of 1000 ms.
	AT+WRIM=1,150	Default value. When an SMS message is received, the RI outputs a pulse waveform with a width of 150 ms.
	AT+WRIM=2,500	When data is received, the RI output pulse width becomes 500 ms.

Summary: When AT+GTWAKE≠0, the changes of pulse length is related to the GTPMTIME configuration.

When AT+GTWAKE=0, the changes of pulse length is related to the WRIM configuration.

- The L716 series modules need to be configured separately, and the host wakeup function needs to be configured through AT+GTWAKE=1,2 and AT+WRIM commands. The GTPMTIME command is not supported.
- For the NL668, MG110, MC116 and MA510 series modules, if the RI needs to output a waveform when an SMS message is generated, the AT+CNMI command needs to be enabled.
- Some modules do not provide the WAKEUP_OUT pin. For details, see the

hardware guide.

- For the AT commands used by the module to wake up the HOST, see Chapter 6.

5.2.1.1 WRIM Configuration Example

The following table lists the example configuration of WRIM wakeup waveform.

Table 7. WRIM wakeup waveform configuration

AT Configuration	Description
AT+GTWAKE=0	Wake up the host through WRIM configuration.
AT+WRIM=1,150	When an SMS message is received, the RI outputs a low-frequency pulse waveform with a width of 150 ms.

Upon receiving the SMS message, the module starts to wake up the HOST. The RI outputs a low-frequency pulse waveform with a width of 150 ms. The SMS URC is reported during this period. The following figure shows the RI waveform:

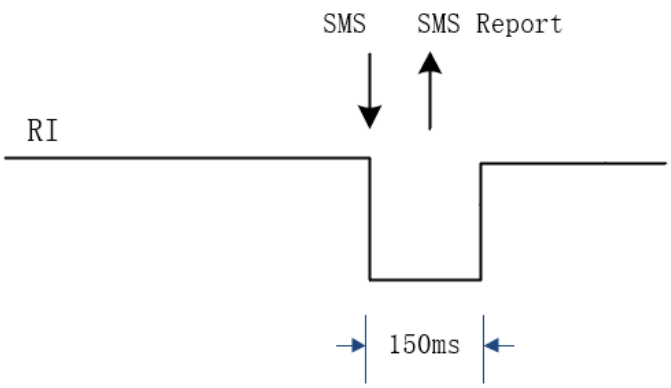

Figure 2. Example RI waveform

5.2.1.2 GTPMTIME Configuration Example

The following table lists the example configuration of GTPMTIME wakeup waveform.

Table 8. GTPMTIME waveform configuration example

AT Configuration	Description
AT+GTWAKE=2,1	Wake up the host through GTPMTIME configuration.
AT+GTPMTIME=300,200,1000	The RI outputs a low-frequency pulse waveform with a width of 1.3s.

The following figure shows the RI output pulse waveform.

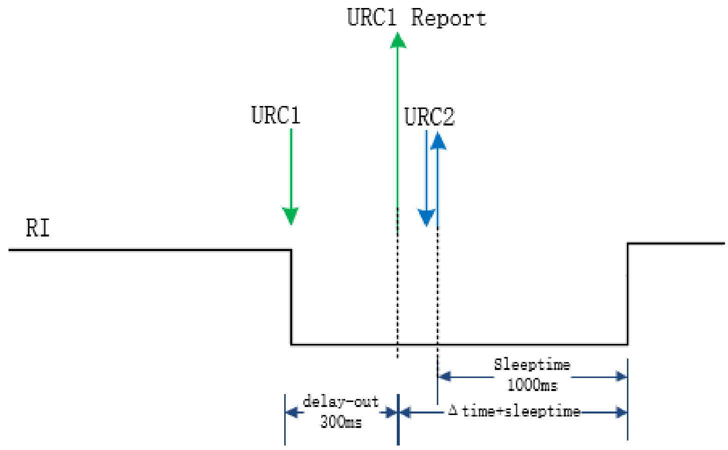

Figure 3. Example RI waveform

1. Upon receiving URC1, the module starts to wake up the HOST. URC1 is reported when the time exceeds the delay_out value. The module implements delayed sleep based on the sleep_time value.
2. If the module receives URC2 during the period of sleep_time, it directly reports URC2 without waiting for a period specified by delay_out and refreshes the sleep_time.

URC delayed reporting function is currently not supported.

5.2.2 Remote Wakeup by USB

A USB SUSPEND signal is sent to make the USB enter the idle state. When the USB of the module receives the URC, it reports the URC directly. When the module receives the PPP data and ECM NIC data, it sends the signal for remote wakeup by USB. The HOST receives the signal for remote wakeup by USB and sends the USB RESUEME signal to wake up the module. Then, the module sends the data out.

The HOST should support the function of remote wakeup by USB. To enable this function, see section 6.7.

6 Sleep and Wakeup Command Description

6.1 +GTLPMMODE

Command	Parameter	Description
AT+GTLPMMODE= <MAIN_MODE>, <SUB_MODE> (Storage upon power failure)	MAIN_MODE	Select the control mode. The default value is 0. 0: Cancel the pin control mode. 1: The WAKEUP_IN is used as the pin that controls the idle state of the UART. 2: The DTR is used as the pin that controls the idle state of the UART.
	SUB_MODE	0: When GPIO is at high level, the UART enters the idle state. When it is at low level, the UART exits the idle state. 1: When GPIO is at low level, the UART enters the idle state. When it is at high level, the UART exits the idle state.

6.2 +CSCLK

Command	Parameter	Description
AT+CSCLK=<MAIN_MODE>	MAIN_MODE	Whether to enable the sleep state. The default value is 0. 0: Disable the sleep state. 1: Enable the sleep state.

6.3 +S24

Command	Parameter	Description
ATS24=<AUTOSLEEP_TIME>	AUTOSLEEP_TIME	The default value is 0. That is, the UART does not enter the idle state. When there is no data within the period defined by AUTOSLEEP_TIME, the UART enters the idle state.

6.4 +GTUSBDETECTEN

Command	Parameter	Description

AT+GTUSBDETECTEN=<MODE> (Storage upon power failure)	MODE	Whether to enable the VBUS function. The default value is 0. 0: Disable the VBUS function. 1: Enable the VBUS function.

6.5 USB VBUS

The host controls the changes of the USB_VBUS level. The module detects USB_VBUS interrupt to determine whether the USB needs to enter the idle state. If the USB_VBUS is at low level, the USB enters the idle state. If the USB_VUBS is at high level, the USB exits the idle state.

6.6 USB SUSPEND/RESUME

Because Linux 2.6.32 or later versions have their own selective suspension power management feature, it is only necessary to enable the power management function of the host and configure it by modifying the kernel's compilation configuration. The modification is as follows:

```
/*Modify the config file of the host kernel*/
CONFIG_USB_SUPPORT=y
CONFIG_USB =y
CONFIG_PM_RUNTIME=y
CONFIG_USB_ACM=y
```

If the module uses the Linux system, modify linux_src/drivers/usb/serial/option.c:

```
#define FIBO_MODULE_VID    xxxx   //xxxx as FIBOCOM MODULE VID
#define FIBO_MODULE_PID    xxxx   //xxxx as FIBOCOM MODULE PID
static int option_probe(struct usb_serial *serial, const struct usb_device_id *id)
{
    if(((dev_desc->idVendor==cpu_to_le16(FIBO_MODULE_VID)&&(dev_desc->idProduct==cpu_to_le16(FIBO_MODULE_PID))))
    {
        usb_enable_autosuspend(serial->dev);
    }
}
```

If the module uses the FreeRtos system, modify linux_src/drivers/usb/class/cdc-acm.c:

```
#define FIBO_MODULE_VID    xxxx   //xxxx as FIBOCOM MODULE VID
#define FIBO_MODULE_PID    xxxx   //xxxx as FIBOCOM MODULE PID
```

```
static int acm_probe(struct usb_interface *intf, const struct usb_device_id *id)

{

    if(usb_dev != NULL)

    {

    if(((usb_dev->descriptor.idVendor==cpu_to_le16(FIBO_MODULE_VID))&&(usb_dev->desc
riptor.idProduct == cpu_to_le16(FIBO_MODULE_PID))))

    {

        usb_enable_autosuspend(usb_dev);

    }

    }

}
```

After modification, if the USB bus of the host does not exchange data within 2s, the host sends the Full Speed J signal to the module. If the USB bus exchanges data, the host sends the Full Speed K signal to the module. The following figure shows the SUSPEND and RESUME signal.

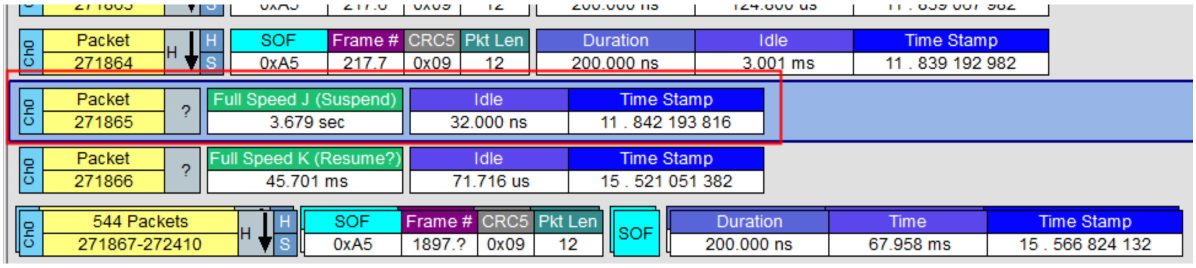

Figure 4. USB log

6.7 Remote Wakeup by USB

If the module uses the Linux system, modify linux_src/drivers/usb/serial/option.c:

```
#define FIBO_MODULE_VID    xxxx   //xxxx as FIBOCOM MODULE VID

#define FIBO_MODULE_PID    xxxx   //xxxx as FIBOCOM MODULE PID

static int option_probe(struct usb_serial *serial, const struct usb_device_id *id)

{

    if(((dev_desc->idVendor==cpu_to_le16(FIBO_MODULE_VID)&&(dev_desc->idProduct==cpu
_to_le16(FIBO_MODULE_PID))))

    {

        device_init_wakeup(&serial->dev->dev, 1);

    }
```

```
}
```

If the module uses the FreeRtos system, modify linux_src/drivers/usb/class/cdc-acm.c:

```
#define FIBO_MODULE_VID    xxxx   //xxxx as FIBOCOM MODULE VID

#define FIBO_MODULE_PID    xxxx  //xxxx as FIBOCOM MODULE PID

static int acm_probe(struct usb_interface *intf, const struct usb_device_id *id)

{

    if(usb_dev != NULL)

    {

    if((usb_dev->descriptor.idVendor==cpu_to_le16(FIBO_MODULE_VID))&&(usb_dev->descriptor.idProduct == cpu_to_le16(FIBO_MODULE_PID)))

    {

        device_init_wakeup(&serial->dev->dev, 1);

    }

    }

}
```

6.8 +GTWAKE

Command	Parameter	Description
AT+GTWAKE= <MAIN_MODE>, <SUB_MODE>	MAIN_MODE	0: The RI pin outputs the low-frequency pulse waveform to wake up the host according to the WRIM command. 1: The WAKEUP_OUT pin outputs the pulse waveform to wake up the host according to the GTPMTIME command. 2: The RI pin outputs the pulse waveform to wake up the host according to the GTPMTIME command.
	SUB_MODE	0: The WAKEUP_OUT/RI pin outputs the high-frequency pulse waveform to wake up the host. 1: The WAKEUP_OUT/RI pin outputs the low-frequency pulse waveform to wake up the host.

6.9 +WRIM

<table>
<tr><th>Command</th><th>Parameter</th><th>Description</th></tr>
<tr><td rowspan="2">AT+WRIM=
<TYPE>,
<TIME></td><td>TYPE</td><td>0: Voice
1: SMS
2: TCP data</td></tr>
<tr><td>TIME</td><td>Low-frequency pulse waveform</td></tr>
</table>

6.10 +GTPMTIME

<table>
<tr><th>Command</th><th>Parameter</th><th>Description</th></tr>
<tr><td rowspan="3">AT+GTPMTIME=
<delay_out >,
<delay_in>
<sleep_time>
(Storage upon power failure)</td><td>delay_out</td><td>Delayed data reporting time</td></tr>
<tr><td>delay_in</td><td>This parameter is non-effective.</td></tr>
<tr><td>sleep_time</td><td>Delayed module sleep time</td></tr>
</table>

7 FAQs

1. How to check whether the module enters the sleep state?
   a. Measure the current of the module to determine the state. For NL668, MC116, MG110, MA510, L716, MC66X, MG66X, MC61X, Lx61X and L610 series modules, the power consumption in sleep state is below 3 mA. For the LE170, LE130, LE270 and LE230 series modules, the power consumption in sleep state is below 1 mA.
   b. NL668, MC116, MG110, L716 enter sleep, and debug serial port cannot input.
2. How to troubleshoot the fault that the module cannot enter the sleep state?
   a. For the NL668, MC116, MG110 and L716 series modules, run debug uart to enter the module terminal. Run the following command to check the sleep lock that prevents the system from going to sleep:

```
# awk '$6 != 0 {print $1" "$6}' /sys/kernel/debug/wakeup_sources
```

   b. For the MC66X, MG66X, MC61X, L610 and Lx61X series modules, program the debug version and capture logs for analysis.
   c. For the LE series modules, run the AT+ECVOTECHK command to check the sleep lock that prevents the system from going to sleep:
3. The module enters sleep, but the power consumption is too high?
   a. Priority should be given to checking whether the sleep bottom current meets the specifications (refer to the hardware guide for specifications).
   b. MC66X, MG66X, MC61X, Lx61X, L610, LE series modules need to turn off the LPG light through AT+GTLPG=0 before going to sleep.
   c. Some models of the MC66X and MG66X series support built-in GNSS/BLE. Before going to sleep, GNSS/BLE must be turned off through AT/API (see AT/API manual for details).