



通用控制协议

缴费机控制协议

(机密等级：1)

机密等级：

- 0 (公开)
- 1 (可提供客户和集成商)
- 2 (仅限于公司内部)
- 3 (仅限于公司内部指定人员)

苏州德亚交通技术有限公司

文件号：SP23d

版本：V1.1

日期：2021/03/06

制定：戴海明

修改记录：

2021/03/06: 初版

- 1) 初版

2021/05/13: 新增 IO 功能

- 1) 增加 IO 输出功能
- 2) 增加 IO 输入变化上报

目 录

1 产品说明.....	2
1.1 功能概要.....	2
2 系统通讯接口.....	3
3 控制协议帧格式.....	4
3.1.1 下行帧格式.....	4
3.1.2 应答帧格式.....	4
3.1.3 系统心跳帧.....	5
3.1.4 系统控制帧.....	5
3.1.5 系统时间设置帧.....	6
3.1.6 LED 屏控制.....	6
清除屏幕.....	6
行模式下的文字显示.....	6
3.1.7 二维码信息上报.....	7
3.1.8 语音播报.....	7
3.1.9 LCD 屏控制.....	8
3.1.10 IO 输出控制.....	8
3.1.11 输入变化上报.....	9

1 产品说明

1.1 功能概要

缴费机有两种类型，根据显示实现具体可以分出 LED 模组版本与 LCD 模组两种版本。版本分别为 TMPE32A、TMPE32B、TMP102。各缴费机由各核心板控制器，LCD 控制器、语音合成板、扫码模组等构成。由于历史原因，设备不同又发展出了不同的对接方式。此协议定制为 TMPE32A，TMP102 两种版本设备使用，既包含核心板控制器的对接协议，在核心板控制器上，单走网口有现成可以用的网口对接动态库，既 RWPAY.dll，但是此库不支持串口协议，但是新出现的需求需要设备具有串口通讯协议，故而制定了此协议标准，考虑到 TMP32B 的对接方式使用 TFI 费显控制协议，因而此协议的制定兼容了 TFI 协议，增加了 LCD 屏控制协议，增加了语音播放器的控制协议。此协议的将会实现串口和网口两部分功能，我司提供协议说明文件，和动态库交互协议，客户可以根据需求使用不同方式实现。

型号	TMPE32A	TMPE32B	TMP102	
核心板控制器	√	×	√	
LED 控制器	√	√	×	
LCD 控制器	×	×	√	
语音合成器	√	√	√	
扫码模块	√	√	√	

2 系统通讯接口

串口

系统与车道计算机采用 RS232 通讯时，串口设置为：

- 波特率：9600 bps
- 数据位：8 位
- 停止位：1 位
- 奇偶校验：无

网口

使用网口通讯时，网口为 Ethernet，使用 TCP 协议。

- 接口：RJ45
- 端口：8016
- Address Family: IPv4
- Interface: Ethernet

不管使用串口或是网口，应用层的协议是一样的。当使用网口时，一个包一个命令帧。不可以一个包里面放几个命令帧。

每个帧的最大长度必须小于 1024，大于 1024 的帧长度会被处理器抛弃，并返回错误码。

3 控制协议帧格式

3.1.1 下行帧格式

LED 显示屏部分的通信协议帧格式为：

SOH	CMD	LEN	PARAM	CRC8
-----	-----	-----	-------	------

其中：

- SOH

为帧起始字段，为 4 个字节整数常量，发送顺序为：0xA0, 0xB0, 0xC0, 0xD0。在小端字节序的 CPU，这个整数常量的定义值为 0xD0C0B0A0。
- CMD

为命令字段，单字节。
- LEN

为后续的 PARAM 字段的长度(字节数)的 BCD 编码。LEN 字段为 2 个字节，例如 PARAM 长度是 132 字节，LEN 的字段内容为 0x01 0x32。因为整个包最大长度为 1024，PARAM 最大只能是 $1024-5 = 1019$ 。
- PARAM

命令的参数部分。可以是空字段（LEN 是 0）。
- CRC8

由 SOH 开始一直到 PARAM 的每个字节的亦或和。整个包做亦或和运算，最后应该获得 0。

3.1.2 应答帧格式

LED 显示屏控制器收到命令后，应该在 100 毫秒内返回应答。应答帧的格式和下行命令帧完全相同，但是有着不同的 SOH 字段。

SOH	CMD	LEN	PARAM	CRC8
-----	-----	-----	-------	------

SOH	上行应答帧的起始字段，4 个字节。内容依照发送顺序为 0xA1, 0xB1, 0xC1, 0xD1。
CMD	这是要应答的下行帧的 CMD 字段复制过来，另外最高位（0x80）如果置位，就表示命令失败。也就是说，这个应答帧的（CMD & 0x7f）应该等于它要应答的下行帧的 CMD 字段。
LEN	和下行帧一样，是 PARAM 字段长度的 BCD 编码，共 2 个字节。
PARAM	应答的参数，如果是命令失败，这里可以放失败的原因说明，或是错误码。
CRC8	和下行帧一样，由 SOH 到 PARAM 的所有字节的亦或和运算结果。

3.1.3 系统心跳帧

CMD 字段：0x00

PARAM 字段：无

PC 端可以根据心跳帧的回复状态判定设备是否离线，若未网口通讯，设备端在 30 秒内收不到客户端的心跳请求，会主动断开此连接，因而建议客户每 10 秒发送一次心跳帧请求。

3.1.4 系统控制帧

CMD 字段：0x10

PARAM 字段：

控制类型：1 个字节

- ◆ 0x01 禁用扫码信息上报
- ◆ 0x02 使能扫码信息上报
- ◆ 0x03 系统重启

3.1.5 系统时间设置帧

CMD 字段: 0x01

PARAM 字段

设置的时间 ASCII 字符串，时间格式为 yyyymmddHHMMSS，一共 14 个字节。例如 "20210304121212" 代表设置的时间为 2021 年 3 月 4 日，12 点 12 分 12 秒。

3.1.6 LED 屏控制

清除屏幕

CMD 字段: 0x04

PARAM 字段: 无

行模式下的文字显示

CMD 字段: 0x20

PARAM 字段: 由以下字节组成:

- ◆ ROW: 显示在第几行【取值范围 1~最大可显示行数】。
- ◆ FORMAT: 显示格式，可以支持以下格式
【bit0~bit3】文字显示模式
0 左对齐，行尾多出的字会砍掉，少的字补空白。
1 居中对齐，两头多出的字砍掉，少的字两端补空白。
2 右对齐，左边多出的字砍掉，少的字补空白。

- 3 由右向左边滚动到最左边:第一个字碰到左边界停止。
- 4 由左向右边滚动到最右边:最后一个字碰到右边界停止。

5 由右向左边滚动循环:最后一个字消失后,又重头开始

6 由左向右边滚动循环:第一个字消失后,又重头开始

【bit4~bit5】 闪烁速度 (取值范围 0~3)

适用于显示模式 0~4。设置闪烁后,该行文字以设置的速度闪烁。

【bit6~bit7】 滚动速度 (取值范围 0~3)

适用于显示模式第 5, 6 两种。这两个模式允许文字总长度大于可显示宽度。数值越大滚动速度越快。

- ◆ COLOR : 3 个字节, 同 3.7 里的 COLOR 定义。
- ◆ CONTENT : 要显示的文字的码。中文使用 GBK 码, 英文数字使用 ASCII 码 (一个字符的宽度是行高的一半)。全宽的英文和数字符号等 (一个字的宽度和行高一样) 使用对应的 GBK 码。不需要结尾字符, 因为由帧的 LEN 字段值减去 5 就是 CONTENT 的长度。

3.1.7 二维码信息上报

CMD 字段: 0x40

PARAM 字段: 由以下字节组成

- ◆ INDEX : 第几个扫码模组的结果, 1-3
- ◆ 二维码上报信息内容

3.1.8 语音播报

CMD 字段: 0x50

PARAM 字段: 由以下字节组成

- ◆ VOL : 播放音量, 0x01-0x08
- ◆ CONTENET: 播放内容, GBK 编码

3.1.9 LCD 屏控制

CMD 字段: 0x60

PARAM 字段: 由以下字节组成:

- ◆ INDEX : (1 个字节) 切换显示第几页场景, 每个场景对应一个静态图片
- ◆ CONTENT : (n 个字节) 场景模式下需要显示的内容信息, 视应用场景不同, 此从字段内容可由客户自定义, 中文编码格式等信息也有客户指定, 控制器只做数据的转发, 不做任何数据的处理工作。此字段可以存放 json、键值对, 自定义字符串等格式数据。

默认场景列表

Index	场景	CONTENT
0	欢迎页面	无
2	支付页面	车辆扣费信息, 支付二维码等
3	支付失败	无
4	支付成功	无

3.1.10 IO 输出控制

CMD 字段: 0x70

PARAM 字段: 由以下字节组成

- ◆ TYPE: (1 字节), 0x00 IO 输出控制, 0x01 正向脉冲输出, 0x02 反向脉冲输出

-
- ◆ PIN : (1 字节) 第几个管脚 (1-8)
 - ◆ PARAM: (4 字节) 输出模式时 , 表示管脚状态 , 0 表示断开 , 非 0 表示闭合。脉冲模式时 , 表示脉冲时间 , 单位毫秒 , 小端模式

3.1.11 输入变化上报

CMD 字段: 0x80

PARAM 字段: 由以下字节组成

- ◆ PARAM: (4 字节) , 当前 IO 映射到 PARAM , 一个 bit 表示一个输入管脚