Grpsdll.dll Document

Data structure：

1．Identification Dtu’s structure：

```
typedef UINT u32t;
typedef UCHAR u8t;
typedef USHORT u16t;
typedef ULONG       u64t;

typedef struct _modem_info_t_
{
      u32t   m_modemId;            //Dtu ID
      u8t        m_dynip[4];     // the dynamic  IP Dtu got
      u8t          m_phoneno[12];          //Dtu’s 11bytes phone, must finish with ‘\0’
      u64t m_conn_time;      //Dtu module TCP connection time last time
      u64t m_refresh_time;       // time for receiving or sending data    last time for Dtu module
} ModemInfoStruct;
```

2．Identification Dtu’s data packet：

```
typedef struct _modem_data_t {
          u32t          m_modemId;                    // Dtu ID
          u8t           m_dtuPhoneNo[12];       // Dtu’s 11bytes phone, must finish with ‘\0’
          u64t          m_recv_time;                  // time for receiving data packet
         u8t           m_data_buf[MAX_RECEIVE_BUF+1];// data
          u16t        m_data_len;                   //data length
          u8t         m_data_type;                  //data style received,
                                                    //     0x01：user data packet
                                                    //     0     ：unrecognizable style
} ModemDataStruct;
```

U64t type time is based on GMT time’s second form 1970-01-01 00:00:00 to now, In a word , it is a Time Interval that measure by the second, most of program language supply API for transformation

Api Description：

1）．BOOL DSStartService(u16t uiListenPort)

function：start a data service for server.

parameter： u16ListenPort:the listening port

description：server will listen the specify port , after you started the data server.

If failed, you can get Error information throught DSGetLastError

2）.BOOL DSStopService(void)

function：stop service

parameter：none

description：stop server's data service ,all DTU will offline

if failed ,you can get Error

3）.BOOL DSGetNextData(ModemDataStruct* pDataStruct,u16t waitseconds)

function： read next information from DTU

parameter：pDataStruct: if successful, the storage data from DTU

waitseconds:this function will return at once when data arrive,if no data, then will wait waitseconds maximum second, range from 0 to 65535.if this value is set zero, this fuction will return at once too. Besides **DSStopService**（） was executed in another thread

return value：if not zero, data was received successful,or no data recived

4).BOOL _stdcall DSSendData(u32t modemId,u16t len,u8t * buf);

Function: send data to appointed modem

Parameter: modemID :PTR which point to 8 bytes modemID:

Len: data length you will send must no more than 1450 bytes

buf:data you will send.

If failed, you can get report throuth DSGetLastError()

5）.u32t DSGetModemCount(void)

function：get all online DTU counts；

6）.BOOL DSGetModemByPosition (u32t pos, ModemInfoStruct *pDtuInfo)

Function: get data from the appointed DTU

Parameter:Pos:DTU's position, begin from zero

pDtuInfo:point to DTU structure to save DTU information

In a general,by **DSGetModemCount** () function coordinateing with **DSGetModemByPosition**,you can get all Dtu Information you want

For example

```
u32t uiDtuCount;
uiDtuCount= DSGetModemCount ();
ModemInfoStruct dtuInfo;
u32t i;
for (i=0;i< uiDtuCount;i++)
{
    DSGetModemByPosition (i,& dtuInfo);
    //operate dtuInfo
}
```

7)BOOL _stdcall DSSendControl(u32t modemId,u16t len,u8t * buf);

function：send control command to apponinted modem

Parameter：modemId:modem's ID number，it is Identifies a Modem

Len: the length of the data of command which you want to send.data length you will send must no more than 1000 bytes

buf:the command you will send

If failed, you can get report throuth DSGetLastError()

8）. void DSGetLastError(char *str,int nMaxStrSize);

function：get error from last API

Parameter：str:the error information buffer

nMaxStrSize:the maximum of buffer length, if your error more than it, the last will be interceped

Call Method

1、 Load Library：

Call windows API function LoadLibrary to load this gprsdll.dll.：

```
HMODULE hDllMudule;                    //point to the LIB Handle
hDllModule=LoadLibrary("gprsdll.dll");
If (hDllModule!=NULL)      //check that loading library is successful or not
  {
        //get function address from library
        DSStartService =GetProcAddress(hDllModule,"DSStartService");
        if (DSStartService!=NULL)     // check that get address is successful or not
         {
              if ((*DSStartService)( 5001)!=FALSE)
                    MessageBox("start sucessfully");
               Else
                    MessageBox("start failed");
         }
  }
```

You can call FreeLibrary to free library,so LoadLibrary and FreeLibrary's count must equal.

```
FreeLibrary(hDllModule);     //TRUE-success    FALSE-failed
```

2、 Start service：

```
BOOL (*DSStartService)(u32t);   //define a point to point this fucntion
DSStartService=
      (BOOL(*)(u32t))GetProcAddress(hDllModule," DSStartService");
if (DSStartService!=NULL)
{
    (*DSStartService)( 5001);
}
```

3、 Stop service：

```
BOOL (*DSStopService)(void);
DSStopService=(int (*)(char *))GetProcAddress(hDllModule," DSStopService");
If (DSStopService!=NULL)
          (*DSStopService)();
```

4、 Read data：

This function must be executed in thread or timer.

```
BOOL DSGetNextData(DtuDataStruct*, u16t);
DSGetNextData = \
    (BOOL(*)(DtuDataStruct*, u16t))GetProcAddress(hDllModule," DSGetNextData");
if (DSGetNextData!=Null)
    if ((*DSGetNextData)(&dtudata,100)==0)
       {
            //deal with dutdata
        }
```

5、 Send data：

```
BOOL DSSendData (unsigned char *,u16t,u8t *);
DSSendData= \
(BOOL(*)(DtuInfoStruct*,u16t,u8t*))GetProcAddress(hDllModule," DSSendData");
if (DSSendData!=NULL)
{
    (*DSSendData)(szPhone, len,buf);
}
```

6、DTU list：

Have created a dut list in this dll service ,if DSC want to know this List information,need call API **DSGetModemCount**、**DSGetModemByPosition**

```
for (u32t I=0;I<(* DSGetDtuCount)();I++)
{
        (*DSGetModemByPosition)(I,&dtuinfo);

}
```

7、Error information：

Get the last Called API Error report By calling DSGetLastError()function