



Access interface of Web server terminal side

(Modification: item 5.6)

1. Interface protocol of software on PC

1.1 User login (PC→LED data center)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Command 1byte	Reserved 1byte	Main body length 2bytes	Main body		Check out 1byte
0x7e0x7e0x7e0x7e	NULL	NULL	0x50		24	User 16byte s	Password 8bytes	

Sum checkout: sum invert starting from IMEI to the end of Main body including command byte.

1.2 User change password and login (PC→LED data center)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Comm and 1byte	Reser ved 1byte	Main body length 2bytes	Main body			Chec kout 1byte
0x7e0x7e0x7e0x7e	NULL	NULL	0x57		32	User 16byt es	Old password 8bytes	New password 8bytes	

Receive respond

(LED data center → PC)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body		Chec kout 1byte
0x7e0x7e0x7e0x7e			0x51		1	Return value		

Return value: -1: Authentication failed, socket error needs get communication again 0: Authentication failed;

2: user not exist 3: success 4: Wrong password

5: old password incorrect 6: modify new password error 7: the user login in other places, force to log out in other places, login success; 8: system maintaining

2 Get DTU information

Type 1 : Get DTU information by command (PC→LED data server center) (check all DTU information when IMEI number is empty)

Protocol head4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body		Chec kout 1byt e



0x7e0x7e0x7e0x7e			0x03		0	Empty	
Respond (led data server center→PC)							
Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body	Chec kout 1byte
0x7e0x7e0x7e0x7e			0x05		2	Status 1byte	Access type

Status: 0x00: no this DTU; 0x01: DTU online; 0x02: DTU offline; 0x03: DTU abnormal

Access type: **enum** _INTERNET_COMMUNICATION_TYPE

```
{
    INTERNET_COMMUNICATION_TYPE_NONE=0,//unknown
    INTERNET_COMMUNICATION_TYPE_NET_WIRE,//wire
    INTERNET_COMMUNICATION_TYPE_NET_WIFI,//wifi
    INTERNET_COMMUNICATION_TYPE_MOBILE_3G,//3G
    INTERNET_COMMUNICATION_TYPE_MOBILE_2G,//GPRS
    INTERNET_COMMUNICATION_TYPE_ADSL,//ADSL
    INTERNET_COMMUNICATION_TYPE_MODEM,//MODEM
    INTERNET_COMMUNICATION_TYPE_COM,//serial port
};
```

Type 2 : Get DTU information from database (led data server center binds to this database, to modify online/offline for all DTU in real time and dynamically)

Database name: UserCenter(can modify according to configuration file)

Table name: Cards

Structure:

Column name	Database type	Allow Null value	Description
IMEI	Nvarchar(128)	NO	Major key, unique mark
CardName	Int	Yes	Spare name
State	Int	NO	1: online, 2: offline
AccessMethod	Int	NO	
Del	Bit	NO	1:deleted, 2:not deleted
Width	Int	YES	
Height	Int	YES	
DiskSize	Int	YES	
MainVer	Int	YES	
ChildVer	Int	YES	
SoftwareVer	Int	YES	
HardwareVer	Int	YES	
Brightness	Int	YES	



Company	Nvarchar(MAX)	YES	
CardType_CardTypeId	Int	YES	Foreign key, card type
CardGroup_CardGroupId	Int	YES	Foreign key, card group

Stored procedure of updating database:

```
CREATE PROCEDURE update_dtu_status
```

```
@dtu_imei varchar(64),          /*对应字段: IMEI*/
@dtu_cardname varchar(64),      /*对应字段: CardName*/
@dtu_master varchar(64),        /*对应字段: Company*/
@dtu_status int,                /*对应字段: State*/
@dtu_accessmode int            /*对应字段: AccessMethod*/
```

```
AS
```

```
DECLARE @type_id int
```

```
IF @dtu_status=2
```

```
    BEGIN
```

```
        UPDATE cards SET state=@dtu_status WHERE imei=@dtu_imei
```

```
    END
```

```
ELSE
```

```
    BEGIN
```

```
        SELECT @type_id=CardTypeId FROM cardtypes WHERE
```

```
cardtypename=substring(@dtu_imei, 1, 3)
```

```
        IF @@ROWCOUNT=0
```

```
            BEGIN
```

```
                INSERT INTO cardtypes(cardtypename) VALUES (substring(@dtu_imei, 1, 3))
```

```
                SET @type_id=@@IDENTITY
```

```
            END
```

```
        SELECT * FROM cards where imei=@dtu_imei
```

```
        IF @@ROWCOUNT=0
```

```
            BEGIN
```

```
                INSERT INTO cards (imei,cardname,Company,state,accessmethod,
```

```
CardGroup_CardGroupId, CardType_CardTypeId) VALUES
```

```
(@dtu_imei,@dtu_cardname,@dtu_master,@dtu_status,@dtu_accessmode, 1, @type_id)
```

```
            END
```

```
        ELSE
```

```
            BEGIN
```

```
                UPDATE cards SET cardname=@dtu_cardname,state=@dtu_status,Company=
```

```
@dtu_master ,accessmethod=@dtu_accessmode, Del=0 WHERE imei=@dtu_imei
```

```
            END
```

```
    END
```

Stored procedure of deleting DTU:



```
CREATE PROCEDURE del_dtu
@dtu_imei varchar(64)
AS
BEGIN
UPDATE cards SET [State]=2, Del=1 WHERE imei=@dtu_imei
END
```

3 User log out (PC→ data server center)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body	Chec kout 1byte
0x7e0x7e0x7e0x7e	NULL	NULL	0x60		16	User 16bytes	

Receive response (data server center→PC)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body	Chec kout 1byte
0x7e0x7e0x7e0x7e			0x61		1	0success 1 failed	

4 PC side receive and send data protocol (data server center←→PC)

Protocol head 4bytes	IMEI 16bytes	Spare name 16 bytes	Com mand 1byte	Reser ved 1byte	Main body length 2bytes	Main body	Chec kout 1byte
0x7e0x7e0x7e0x7e			0x00			5.1~5.6 blue color content	

5.1Get terminal side information (web server-→terminal side)

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x17	0x00000000	sizeof(SYSTIMER)	SYSTIMER	

Response

Protocol head 3bytes	Com man d 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes



0x7e0x7e0x55	0x57	0x00000000	sizeof(STRUCT_SYSTEM_STATUS)	STRUCT_SYSTEM_STA TUS	
--------------	------	------------	------------------------------	--------------------------	--

5.2turn on/off led sign

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x10	0x00000000	Sizeof (UINT)	0 off 1 on UINT	

Response

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x50	0x00000000	Sizeof (UINT)	0 success 1failed UINT	

5.3adjust time

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x11	0x00000000	sizeof(SYSTIMER)	SYSTIMER	

Response

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x51	0x00000000	Sizeof (UINT)	0 success 1 failed UINT	

5.4adjust brightness

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x12	0x00000000	Sizeof (UINT)	Brightness value UINT	

Response

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x52	0x00000000	Sizeof (UINT)	0 success 1 failed UINT	

**5.5 remote anycast programs in controller**

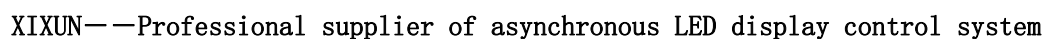
Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0x97	0x00000000	Size of (UINT)	Program number UINT	

Response

Protocol head 3bytes	Command 1byte	Reserved 4bytes	Main body length 4bytes	Main body	Checkout 1bytes
0x7e0x7e0x55	0XD7	0x00000000	Main body character string length	“Success” success other failed	

5.6 inform controller to download program from specific FTP

Protocol head 3bytes	command 1byte	Reserved 4bytes	Main body length 4bytes	Main body												Checkout 1bytes
0x7e0x7e0x55	0x05	0x00000000		Web response way 0x01	Domain name character string (must end with '\0') 64 bytes	P o t l N T	M e r o d w e b c h a r a c t e r s t r i n g ((m u s t e n d w i t h '\0')) 25 6 b y t e	P r o g r a m s i z e	M a s t e r s i o n	S u b v e r s i o n	W i d t h	H e i g h t	P r o g r a m n a m e c h a r a c t e r s t r i n g			



Response

NOTE: Web response way

Method webpage: GET_/Task/UpdateState?imei=M10-550-00001&taskId=1&statId=

Get the controller status of downloading program from FTP

Controller side:

Controller side (terminal side) returns status of downloading program (updating WEB database)



1. Build socket with website, IP: WEB's IP address, Port: WEB's port;

Send character string:

"GET_/Task/UpdateState?imei=M10-550-00001&taskId=1&stateId=2_HTTP/1.1\r\nHost:www.ledm2m.net:8083\r\n\r\n", while "_" is a blank space in actual character string,

Host: www.ledm2m.net:8083 this is the domain name of website and port.

Server side:

Receiving character string instruction:

- this character string is HTTP protocol head,
- GET means way of request
- "/Task/UpdateState" is the website in root of website, write down the method of how to updating database in this webpage, receiving 3 parameters imei (controller serial number), taskId (task id), stateId (task status's ID) , for example: Update_TaskState(imei, taskId, stateId);
- Request format of Get with parameter is : WebPath?parameter1=value1¶meter2=value2¶meter3=value3&....
- "\r\n" means put newline cursor at first place;
- Attention: please do not forget "\r\n" in this character string (for HTTP protocol 's rule)

NOTE: APP response way

Send status directly via socket of receiving command;

annotations: Related structure definition

```
typedef struct _SYSTIMER
{
    int y; //get year information
    int m; //get current month information
    int d; //get date information
    int h; //get current hour information
    int mm; //get minutes information
    int s; //get seconds information
    int w; //get week information, 1 is Sunday, 7 is Saturday
}SYSTIMER,*P_SYSTIMER;
```




```
typedef struct _ST_VERSION
{
    int iMainVer;
    int iChildVer;
}ST_VERSION;
typedef struct _ST_CARD_VERSION
{
    int iSoftwareVer;
    int iHardwareVer;
}ST_CARD_VERSION;
typedef struct _STRUCT_SYSTEM_STATUS
{
    int iWidth;
    int iHeight;
    int iStatus;
    int iDiskSize;
    ST_VERSION stVersion;
    ST_CARD_VERSION stCardVersion;
    int iBrightValue;
    int iPasswordId;//20100324
    int iPassword;//20100324
}STRUCT_SYSTEM_STATUS;
```

Check code calculate functions

```
unsigned char    GetCheckCodeIn8(unsigned char * pBuffer,unsigned int uiSize)
{
    unsigned int    i=0;
    unsigned char    ucCheckCode=0;
    for (i=0;i<uiSize;i++) ucCheckCode+=pBuffer[i];
    return (~ucCheckCode)&0x0ff;
}
```