

GSM .Net Library

for GTM Modem series

User's Manual V1.1



High Quality, Industrial Data Acquisition, and Control Products

Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, or for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright 2016 by ICP DAS CO., LTD. All rights reserved worldwide.

Trademark

The names used for identification only may be registered trademarks of their respective companies.

Contact us

If you have any problem, please feel free to contact us.
You can count on us for quick response.

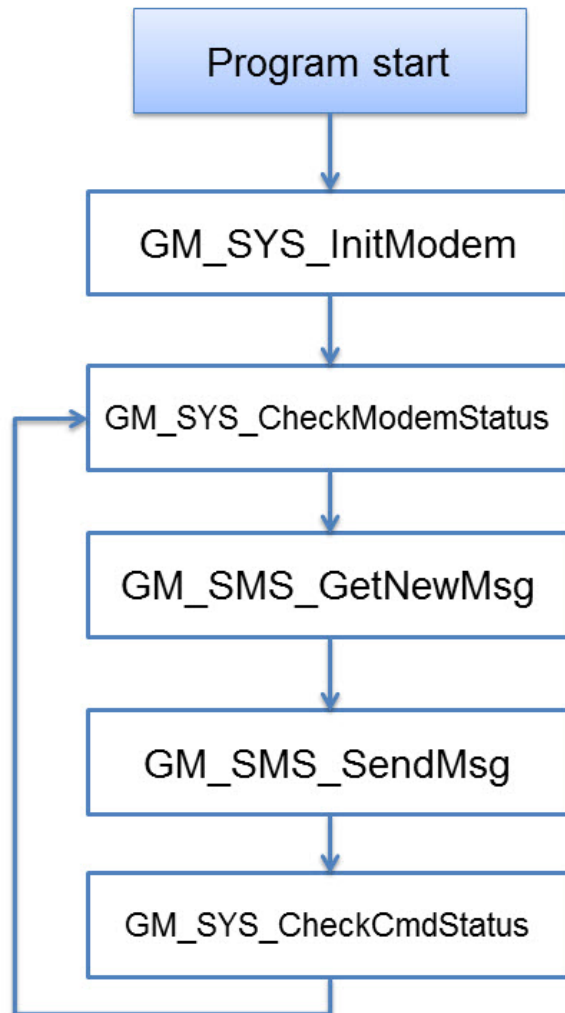
Email : service@icpdas.com

Table of Contents

1. Introduction	1
1.1 Design Flowchart.....	1
2. GSM .Net Library	2
2.1 Data Structure Define	2
2.2 SYSTEM Function.....	3
2.2.1 GM_SYS_GetLibVersion.....	4
2.2.2 GM_SYS_GetLibDate	5
2.2.3 GM_SYS_InitModem	6
2.2.4 GM_SYS_CloseModem	7
2.2.5 GM_SYS_CheckModemStatus	8
2.2.6 GM_SYS_CheckCmdStatus.....	9
2.2.7 GM_SYS_CheckSignal	10
2.2.8 GM_SYS_CheckReg	11
2.3 SMS Function.....	12
2.3.1 GM_SMS_SendMsg	13
2.3.2 GM_SMS_GetNewMsg.....	14
3. Revision History	15

1. Introduction

1.1 Design Flowchart



2. GSM .Net Library

2.1 Data Structure Define

There are some data structure that is useful when you program with GSM library.

SMS :

```
//-- class for sending/reading SMS
public class Encode_Msg
{
    public string phoneNumber;    //phone number
    public string time;          //sms_time_stamp
    public string msg;           //message's content
    public int mode;             //encode style: 0=GSM_7BIT, 8=GSM_UCS2(uni-code)
}
```

2.2 SYSTEM Function

Function definition	Description
GM_SYS_GetLibVersion	Get Library version
GM_SYS_GetLibDate	Get Library date
GM_SYS_InitModem	Initialize Modem
GM_SYS_CloseModem	Close the modem
GM_SYS_CheckModemStatus	Check modem status, and suggest you check it in your loop every time
GM_SYS_CheckCmdStatus	Get the status of the command you sent
GM_SYS_CheckSignal	Check signal quality
GM_SYS_CheckReg	Check register

2.2.1 GM_SYS_GetLibVersion

Get library version.

Syntax

```
public static string GM_SYS_GetLibVersion( );
```

Parameters

None

Return values

Version

2.2.2 GM_SYS_GetLibDate

Get library date.

Syntax

```
public static string GM_SYS_GetLibDate( );
```

Parameters

libDate

a string of lib. Date, format="Jul 21 2014"

Return values

None

2.2.3 GM_SYS_InitModem

Initialize Modem.

**must use GM_SYS_CheckModemStatus() to check modem status later

Syntax

```
public static int GM_SYS_InitModem(  
    string PinNum,  
    int modem  
);
```

Parameters

PinNum

PIN code

modem

port number of the modem

Return values

GM_NOERROR : success

GM_COMERROR : comport error

GM_INITERROR : init fail error

2.2.4 GM_SYS_CloseModem

Close the modem.

**Please call GM_SYS_InitModem() to wake up modem after using GM_SYS_CloseModem(1) to shut down the modem.

Syntax

```
public static int GM_SYS_CloseModem(  
    int mode  
);
```

Parameters

mode

0 : close modem, but maintain it power on

1 : close modem and set it power off

Return values

GM_NOERROR : no error

GM_CMDERROR : command error

2.2.5 GM_SYS_CheckModemStatus

Check modem status, and suggest you check it in your loop every time.

Syntax

```
public static int GM_SYS_CheckModemStatus( );
```

Parameters

None

Return values

GM_NOERROR : modem register success, can service

GM_NOREG : modem not registered, can't service

2.2.6 GM_SYS_CheckCmdStatus

Get the status of the command you sent.

Syntax

```
public static int GM_SYS_CheckCmdStatus( );
```

Parameters

None

Return values

GM_BUSY : modem busy, you can't send other command

GM_NOERROR : success

GM_TIMEOUT : time out

GM_CMDERROR : command error

Other : please refer to error codes of GSM.h

2.2.7 GM_SYS_CheckSignal

Check signal quality.

Syntax

```
public static int GM_SYS_CheckSignal( );
```

Parameters

None

Return values

signal quality

1	-113 dBm or less
1	-111 dBm
2...30	-109... -53 dBm
31	-51 dBm or greater

2.2.8 GM_SYS_CheckReg

Check register.

Syntax

```
public static int GM_SYS_CheckReg( );
```

Parameters

None

Return values

Register flag

- 0 : not registered
- 1 : registered, home network
- 2 : not registered, and searching...
- 3 : registration denied
- 4 : unknown
- 5 : registered, roaming

2.3 SMS Function

Function definition	Description
GM_SMS_SendMsg	Send a message
GM_SMS_GetNewMsg	Get a new sms message

2.3.1 GM_SMS_SendMsg

Send a message.

**must use "GM_SYS_CheckCmdStatus()" to check status later

Syntax

```
public static int GM_SMS_SendMsg(  
    Encode_Msg strMsg  
);
```

Parameters

strMsg

the message that will be sent.

Return values

GM_NOERROR : no error

GM_NOREG : not registered, or can't service

GM_BUSY : modem busy

2.3.2 GM_SMS_GetNewMsg

Get a new sms message.

Syntax

```
public static int GM_SMS_GetNewMsg(  
    ref Encode_Msg msg  
);
```

Parameters

msg
new sms message

Return values

0 : no new message
1 : new message coming

3. Revision History

Revision	Date	Author	Description
1.0	2015/07/03	William	Release version
1.1	2016/07/07	Eddie	Modify