

StarRiver 数据库设计

卿培 <qingpei@sansitech.com>
研发一部软件组
上海三思电子工程有限公司

2014 年 8 月 1 日

目录

1	概述	3
1.1	通信方式	3
1.2	特殊的表	3
1.3	表的命名规则	3
1.4	表中读写的字段	3
1.5	其他	3
2	流程	4
2.1	前台发送一般的命令	4
2.2	前台上载文件	4
3	表结构	5
3.1	SystemConfig	5
3.2	User	5
3.3	Controller	6
3.4	ControllerStatus	7
3.5	ControllerStatusChanges	7
3.6	ControllerSchedule	8
3.7	ControllerCellNumbers	8
3.8	ControllerSms	8
3.9	ControllerLog	9
3.10	Device	9
3.11	DeviceStatus	9
3.12	DeviceStatus_edit	11
3.13	DeviceStatusChanges	11
3.14	DeviceStatusHistory*	12
3.15	DeviceSchedule	13
3.16	DeviceScene	13
3.17	DeviceReportCond	13
3.18	DeviceAlarmCond	14
3.19	DeviceMessages	14
3.20	TaskTodo	14
3.21	TaskDone	15
3.22	Firmware	15
3.23	FrontendMapBmp	16
3.24	FrontendGroup	16
3.25	FrontendScene	16
3.26	FrontendControllerDevices	16
3.27	FrontendGroupDevices	17
3.28	FrontendDeviceMap	17
4	附录：表 TaskTodo 的任务类型及参数	18

1 概述

1.1 通信方式

后台程序采用异步通信方式，除非有明确的时序关系，否则全部操作都可以认为是并行的。

1.2 特殊的表

- **TaskTodo**: 记录待操作的命令。前台程序添加的用户指令，或是后台程序添加的定时操作。
- **TaskDone**: 记录已完成的命令的结果。命令完成后会从 TaskTodo 中删除，并在此插入一条记录。

1.3 表的命名规则

- 控制器相关的表，加 Controller 前缀；
- 终端设备相关的表，加 Device 前缀；
- 前台程序相关的表，加 Frontend 前缀。

1.4 表中读写的字段

除非明确指出，否则前台程序从 Controller* 和 Device* 表中读数据，写操作主要在 Frontend* 和 TaskTodo 中进行。

通信程序不写 Frontend* 表。

前台程序和通信程序时间同步：如果有 Internet 连接，都与 `cn.pool.ntp.org` NTP 服务器对时；否则可以考虑在局域网内由一台 Windows Server 或 Linux 主机作为 NTP 服务器，其他主机与该主机对时。

多个前台程序进程：依赖不同的用户名进行区分。暂不考虑软件上限制同一用户多处同时登录的情况。

1.5 其他

假设表中某个字段定义了 N 个字节，但是某个协议只用 M 个字节 ($M < N$)，默认都是用最低 M 个字节，除非有特殊说明。

目前仅支持系统下的控制器属于同一种协议。

2 流程

2.1 前台发送一般的命令

1. 前台线程往 `TaskTodo` 插一条记录
2. 通信线程从 `TaskTodo` 读记录
3. 通信线程执行，对相关的表进行必要的更新，然后把结果写到 `TaskDone` 。
4. 前台线程从 `TaskDone` 读结果，去相应表读数据，然后显示。

2.2 前台上载文件

暂定：配置一个文件服务器，协议可以从 FTP、HTTP、Samba、NFS 中选择。

前台程序往服务器上传一个文件后记录该文件的 URI 。

通信程序通过参数中读到的 URI 提取该文件。

3 表结构

除特殊说明外，每张表的第一个字段为主键 (Primary Key)。

3.1 SystemConfig

字段名	类型	含义	备注
idSystemConfig	int unsigned AUTO_INCREMENT		
NetworkMode	tinyint unsigned	通信程序角色	0: 服务器, 1: 客户端。考虑同时作为多种角色与不同控制器通信的需求
NetworkPort	smallint unsigned	服务器侦听端口	
LastServerActivity	timestamp	通信程序最近连接时间	通信程序定时更新这个字段。
IntervalReportActivity	int unsigned	更新最近连接时间间隔	
IntervalAutoStatusQuery	int unsigned	自动查询控制器状态间隔	
IntervalAutoDeviceStatusQuery	int unsigned	自动查询设备状态间隔	
IntervalRetryWrite	int unsigned	连接建立期间，写操作重试间隔	单位为毫秒
IntervalKeepalive	int unsigned	心跳包发送间隔	
TimeoutCmdAck	smallint unsigned	指令响应超时	CSA 6.1 T1
TimeoutCmdResult	smallint unsigned	指令结果超时	CSA 6.1 T2
TimeoutEventAck	smallint unsigned	事件响应超时	CSA 6.2 T3
TimeoutIdle	smallint unsigned	通信空闲时间	CSA 6.3 T4
TimeoutKeepaliveAck	smallint unsigned	心跳响应超时	CSA 6.3 T5
RetryCmd	tinyint unsigned	指令重试次数	CSA 6.1 N1
RetryEvent	tinyint unsigned	事件重试次数	CSA 6.2 N2
RetryKeepalive	tinyint unsigned	心跳重试次数	CSA 6.3 N3
MapFormat	int	地图类型	0: BMP, 1: GIS

3.2 User

字段名	类型	含义	备注
idUser	int unsigned		
Username	varchar(64)		
MD5	varchar(40)	MD5(username+password)	128-bit, 32 digits long hex string
Admin	tinyint	用户权限	0: 通信程序; 1: 一般用户, 2: 管理员
Note	varchar(450)	备注	UTF8

默认分配如下两个用户：

idUser	Username	备注
0	comm	通信程序
1	admin	管理员

3.3 Controller

字段名	类型	含义	备注
idController	int unsigned		
Name	varchar(128)		
MAC	binary(8)	8 字节控制器地址	
CommProtocol	tinyint unsigned	通信协议	0: Shanghai, 1: Sansi LC300, 2: CSA
DataProtocol	tinyint unsigned	通信方式	0: 串口, 1: TCP, 2: UDP
ComPort	varchar(8)	串口号	COM0, COM1, ttyS0, ttyS1 等
ComBaud	int unsigned	串口波特率	9600, 115200 等
ComAddr	tinyint unsigned	使用 485 串口通信时指明地址	
IPAddr	varchar(16)		xxx.xxx.xxx.xxx
IPPort	smallint unsigned		
Note	varchar(450)	备注	比如位置信息
DisplayOrder	int unsigned	前台程序界面上显示的顺序	不宜重复

3.4 ControllerStatus

状态量不支持查询的话对应字段写 NULL。

字段名	类型	含义	备注
idController	int unsigned	Controller.idController	
CommState	tinyint unsigned	通信状态	0: 正常, 1: 故障
LoginState	tinyint unsigned	是否已注册到通信程序服务端	0: 未注册, 1: 已注册
VersionSoftware	binary(4)	软件版本	
VersionSystem	binary(4)	系统版本	
VersionKernel	binary(4)	内核版本	
VersionHardware	binary(4)	硬件版本	
ManufacturerInfo	varchar(450)	厂商信息文本内容	UTF8
Type	binary(2)	控制器类型	上海: 0x0001 => 485 控制器, 0x0002 => PLC 控制器, 0x0003 => 无线控制器。LC300 同上
WorkMode	binary(1)	工作模式	上海: 0x01 => 远程控制模式, 0x02 => 时控工作模式。LC300 同上。
StatusCode	binary(1)	控制器状态代码	上海: 0x00 => 正常工作, 0x01 => 下行通道异常, 0x02 => 升级模式, 0xFF => 未知错误。LC300 未描述。
LastTime	datetime	最后一次从控制器读取的时间	用于检查控制器时间是否正常
SmsCount	smallint unsigned	短信数量	从控制器读回来的

3.5 ControllerStatusChanges

记录控制器状态变化。

字段名	类型	含义	备注
idController	int unsigned	Controller.idController	PK1
field	varchar(32)	被修改的字段名	PK2
field_value	varchar(255)	变化后的字段值	
time	timestamp	修改时间	PK3

目前记录如下两个状态的变化, 由 ControllerStatus 表中名为 log_controller_changes 的 trigger 执行:

1. CommState: field_value 存储 CAST(CommState AS char)
2. StatusCode: field_value 存储 HEX(StatusCode)

3.6 ControllerSchedule

查询时控计划时把结果放到这张表里, 将原来该控制器的删掉, 再重新插入。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	
idController	int unsigned	Controller.idController	
Item	varchar(128)	时间亮度表的一项	上海: MM, DD, MM, DD, HH, mm, ss, Addr, mode, value (逗号后不加空格)

3.7 ControllerCellNumbers

控制器上的手机号。往控制器上面加载手机号, 或者从控制器上下载手机号, 都是一帧就完成了。删除手机号是把所有的手机号删除, 不能只删除其中的一个。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	
idController	int unsigned	Controller.idController	
Cell	varchar(20)	一个手机号	不支持国家代码

3.8 ControllerSms

从控制器里面读回来的短信内容放在这张表里面。

字段名	类型	含义	备注
idController	int unsigned	Controller.idController	PK1
SmsIndex	smallint unsigned	短信在设备中的序号	PK2
SmsContent	varchar(450)		UTF8

3.9 ControllerLog

从控制器里面读回来的日志内容放在这张表里面。

字段名	类型	含义	备注
idController	int unsigned	Controller.idController	PK1
LogTime	datetime	日志记录时间	PK2
LogContent	text	日志内容	UTF8

3.10 Device

字段名	类型	含义	备注
idDevice	int unsigned		
Name	varchar(128)		
MAC	binary(8)		
DisplayOrder	int unsigned	前台程序界面上显示的顺序	
Note	varchar(450)	备注，比如位置信息	UTF8

3.11 DeviceStatus

字段名	类型	含义	备注
idDevice	int unsigned	Device.idDevice	
VersionSoftware	binary(4)	软件版本号	LC300: 只用 2 字节
VersionHardware	binary(4)	硬件版本号	LC300: 只用 2 字节
ManufacturerInfo	varchar(450)	厂商信息文本	上海: 厂商描述。 LC300: 终端设备产品信息, <=32 字节, ASCII 码。
ManufactureTime	datetime	出厂时间	LC300: 无
Type	binary(4)	设备类型	上海 4.4.1 LC300 4.6.1 0x0E
SN	varchar(64)	产品序列号	
InputVolt	double	输入电压采样值	
InputAmp	double	输入电流采样值	
OutputVolt	double	输出电压采样值	
OutputAmp	double	输出电流采样值	

字段名	类型	含义	备注
ActivePower	double	有功功率采样值	
Temperature	smallint	温度采样值	LC300: 1 字节
TemperatureGuard	binary(4)	过温保护状态 过温保护参数	LC300 4.6 0x0A
Uptime	int unsigned	上电工作时间	
UptimeTotal	int unsigned	总工作时间	
ElectricityConsumption	int unsigned	消耗电量值	
TransitionTime	tinyint unsigned	调光渐变时间	
Brightness	tinyint unsigned	当前亮度	
BrightnessMin	tinyint unsigned	最小亮度值	
BrightnessMax	tinyint unsigned	最大亮度值	
BrightnessMinPhysical	tinyint unsigned	物理最小亮度值	
BrightnessMaxPhysical	tinyint unsigned	物理最大亮度值	LC300 未定义
BrightnessPowerOn	tinyint unsigned	上电亮度值	
BrightnessDefault	tinyint unsigned	默认故障亮度值	
BrightnessCoefficient	tinyint unsigned	调光系数	实际亮度 = 设置亮度 * 调光系数 / 100
Status	binary(1)	设备状态	上海: 4.5.1 LC300: 4.6 0x06
StatusComm	binary(1)	通讯状态	上海: 4.5.1
StatusLamp	binary(1)	灯具状态	上海: 4.5.1
SensorI	binary(4)	光强传感器采样值	上海: 光感亮度采样值
LC300: 终端传感器设备光强值			
SensorL	binary(4)	光照传感器采样值	LC300: 终端传感器设备光照值
SensorH	binary(4)	湿度传感器采样值	
SensorT	binary(4)	车流量传感器采样值	
GroupMask	binary(32)	分组掩码	一共 256 bit, 若灯属于第 N 个组, 则 bit(N)=1。上海: 组地址是 1-254。LC300: 组地址是 0-63

3.12 DeviceStatus_edit

StarRiver Config 在此记录用户录入的设备初始信息。

字段名	类型	含义	备注
idDevice	int unsigned	Device.idDevice	
TransitionTime	tinyint unsigned	调光渐变时间	
BrightnessMin	tinyint unsigned	最小亮度值	
BrightnessMax	tinyint unsigned	最大亮度值	
BrightnessMinPhysical	tinyint unsigned	物理最小亮度值	
BrightnessMaxPhysical	tinyint unsigned	物理最大亮度值	LC300 未定义
BrightnessPowerOn	tinyint unsigned	上电亮度值	
BrightnessDefault	tinyint unsigned	默认故障亮度值	
BrightnessCoefficient	tinyint unsigned	调光系数	实际亮度 = 设置亮度 * 调光系数 / 100
GroupMask	binary(32)	分组掩码	一共 256 bit, 若灯属于 第 N 个组, 则 bit(N)=1。 上海: 组地址是 1-254。 LC300: 组地址是 0-63。

3.13 DeviceStatusChanges

记录设备状态变化。

字段名	类型	含义	备注
idDevice	int unsigned	Device.idDevice	PK1
field	varchar(32)	被修改的字段名	PK2
field_value	varchar(255)	变化后的字段值	
time	timestamp	修改时间	PK3

目前记录如下三个状态的变化, 由 DeviceStatus 表中名为 log_changes 的 trigger 执行:

1. Status: field_value 存储 HEX(Status)
2. StatusComm: field_value 存储 HEX(StatusComm)
3. StatusLamp: field_value 存储 HEX(StatusLamp)

3.14 DeviceStatusHistory*

以下三张表的字段定义大部分相同，分别记录不同粒度的状态量历史。

1. DeviceStatusHistory
2. DeviceStatusHistory_hourly
3. DeviceStatusHistory_daily

三张表共同部分的定义如下：

字段名	类型	含义	备注
idDevice	int unsigned	Device.idDevice	PK1
field	varchar(32)	被修改的字段名	PK2
field_value	double	字段采样值	
time	timestamp	修改时间	PK3

两张汇总表有额外记录汇总时段内峰值、谷值的字段。

1. DeviceStatusHistory_hourly
2. DeviceStatusHistory_daily

峰谷值定义如下：

字段名	类型	含义	备注
field_max	double	汇总时段内最大采样值	浮点数可能无法精确表示整数采样
field_min	double	汇总时段内最小采样值	浮点数可能无法精确表示整数采样

有三个 event 更新这些表：

1. 每 5 分钟，对 DeviceStatus 进行采样，记录设备的状态量。
2. 每天，将前一天的采样归纳成每小时均值。
3. 每周，将前一周的采样归纳成每日均值。

3.15 DeviceSchedule

LCP-SH-D: 该功能是在单灯控制器与集中控制器失去联系后采用的异常模式, 如果在某一时间段内没有设置该亮度值的话, 将采用默认故障亮度值显示。

注意调光计划的总数不能超过 7。

LC300 没有定义。

字段名	类型	含义	备注
idDeviceSchedule	bigint unsigned	自增 ID	
idDevice	int unsigned	Device.idDevice	
Item	varchar(32)	自动亮度表的一项	HH,mm,value

3.16 DeviceScene

LCP-SH-D 协议场景号是 1-254。LC300 协议, 场景号是 1-16, 0 表示无场景。

字段名	类型	含义	备注
idDeviceScene	tinyint unsigned		
idDevice	int unsigned	Device.idDevice	
Brightness	tinyint unsigned	灯的场景亮度值	0-100

3.17 DeviceReportCond

信息上报条件。用于保存要给哪些灯设置信息上报条件。

LC300 未定义。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	
idDevice	int unsigned	Device.idDevice	
Item	varchar(128)	信息上报的一项	StatusID,low,high

3.18 DeviceAlarmCond

报警阈值条件。用于保存要给哪些灯设置报警阈值条件。

LC300 未定义。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	
idDevice	int unsigned	Device.idDevice	
Item	varchar(450)	报警阈值的一项	StatusID, low, high, warn_msg (逗号后不加空格)

3.19 DeviceMessages

和上面两张表相关，上面设置了信息上报条件，这样对上报上来的信息应该有一个地方记住。同一个类型的上报报警信息，如果报了两条，则后面的那条应该把前面那条覆盖掉。

LC300 未定义。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	
idDevice	int unsigned	Device.idDevice	
MsgTime	timestamp		
Item	varchar(450)	报警阈值的一项	StatusID,value,msg (若非报警，msg 留空)

3.20 TaskTodo

待处理的用户操作。

服务器进程每隔一段时间在这里取任务，分别处理。

字段名	类型	含义	备注
id	bigint unsigned	自增 ID	当流水号
Time	datetime		
idUser	int unsigned	User.idUser	
Command	smallint unsigned	操作类型	
Parameters	varchar(1024)	操作的参数	

字段名	类型	含义	备注
MD5	varchar(40)	命令的 MD5 散列值	md5(Time + idUser + Command + Parameters) 作为查询条件供前段程序在 TaskDone 中查询任务完成状态

注意：这个表使用 MyISAM 引擎存储。自增的 id 字段如果使用 InnoDB 存储，计数器在内存里，数据库服务器一旦重启，又将从 1 开始计数，完成任务后向 TaskDone 中插入记录会引发冲突导致失败。

3.21 TaskDone

记录那些已完成的用户操作。

TaskTodo 中的任务，处理完毕后，从 TaskTodo 删除，将结果插入到表 TaskDone，保持 TaskTodo 中字段信息不变。

除了 TaskTodo 的字段，另有以下字段：

字段名	类型	含义	备注
FinishedTime	timestamp	完成时间	插入记录时由数据库自动生成
ReturnCode	smallint	返回代码	成功: 0; 失败: 控制器返回的确认码 bitwise-OR; 控制器未能返回结果（如超时）: -2。
ErrorMsg	varchar(255)	错误描述信息	UTF8

3.22 Firmware

字段名	类型	含义	备注
idFirmware	int unsigned		
Firmware	mediumblob	二进制内容	
FirmwareMD5	char(32)	md5(Firmware)	
FirmwareName	varchar(64)	固件包名称	终端用
FirmwareVersion	binary(4)	固件版本号	终端用
Note	varchar(300)		UTF8

MediumBlob 可以存储 16MB 大小的固件，MySQL 服务器也已经设置：

max_allowed_packet=16M

3.23 FrontendMapBmp

字段名	类型	含义	备注
idFrontendMapBmp	tinyint unsigned		
Name	varchar(64)		
DisplayOrder	tinyint unsigned	前台程序界面上显示地图的顺序	不宜重复

3.24 FrontendGroup

对于 LCP-SH-D 和 LC300 协议：现在用户的组和灯的组是统一的，即用户的第一个组就是所有灯的第一个组；对于 CSA 来说，一个用户的组可能对应控制器 1 的网关 ID1 和控制器 2 的网关 ID2，所以还要建一张表来指定用户组和控制器的网关 ID 的关系 (有点像以前 LC200)。

字段名	类型	含义	备注
idFrontendGroup	tinyint unsigned		
Name	varchar(64)		
Color	tinyint unsigned		Config 程序里作为选项

3.25 FrontendScene

这是用户定义的场景。

字段名	类型	含义	备注
idFrontendScene	tinyint unsigned		
Name	varchar(64)		
DisplayOrder	tinyint unsigned		不宜重复

3.26 FrontendControllerDevices

描述灯属于哪个控制器。

对于 CSA 的协议，灯是属于控制器下的某个网关 ID，所以可能会增加一个字段表示灯属于控制器的哪个网关。

字段名	类型	含义	备注
idController	int unsigned	Controller.idController	PK1
idDevice	int unsigned	Device.idDevice	PK2
DeviceDisplayOrder	int unsigned	界面显示该控制器下的灯时的序号	
PortOnController	smallint unsigned	灯在控制器的哪个口上	可能不需要

3.27 FrontendGroupDevices

描述灯属于哪个组。

字段名	类型	含义	备注
idFrontendGroup	tinyint unsigned	FrontendGroup. idFrontendGroup	PK1
idDevice	int unsigned	Device.idDevice	PK2
DeviceDisplayOrder	int unsigned	界面显示该组内的灯时的序号	不宜重复

3.28 FrontendDeviceMap

描述各种设备在 BMP 地图上的信息。

字段名	类型	含义	备注
idFrontendMapBmp	int unsigned	FrontendMapBmp. idFrontendMapBmp	PK1
DeviceType	tinyint unsigned	表明设备类型，如控制器、灯等	PK2
idDevice	int unsigned	Device.idDevice or Controller.idController	PK3
PosX	int	设备在地图上的横坐标	
PosY	int	设备在地图上的纵坐标	
idIcon	tinyint unsigned		对应的图标文件必须存在

4 附录：表 TaskTodo 的任务类型及参数

参数之间用英文的逗号，隔开；项与项之间用 \n 分开。

Id	Command	Parameters	备注
0	密钥更新		
1	重启控制器	控制器地址（8 字节 MAC）	
2	控制器时间同步	控制器地址	如果控制器地址（8 字节 MAC）为广播地址的话则所有控制器都要时间同步
3	设置控制器调光模式	控制器地址, 模式	模式 =0 远程调光，调光指令由中央控制器发送；模式 =1 自动调光，集中控制器根据预先设置的时控计划进入时控模式
4	设置控制器时控调光计划	控制器地址, 终端地址（8 字节 MAC），开始月, 开始日, 结束月, 结束日, 时, 分, 秒, 调光模式, 调光值 \n	调光模式 =0 表示后面是直接亮度值，=1 表示后面是场景值
5	清空控制器时控调光计划	控制器地址	
6	设置控制器自动调光计划		
7	清空控制器自动调光计划	控制器地址	
8	下载手机号码到控制器		
9	查询控制器手机号	控制器地址	
10	删除控制器中所有手机号	控制器地址	
11	查询控制器内短信数量	控制器地址	
12	查询控制器某条短信内容		
13	删除控制器某条短信内容		
14	查询控制器状态	控制器地址	上海：命令 0x0080；LC300：是多条命令的组合。
15	查询控制器时控调光计划	控制器地址	
16	查询控制器自动调光计划	控制器地址	
17	查询控制器日志	控制器地址	
18	升级控制器固件	控制器地址,FirmwareMD5	控制器进入固件升级模式，固件升级，退出固件升级模式
19	设置控制器所有属性		
20	初始化终端信息	控制器地址	
21	查询终端初始化信息		上海没有
22	删除终端信息	控制器地址	删除控制器下所有的终端

Id	Command	Parameters	备注
23	终端调光	控制器地址, 终端地址, 亮度值 \n	这个可能主要用于单灯调光。控制器地址（8 字节 MAC）为广播地址的话是向所有控制器发；终端地址（8 字节 MAC）是广播地址的话是向控制器下的所有终端发、终端地址（8 字节 MAC）是组地址的话是向控制器下这个组的终端发、终端地址（8 字节 MAC）也可以是单个终端的地址
24	初始化终端亮度	终端地址, 调光渐变时间, 上电亮度值, 最大亮度值, 最小亮度值, 故障亮度值, 调光系数	25-30 的批处理
25	设置终端调光渐变时间	终端地址, 调光渐变时间	
26	设置终端上电亮度值	终端地址, 上电亮度值	
27	设置终端最大亮度值	终端地址, 最大亮度值	
28	设置终端最小亮度值	终端地址, 最小亮度值	
29	设置终端故障亮度值	终端地址, 故障亮度值	
30	设置终端调光系数	终端地址, 调光系数	
31	终端恢复出厂设置	终端地址	
32	终端组设置	终端地址, 组号 \n	LC300: 组 0 不用, 就用组 1-63
33	终端组删除	终端地址	删除终端下的所有组
34	终端场景设置	终端地址, 场景号, 亮度值 \n	
35	终端场景删除	终端地址	删除终端下的所有场景
36	设置终端自动调光计划		
37	删除终端自动调光计划	终端地址	
38	设置终端信息上报条件		
39	设置终端报警阈值条件		
40	查询终端状态量终端地址	终端地址 \n	
41	查询终端自动调光计划	终端地址	
42	查询终端组号	终端地址	
43	查询终端场景	终端地址	
44	升级终端固件	终端地址,FirmwareMD5	通信线程按流程走
45	清零终端总工作时间	终端地址	
46	清零终端总用电量	终端地址	
47	设置终端过温保护参数		
48	开启关闭终端过温保护		LCP-SH-D 没有