

MOTINOVA E-Bike 电气系统通信协议

[填入文件编号]

编制: 周雄

审核: _____

批准: _____

武汉天腾动力科技有限公司

二〇二二年十一月九日

目录

1 总则.....	5
2 目的.....	5
3 适用范围.....	5
4 修订记录.....	5
MOTINOVA 小牙盘系列中置系统通信协议.....	6
1 系统组成.....	6
2 硬件接口.....	6
2.1 通信接口参考电路.....	6
2.2 注意事项.....	7
3 通信协议规则.....	7
3.1 软件配置.....	7
3.2 数据帧封装格式.....	7
3.2.1 数据帧格式.....	7
3.2.2 ID 分配.....	8
3.2.3 分包方式.....	8
3.3 CDL 数据格式转换规则.....	8
3.3.1 接收处理.....	8
3.3.2 发送处理.....	9
3.3.3 CDL 专用指令.....	9
4 通信内容.....	10
4.1 MC 命令字定义.....	10
4.2 BMS 命令字定义.....	18
4.3 PBU/OBC/ECU 命令字定义.....	22
4.4 HMI 命令字定义.....	27
4.5 CDL 命令字定义.....	30
5 附录 1: CRC32 计算方法.....	40
5.1 CRC32 计算多项式表.....	40
5.2 CRC32 计算方法.....	41
6 附录 2: 警告/故障码列表.....	42
7 附录 3: 升级协议及流程.....	44
7.1 升级流程图.....	44
7.2 升级协议.....	44
8 附录 4: PBU 开机密码工作流程图.....	47
9 附录 5: MC 故障日志数据结构定义.....	48
9.1 目的.....	48
9.2 方法.....	48
9.3 协议.....	48
9.4 数据结构定义.....	49
10 附录 6: BMS 故障日志数据结构定义.....	51
10.1 目的.....	51
10.2 方法.....	51

10.3 协议.....	51
10.4 数据结构定义.....	52
MOTINOVA 大牙盘系列中置系统通信协议.....	54
1 系统组成.....	54
2 硬件接口.....	54
2.1 通信接口参考电路.....	54
2.2 注意事项.....	54
3 通信协议规则.....	55
3.1 软件配置.....	55
3.2 数据帧封装格式.....	55
3.2.1 数据帧格式.....	55
3.2.2 ID 分配.....	56
3.2.3 分包方式.....	56
3.3 CDL 数据格式转换规则.....	56
3.3.1 接收处理.....	56
3.3.2 发送处理.....	56
3.3.3 CDL 专用指令.....	56
4 通信内容.....	58
4.1 MC 命令字定义.....	58
4.2 BMS 命令字定义.....	67
4.3 OBC 命令字定义.....	70
4.4 CDL 命令字定义.....	73
5 附录 1: CRC32 计算方法.....	82
5.1 CRC32 计算多项式表.....	82
5.2 CRC32 计算方法.....	83
6 附录 2: 警告/故障码列表.....	84
7 附录 3: 升级协议及流程.....	86
7.1 升级流程图.....	86
7.2 升级协议.....	86
8 附录 4: MC 故障日志数据结构定义.....	88
8.1 目的.....	88
8.2 方法.....	88
8.3 协议.....	88
8.4 数据结构定义.....	89
9 附录 5: BMS 故障日志数据结构定义.....	91
9.1 目的.....	91
9.2 方法.....	91
9.3 协议.....	91
9.4 数据结构定义.....	92
MOTINOVA 轮毂控制器系统通信协议.....	94
1 系统组成.....	94
2 硬件接口.....	94
2.1 通信接口参考电路.....	94
2.2 注意事项.....	94

3 通信协议规则.....	95
3.1 软件配置.....	95
3.2 数据帧封装格式.....	95
3.2.1 数据帧格式.....	95
3.2.2 ID 分配.....	96
3.2.3 分包方式.....	96
3.3 CDL 数据格式转换规则.....	96
3.3.1 接收处理.....	96
3.3.2 发送处理.....	96
3.3.3 CDL 专用指令.....	96
4 通信内容.....	98
4.1 MC 命令字定义.....	98
4.2 BMS 命令字定义.....	107
4.3 OBC 命令字定义.....	110
4.4 CDL 命令字定义.....	113
5 附录 1: CRC32 计算方法.....	122
5.1 CRC32 计算多项式表.....	122
5.2 CRC32 计算方法.....	123
6 附录 2: 警告/故障码列表.....	124
7 附录 3: 升级协议及流程.....	126
7.1 升级流程图.....	126
7.2 升级协议.....	126
8 附录 4: MC 故障日志数据结构定义.....	128
8.1 目的.....	128
8.2 方法.....	128
8.3 协议.....	128
8.4 数据结构定义.....	129
9 附录 5: BMS 故障日志数据结构定义.....	131
9.1 目的.....	131
9.2 方法.....	131
9.3 协议.....	131
9.4 数据结构定义.....	132

1 总则

系统通信总线采用 CAN2.0A 协议标准帧格式进行数据传输, 本文件制定了各设备 ID、报文传输格式、以及数据协议。

2 目的

提高 MOTINOVA E-Bike 电气系统各设备之间数据通信的信号稳定性、格式正确性及信息完整性。

3 适用范围

本文件适用于 MOTINOVA 小牙盘中置系统、MOTINOVA 大牙盘中置系统、MOTINOVA 轮毂电机控制系统的软件开发, 以及智能电池保护板、各类仪表的功能开发。

4 修订记录

表1 修订记录

日期	修订人	修订内容	版本
20221023	周雄	基于《MOTINOVA 中置驱动系统通信协议_V3r4_20220512》, 增加大牙盘中置系统和轮毂控制系统的相关内容。	V4.0
20221030	周雄	1、大牙盘和轮毂控制器协议 MC 发送给 OBC 数据中增加自动关机时间; 2、大牙盘和轮毂控制器协议整车参数增加一预留位; 3、大牙盘 OBC 类仪表增加按键状态发送给电机指令。	V4.1
20221109	周雄	增加 BMS 故障日志的定义和读取指令	V4.2

MOTINOVA 小牙盘系列中置系统通信协议

1 系统组成

MC: 电机控制器 Motor Controller

BMS: 电池管理系统 Battery Management System

PBU: 按键单元 Push Button Unit

HMI: 显示单元 Human Machine Interface

OBC: 车载计算机 On Board Computer

CDL: 通讯适配器 CAN Dongle

APP: 用户程序 Application

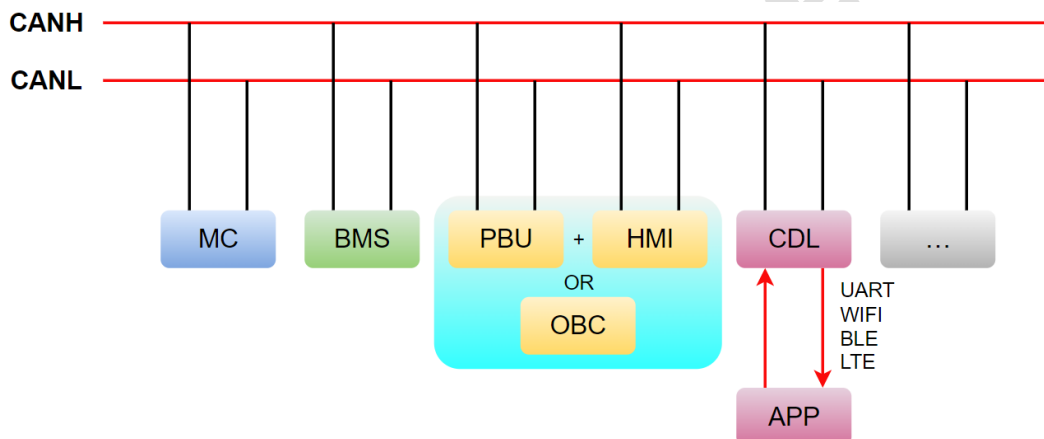


图1 系统通信接口示意图

2 硬件接口

2.1 通信接口参考电路

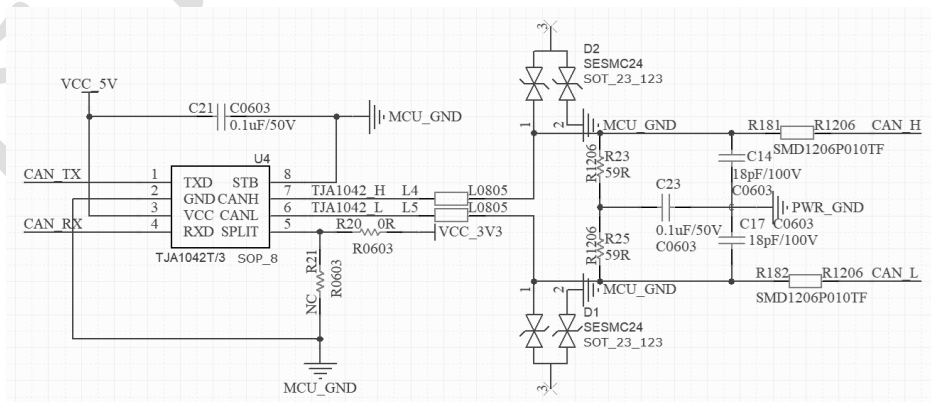


图2 通信接口参考电路

2.2 注意事项

图 2 展示的为各组件 CAN 通信接口参考电路, 设计时注意以下几点:

- 1) ESD 保护电路设计需考虑电源线与通信线发生短路时, 不会导致 ESD 保护器件被击穿, 电源线最高电压按照 48V 系统最高工作电压 54.6V 测试;
- 2) 终端匹配电阻设计在 BMS 和 PBU/OBC 内部, 其它部件预留;
- 3) BMS 需考虑通信接口电气隔离设计。

3 通信协议规则

3.1 软件配置

CAN 控制器推荐配置如下:

时钟频率: 750 KHz/1500 KHz

SJW: 1

BS1: 4

BS2: 1

波特率: 125 Kbps/250 Kbps

3.2 数据帧封装格式

所有传输的数据按照标准帧进行封装, 将传输数据按顺序填入数据帧中。以下对数据帧格式、ID 分配、分包方式进行介绍。

3.2.1 数据帧格式

协议描述了每帧数据内容, 包括帧头、帧模式、命令段长度、命令字、数据段、校验位、帧尾。每帧格式如下:

表2 数据帧格式

帧头	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中:

- 1) 帧头固定为 0x55 0xAA, 帧尾固定为 0xF0;
- 2) 帧模式包含读 0x11, 写 0x16, 和上报 0x0C, 任何设备收到写指令时, 需根据数据来源发送通用反馈指令;
- 3) LENGTH 命令段总长度, 占用 1 字节, 有效值为 0x02~0xFF;
- 4) COMMAND 为命令字, 占用 2 个字节, 第 1 字节为命令字序号, 第 2 字节为数据段长度;

- 5) DATA 为数据段, 长度为 LENGTH - 2;
- 6) CRC 为校验位, 占用 4 字节, 由帧头开始, CAN_ID 插入到帧头和帧模式之间, 计算到数据段最后一个字节, 计算方法见附录 1, 计算结果高字节在前, 如: CAN_ID 为 0x0712, 数据帧为 55 AA 11 03 22 01 00 CRC1 CRC2 CRC3 CRC4 F0, CRC 计算函数输入数据为 55 AA 07 12 11 03 22 01 00, 计算结果依次由高到低写入 CRC1、CRC2、CRC3、CRC4;
- 7) 数据段发送时, 采用小端模式。

3.2.2 ID 分配

表3 ID 分配

MC	Target	广播	MC	BMS	PBU/OBC/ECU	HMI	CDL
	CAN ID	0x710		0x712	0x713	0x714	0x715
BMS	Target	广播	MC	BMS	PBU	HMI	CDL
	CAN ID	0x720	0x721		0x723	0x724	0x725
PBU/OBC/ ECU	Target	广播	MC	BMS	PBU	HMI	CDL
	CAN ID	0x730	0x731	0x732		0x734	0x735
HMI	Target	广播	MC	BMS	PBU	HMI	CDL
	CAN ID	0x740	0x741	0x742	0x743		0x745
CDL	Target	广播	MC	BMS	PBU	HMI	CDL
	CAN ID	0x750	0x751	0x752	0x753	0x754	

3.2.3 分包方式

对于长度超过 8bytes 的数据帧, 按照 8+N 的方式分包, 每个数据包填入相同的 ID 号, 如下表所示:

表4 封装方式

包序号	1				N	
内容	ID	Byte1~Byte8	ID	Byte1~Byte8	ID	Byte1~ByteN

3.3 CDL 数据格式转换规则

3.3.1 接收处理

CDL 接收到 CAN 总线数据后, 按照下表格式插入 ID, 完成转发:

表5 CDL 数据转换格式

帧头	CAN ID	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	ID	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中 ID 占用 2 字节。

3.3.2 发送处理

CDL 接收到其它设备按照上表格式发送的数据后,取数据包中的 ID 作为当前发送数据帧的 CAN ID,并以表 1 格式完成分包转发。

3.3.3 CDL 专用指令

用于 CDL 与 APP 之间的数据交互,此部分指令不需要通过 CDL 发送到 CAN 总线。

3.3.3.1 CDL 在线检测

在其它 UART、WIFI、BLE 等接口方式的设备 APP 通过 CDL 访问 CAN 总线上 MC、BMS、PBU、HMI、OBC 等设备时,需要先确认 CDL 连接是否正常。

1) APP 定时 200ms 发送,收到 CDL 返回或超时 2s 停止发送,收到 CDL 返回表示 CDL 在线,超时 2s 未收到 CDL 返回表示 CDL 离线:

[55 AA 07 FF 11 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 返回:

[55 AA 07 FF 0C 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.2 设备开关机

1) APP 定时 200ms 发送,收到 CDL 返回或超时 2s 停止发送,并显示开机结果:

[55 AA 07 FF 16 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

2) CDL 接收到指令后使能开机信号,同时返回:

[55 AA 07 FF 0C 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

3.3.3.3 上报系统供电方式

CDL 定时 2s 发送一次系统供电方式,APP 显示供电方式,数据如下:

[55 AA 07 FF 0C 03 33 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下:

0x00: 适配器未接入、外部设备无电池

0x01: 适配器未接入、外部设备含电池

0x11: 适配器接入、外部设备含电池

0x10: 适配器接入、外部设备无电池

3.3.3.4 CDL 复位

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 并显示复位结果:

[55 AA 07 FF 11 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 收到指令后返回指令, 然后复位:

[55 AA 07 FF 0C 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.5 设备校验

通过其它 UART、WIFI、BLE 等接口方式的设备 APP 连接 CDL, 需要对设备与 CDL 进行双向校验。

1) APP 发送随机码和校验密钥给到 CDL:

[55 AA 07 FF 16 16 55 14 随机码 密钥 CRC1 CRC2 CRC3 CRC4 F0]

其中随机码长度为 12Bytes, 密钥长度为 8Bytes。

2) CDL 根据随机码和密钥进行计算后, 将校验码发送给 APP, 由 APP 判断校验是否成功:

[55 AA 07 FF 0C 1D 55 1B 校验结果 版本号 CRC1 CRC2 CRC3 CRC4 F0]

其中校验结果长度为 12Bytes, 版本号为 15Bytes, 格式为 Vx.x.x_YYYYMMDD。

4 通信内容

4.1 MC 命令字定义

表6 MC 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x710	0x0C	0x1020	MC 运行信息 (收到 PBU/OBC 控制指令时返回)	车速:2bytes 输出转速:2bytes 电功率:2bytes 母线电压:2bytes 母线电流:2bytes 踏板:1byte 踩踏力矩:1byte 踩踏方向:1byte 助力档位:1byte	0.1km/h 1rpm 1W 1mV 1mA 1rpm 1N.m 0-正,1-反,2-停止 0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式

				大灯状态:1byte 剩余电量:1byte 续航里程:2bytes ODO 里程:2bytes 平均功耗:1byte PCB 温度:1bytes 绕组温度:1bytes MCU 温度:1bytes 开机后骑行里程:2bytes 开机后骑行时间:2byte 传感器 1:1byte 传感器 2:1byte 传感器 3:1byte 预留:1byte	0xF0-关, 0xF1-开 1%, 无效为 0xFF 1km, 无效为 0xFFFF 1km 0.01Ah/km, 无效为 0xFF +40℃ +40℃ +40℃ 0.1km 1s 适用三力矩机型 适用三力矩机型 适用三力矩机型
0x710	0x0C	0x1104	MC 故障码 (存在故障时 200ms 自动发 送, 故障消失后 停止发送)	0:无故障 低 16 位: 0x0001:过流保护 0x0002:低压保护 0x0004:过压保护 0x0008:堵转保护 0x0010:过热保护 0x0020:SPS 故障 0x0040:TQS 故障 0x0080:霍尔故障 0x0100:马达缺相 0x0200:NTC 故障 0x0400:BMS 校验失败 0x0800:HMI 校验失败 0x1000:PBU/OBC 校验失败 0x2000:MCU 故障 0x4000:踏频故障 0x8000:指拨故障 高 16 位: 0x0001:MOS 短路 0x0002:电压异常 0x0004:电路故障 0x0008:TE 故障 0x0010:TE 电路故	按位或输出, 0-正常, 1-故障, 未描述的故障位填 0

				障 0x0020:MC 校验失败 0x0040: 马达失速故障 0x0080: 预留故障 3	
0x710	0x0C	0x1240	电机版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 ' . ', 无效填充 0x20 FW 命名格式为 Vxrxrx_YYYYMMDD
0x710	0x0C	0x1305	关机就绪 (返回指令)	ASCII 字符	READY
0x710	0x0C	0x1401	在线检测结果 (返回指令)	0x00: 系统正常 0x01: BMS 离线 0x02: PBU 离线 0x04: HMI 离线 0x08: 预留 0x10: 预留 0x20: 预留 0x40: 预留 0x80: 预留	按位或输出, 0-在线, 1-离线
0x710	0x0C	0x1510	骑行历史信息 (返回指令)	ODO 里程: 4bytes ODO 时间: 4bytes TRIP 里程: 4bytes TRIP 时间: 4bytes	0.1km 1min 0.1km 1min
0x710	0x0C	0x1608	角度传感器原始数值 (返回指令)	俯仰角绝对数值: 2bytes 横滚角绝对数值: 2bytes 俯仰角相对数值: 2bytes 横滚角相对数值: 2bytes	0.1°, ±1800 0.1°, ±1800 0.1°, ±1800 0.1°, ±1800
0x710	0x0C	0x1720	特殊性说明	ASCII 字符	结束符为 ' . ', 无效填充 0x20
发送给 BMS					
0x712	0x11	0x3009	BMS 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE

0x712	0x11	0x3100	查询 BMS 物理 ID (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3200	查询 BMS 校验码 (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3300	查询 BMS 设计信息 (主动发送, 收到返回或超时停止)		
发送给 PBU/OBC/ECU					
0x713	0x11	0x5009	PBU/OBC 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x713	0x11	0x5100	查询 PBU/OBC 物理 ID (主动发送, 收到返回或超时停止)		
0x713	0x11	0x5200	查询 PBU/OBC 校验码 (主动发送, 收到返回或超时停止)		
0x713	0x0C	0x5303	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x713	0x0C	0x5408	返回 OBC 用户参数	轮胎默认周长:1byte 启动模式:1byte 限速:1byte 周长微调:1byte 预留:4bytes	1cm 1-柔和, 2-正常, 3-强劲 1km/h ±10cm, 有符型填充 0x00
0x713	0x0C	0x5410	返回 ECU 用户参数	轮胎周长:1byte 限速 1:1byte 低压保护:2bytes 预留:12bytes	1cm km/h mV 填充 0x00
发送给 HMI					

0x714	0x11	0x7009	HMI 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x714	0x11	0x7100	查询 HMI 物理 ID (主动发送, 收到返回或超时停止)		
0x714	0x11	0x7200	查询 HMI 校验码 (主动发送, 收到返回或超时停止)		
0x714	0x0C	0x7308	配置参数查询结果 (返回指令)	轮胎默认周长:1byte 启动模式:1byte 限速:1byte 周长微调:1byte 预留:4bytes	1cm 1-柔和, 2-正常, 3-强劲 1km/h ±10cm, 有符型填充 0x00
0x714	0x0C	0x7403	通用反馈指令 (返回指令)	ASCII 字符	ACK
发送给 CDL					
0x715	0x0C	0xA020	用户参数 1 (返回指令)	指拨模式:1byte 启动模式:1byte 停机时间:2bytes 限速值:1byte 下降速度:1byte 前飞:1byte 后飞:1byte 限流:1byte 温度预警:1byte 温度保护:1byte 无码表支持:1byte 轮胎周长:1byte 电机系列号:1byte ECO 助力比增益:1byte ECO 加速度增益:1byte	0x55-不支持 0xAA-支持 1-柔和, 2-正常, 3-强劲 1ms 1km/h 1~5 个等级 T T 1A +40℃ +40℃ 0x55-不支持 0xAA-支持 1cm 1~15 0.01 倍, 50~150 0.01 倍, 50~150

				NOMA 助力比增益:1byte	0.01 倍, 50~150
				NOMA 加速度增益:1byte	0.01 倍, 50~150
				SPORT 助力比增益:1byte	0.01 倍, 50~150
				SPORT 加速度增益:1byte	0.01 倍, 50~150
				TURBO 助力比增益:1byte	0.01 倍, 50~150
				TURBO 加速度增益:1byte	0.01 倍, 50~150
				SMART 助力比增益:1byte	0.01 倍, 50~150
				SMART 加速度增益:1byte	0.01 倍, 50~150
				速度检测信号个数:1byte	1~100
				踏频启动信号个数:1byte	1~100
				速度传感器信号来源:1byte	0x55-车轮触发 0xAA-踏频估算 0xEE-通讯获取
				周长微调值:1byte	1cm, 有符型
				低压保护阈值:2bytes	mV
				推行限速:1byte	0.1km/h
				推行马达转速:1byte	1rpm
0x715	0x0C	0xA110	马达参数 (返回指令)	额定功率:2bytes	1W
				额定转速:2bytes	1rpm
				定子电阻:2bytes	1mOhm
				定子 Lq:2bytes	1mH
				定子 Ld:2bytes	1mH
				反电动势:2bytes	1mV
				额定电压:1bytes	1V
				预留:3bytes	填充 0x00
0x715	0x0C	0xA230	运行历史 1 (返回指令)	开机次数:4bytes	次
				使用时间:4bytes	1min
				过流保护:2bytes	次
				低压保护:2bytes	次
				过压保护:2bytes	次
				堵转保护:2bytes	次
				过热保护:2bytes	次

				SPS 故障:2bytes TQS 故障:2bytes 霍尔故障:2bytes 马达缺相:2bytes NTC 故障:2bytes BMS 校验失败:2bytes HMI 校验失败:2bytes PBU 校验失败:2bytes PCB 最高温度:1byte PCB 最低温度:1byte 预留:4bytes 绕组最高温度:1byte 绕组最低温度:1byte MCU 最高温度:1byte MCU 最低温度:1byte 预留:4bytes	次 次 次 次 次 次 次 次 +40℃ +40℃ 填充 0x00 +40℃ +40℃ +40℃ +40℃ +40℃ 填充 0x00
0x715	0x0C	0xA318	力矩传感器零点值 (返回指令) [单力矩]	历史数值 1~历史数值 9:18bytes 最大值:2bytes 最小值:2bytes 当前值:2bytes	
0x715	0x0C	0xA408	校验码计算密钥 (返回指令)	密钥:8bytes	ASCII
0x715	0x0C	0xA520	生产信息 (返回指令)	生产商:8bytes 产地:8bytes 生产日期:8bytes 产品标识:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD MM_VP1.
0x715	0x0C	0xA610	自定义可存储字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA710	自定义可存储字符串 2	ASCII 字符	结束符为 0x2E, 无效填充 0x20

			(返回指令)		
0x715	0x0C	0xA810	自定义可存储字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA903	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x715	0x0C	0xAA04	力矩传感器校正信息 [单力矩]	启动值:2bytes 标定系数:1byte 预留:1byte	填充 0x00
0x715	0x0C	0xAB88	返回存储器指定起始和结束地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 填充 0xFF
0x715	0x0C	0xAC10	运行历史 2 (返回指令)	MCU 故障次数:2bytes 踏频故障次数:2bytes 指拨故障次数:2bytes MOS 短路次数:2bytes 电压异常次数:2bytes 电路异常次数:2bytes TE MCU 异常次数:2bytes TE 电路异常次数:2bytes	次 次 次 次 次 次 次 次
0x715	0x0C	0xAD20	用户参数 2 (返回指令)	俯仰角零度偏移:2bytes 横滚角零度偏移:2bytes 支持角度传感器标志:1byte 尾灯工作模式:1byte	0.1°, ±900 0.1°, ±900 0x55-不支持 0xAA-支持 0x01: 模式 1, 连接尾灯, 开灯时低亮, 刹车时高亮 0x02: 模式 2, 连接尾灯, 开灯时高亮, 刹车时闪烁 0x03: 模式 3, 连接刹车灯, 刹车时高亮 0x04: 模式 4, 连接刹

				前灯电压:1byte 尾灯电压:1byte 预留:24bytes	车灯, 刹车时闪烁 0x06: 6V 0x0C: 12V 0xFF: 随电池电压 0x06: 6V 0x0C: 12V 0xFF: 随电池电压 填充 0x00
0x715	0x0C	0xAE1A	力矩传感器零点值 (返回指令) [三力矩]	传感器序号:1byte 历史数值 1~历史 数值 9:18bytes 最大值:2bytes 最小值:2bytes 当前值:2bytes RS:1byte	针对采用 3 个力矩传感器的电机使用该协议 填充 0x00
0x715	0x0C	0xAF02	力矩传感器标定系数 [三力矩]	传感器序号:1byte 标定系数:1byte	针对采用 3 个力矩传感器的电机使用该协议
0x715	0x0C	0xB002	启动值	启动值:2bytes	
0x715	0x0C	0xB105	力矩传感器线性校正参数 [三力矩 J01 系列]	序号:1byte AD 采集值:2byte 力矩值:1byte 校正参数:1byte	1~15 N. m 放大 128 倍

4.2 BMS 命令字定义

表7 BMS 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x720	0x0C	0x1010	电池运行信息 (返回指令)	电压:2bytes 平均电流:2bytes 剩余容量:2bytes 满充容量:2bytes 电芯温度:1byte 剩余电量:1byte 运行状态:1byte (按位或输出)	1mV 1mA, 有符型, 放电为负, 充电为正 1mAh 1mAh +40℃ 0~100% 0x00: 休眠 0x01: 充电器接入 0x02: 预留 0x04: 预留 0x08: 预留 0x10: 预留 0x20: 预留 0x40: 预留

				SOH:1byte 预留:4bytes	0x80:预留 0~100% 填充 0x00
0x720	0x0C	0x1120	电芯电压 (返回指令)	Cell_1:2bytes Cell_16:2bytes 不足部分填充 0x00	1mV 1mV
0x720	0x0C	0x1204	BMS 故障码 (存在故障时 200ms 自动发 送,故障消失后 停止发送)	高 16 位: 0x0001:充电过压 警告 0x0002:放电低压 警告 0x0004:充电过流 警告 0x0008:放电过流 警告 0x0010:充电高温 警告 0x0020:充电低温 警告 0x0040:放电高温 警告 0x0080:放电低温 警告 0x0100:MOS 高温警 告 低 16 位: 0x0001:二级放电 过流保护 0x0002:充电过流 保护 0x0004:短路保护 0x0008:过放保护 0x0010:过充保护 0x0020:放电低温 保护 0x0040:放电高温 保护 0x0080:充电低温 保护 0x0100:充电高温 保护 0x0200:放电 MOS 故 障	按位或输出, 0-正 常, 1-故障

				0x0400: 充电 MOS 故障 0x0800: 温度传感器故障 0x1000: 预留 0x2000: 一级过流保护 0x4000: AFE 故障 0x8000: MCU 故障	
0x720	0x0C	0x1308	关机指令 (主动发送, 收到返回或超时停止)	ASCII 字符	常按开关键 3s 或监测到母线电流小于 50mA 且 CAN 总线空闲持续 30min 后, 执行发送 SHUTDOWN, 延时 1s 后, 关闭放电开关
0x720	0x0C	0x1410	电池设计信息 (返回指令)	设计容量: 2bytes 设计电压: 1byte 电芯型号: 8bytes 预留: 5bytes	1mAh 1V ASCII, 0x2E 结束, 无效填充 0x20 填充 0x00
0x720	0x0C	0x1540	电池版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20; FW 命名格式为 Vxrxrx_YYYYMMDD.
0x720	0x0C	0x160C	电池物理 ID (返回指令)	ID: 12bytes	不足位填充 1
0x720	0x0C	0x170C	电池校验码 (返回指令)	校验码: 12bytes	
发送给 MC					
0x721	0x0C	0x3005	在线检测反馈 (返回指令)	ASCII 字符	READY
发送给 CDL					
0x725	0x0C	0x5028	电池 BMS 历史信息 (返回指令)	电芯最高温: 1byte 电芯最低温: 1byte 最大放电电流: 2bytes 最大充电电流: 2bytes 循环次数: 2bytes 最近充电间隔时	+40℃ +40℃ 无符型, 单位 1mA 无符型, 单位 1mA 次 小时

				间:2bytes 最大充电间隔时间:2bytes 充电过流保护次数:2bytes 放电过流保护次数:2bytes 过充保护次数:2bytes 过放保护次数:2bytes 短路保护次数:2bytes 充电低温保护次数:2bytes 充电高温保护次数:2bytes 放电低温保护次数:2bytes 放电高温保护次数:2bytes 运行时间:4bytes SOH:1byte 预留:5bytes	小时 次 次 次 次 次 次 次 次 次 1min 0~100% 填充 0x00
0x725	0x0C	0x5120	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x725	0x0C	0x5210	自定义可存储 字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5310	自定义可存储 字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5410	自定义可存储 字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5503	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x725	0x0C	0x5688	存储器指定起始和结束地址	起始地址:4bytes 结束地址:4bytes	结束地址-起始地址 <128 时, 无效部分

的数据

数据:128bytes

填充 0xFF

4.3 PBU/OBC/ECU 命令字定义

表8 PBU/OBC/ECU 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令（未注明为 PBU/OBC/ECU 通用）					
0x730	0x0C	0x1008	关机指令 （主动发送，收到返回或超时停止）	ASCII 字符	SHUTDOWN，收到 MC、HMI 的 READY 或超时 1s 后，关闭开机信号
0x730	0x0C	0x1140	PBU/OBC 版本信息 （返回指令）	ASCII 字符	排列顺序为：MODE、SN、HW、FW； 每条信息长度为 16 bytes，结束符为 ' '，无效填充 0x20 FW 命名格式为 Vxrxrx_YYYYMMDD
0x730	0x0C	0x120C	PBU/OBC 物理 ID （返回指令）	ID:12bytes	不足位填充 1
0x730	0x0C	0x130C	PBU/OBC 校验码 （返回指令）	校验码:12bytes	
0x730	0x0C	0x1405	关机就绪 （返回指令）	ASCII 字符	READY
0x730	0x0C	0x1504	PBU/OBC 故障代码 （存在故障时 200ms 自动发送，故障消失后停止发送）	高 16 位:0x0000 低 16 位 0x0000:无故障 0x0001:+键失效 0x0002:-键失效 0x0004:i 键失效 0x0008:灯键失效 0x0010:Walk 键失效 0x0020:电源键失效 0x0040:MCU 故障 0x0080:MOS 短路 0x0100:电压检测异常 0x0200:HMI 通讯异常 0x0400:MC 通讯异常 0x0800:预留故障 1	按位或输出，0-正常，1-故障

				0x1000: 预留故障 2 0x2000: 预留 0x4000: 预留 0x8000: 预留	
发送给 MC (未注明为 PBU/OBC/ECU 通用)					
0x731	0x0C	0x3002	控制电机指令 (定时上传)	助力档位: 1byte 大灯状态: 1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x731	0x0C	0x3105	在线检测反馈 (返回指令)	ASCII 字符	READY
0x731	0x16	0x3208	PBU 设置用户控制参数 (主动发送, 收到返回或超时停止)	档位数量: 1byte 支持无 HMI: 1byte 预留: 6bytes	3~5 0x55-不支持, 0xAA-支持 填充 0x00
0x731	0x11	0x3300	OBC 查询用户参数 (主动发送, 收到返回或超时停止)		
0x731	0x16	0x3408	OBC 设置用户参数 (主动发送, 收到返回或超时停止)	周长微调值: 1byte 启动模式: 1byte 预留: 6bytes	有符型, 单位 1cm, ±10.0cm 1-柔和, 2-正常, 3-强劲 填充 0x00
0x731	0x11	0x3500	OBC 查询骑行历史信息 (主动发送, 收到返回或超时停止)		
0x731	0x16	0x3605	OBC 清除电机 TRIP 信息(主动发送, 收到返回或超时停止)	ASCII 字符	CLEAR
0x731	0x0C	0x3708	ECU 控制电机指令 (主动发送, 收到返回或超	助力档位: 1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM

			时停止)		0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x11: 踢频 ECO 0x12: 踢频 NORM 0x13: 踢频 SPORT 0x14: 踢频 TURBO 0x15: 踢频预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开 填充 0x00
0x731	0x16	0x3810	ECU 设置用户参数 (主动发送, 收到返回或超时停止)	大灯状态:1byte 预留:6bytes	
0x731	0x11	0x3900	OBC/ECU 读取电机版本信息		
发送给 BMS (未注明为 PBU/OBC/ECU 通用)					
0x732	0x11	0x5000	查询 BMS 运行信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5100	OBC/ECU 查询 BMS 版本信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5200	OBC/ECU 查询 BMS 设计信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5300	OBC/ECU 查询 BMS 电芯电压 (主动发送, 收到返回或超时停止)		
发送给 HMI (仅 PBU 有效)					
0x734	0x0C	0x7006	设置调节按键 (发送一次)	+键:1byte -键:1byte	0-松开 1-短按

				i 键:1byte 预留:3bytes	2-长按 填充 0x00
0x734	0x0C	0x7103	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x734	0x16	0x7203	HMI 进入设置模式 (主动发送, 收到返回或超时停止)	ASCII 字符	SET
0x734	0x16	0x7304	HMI 进入预推行模式 (主动发送, 收到返回或超时停止)	ASCII 字符	STAR:进入 STOP:退出
0x734	0x0C	0x7402	助力档位指令 (主动定时发送, 超时未收到 HMI 广播的运行信息, 认为 HMI 通讯异常, HMI 显示档位以该指令为准, 若超时未收到该指令, 显示档位切换为 OFF, 且提示 PBU 通讯故障代码)	助力档位:1byte 大灯状态:1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x11: 踢频 ECO 0x12: 踢频 NORM 0x13: 踢频 SPORT 0x14: 踢频 TURBO 0x15: 踢频预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x734	0x16	0x7501	开机指令 (主动发送, 收到返回或超时停止)	功能:1byte	0x01: 运行 0x02: 密码校验
0x734	0x0C	0x7601	密码状态指令 (HMI 查询密码状态时发送)	状态:1byte	0x01:ON 0x02:OFF
发送给 CDL (未注明为 PBU/OBC/ECU 通用)					
0x735	0x0C	0x9003	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x735	0x0C	0x9101	PBU/OBC 发送按键状态 (定时上传)	+键:Bit0 -键:Bit1 i 键:Bit2 灯键:Bit3 Walk 键:Bit4	0-松开 1-按下 多余位填充 0

				电源键:Bit5	
0x735	0x0C	0x9210	PBU 出厂配置参数 (返回指令)	额定电压:1byte 助力档位:1byte 支持无 HMI:1byte 自动关机时间:1byte 预留 12bytes	0-24V, 1-36V, 2-48V 3~5 档 0x55-支持, 0xAA 不支持 单位: min, 0xFF 为不自动关机 填充 0x00
0x735	0x0C	0x9310	运行历史信息 (返回指令)	运行时间:4bytes 开机次数:4bytes 预留:8bytes	1min 次 填充 0x00
0x735	0x0C	0x9420	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x735	0x0C	0x9510	自定义可存储字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x735	0x0C	0x9610	自定义可存储字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x735	0x0C	0x9710	自定义可存储字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x735	0x0C	0x9888	存储器指定起始和结束地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF
0x735	0x0C	0x9910	OBC 出厂默认配置	额定电压:1byte 预留:1byte 启动模式:1byte 显示续航:1byte 显示电量百分比:1byte 显示大灯图标:1byte 亮度等级:1byte 显示单位:1byte 启动模式设置:1byte 语言:1byte	0-24V, 1-36V, 2-48V 填充 0x00 1-柔和, 2-正常, 3-强劲 0-不显示, 1-显示 0-不显示, 1-显示 0-不显示, 1-显示 1~5 0-km/h, 1-mph 0-不支持, 1-支持 0-中文, 1-英文

预留:6bytes

填充 0x00

4.4 HMI 命令字定义

表9 HMI 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x740	0x0C	0x1040	HMI 版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、 SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20 HW 和 FW 命名格式为 Vxrxrx_YYYYMMDD
0x740	0x0C	0x110C	HMI 物理 ID (返回指令)	ID:12bytes	不足位填充 1
0x740	0x0C	0x120C	HMI 校验码 (返回指令)	校验码:12bytes	
0x740	0x0C	0x1305	关机就绪 (返回指令)	ASCII 字符	READY
0x740	0x0C	0x1404	HMI 故障码 (定时上报)	高 16 位:0x0000 低 16 位 0x0000:无故障 0x0001: LCD 故障 0x0002: MCU 故障 0x0004: 预留 0x0008: 预留 0x0010: 预留 0x0020: 预留 0x0040: 预留 0x0080: 预留 0x0100: 预留 0x0200: 预留 0x0400: 预留 0x0800: 预留 0x1000: 预留 0x2000: 预留 0x4000: 预留 0x8000: 预留	按位或输出, 0-正 常, 1-故障, 存在故 障时 200ms 自动发 送, 故障消失后停止 发送
0x740	0x0C	0x1508	上报 HMI 运行信 息	预留:8Bytes	填充 0x00
发送给 MC					
0x741	0x11	0x3000	查询电机版本 信息		

			(主动发送, 收到返回或超时停止)		
0x741	0x11	0x3100	查询电机配置参数 (主动发送, 收到返回或超时停止)		
0x741	0x16	0x3208	设置电机配置参数 (主动发送, 收到返回或超时停止)	周长微调值:1byte 启动模式:1byte 预留:6bytes	有符型, $\pm 10\text{cm}$ 1-柔和, 2-正常, 3-强劲 填充 0x00
0x741	0x0C	0x3305	在线检测反馈 (返回指令)	ASCII 字符	READY
0x741	0x11	0x3400	查询骑行历史信息 (主动发送, 收到返回或超时停止)		
0x741	0x16	0x3505	清除电机 TRIP 信息 (主动发送, 收到返回或超时停止)	ASCII 字符	CLEAR
发送给 BMS					
0x742	0x11	0x5000	查询BMS版本信息 (主动发送, 收到返回或超时停止)		
0x742	0x11	0x5100	查询BMS设计信息 (主动发送, 收到返回或超时停止)		
0x742	0x11	0x5200	查询BMS电芯电压 (主动发送, 收到返回或超时停止)		
发送给 PBU					
0x743	0x11	0x7000	查询PBU版本信息 (主动发送, 收到返回或超时停止)		

			停止)		
0x743	0x16	0x7100	退出设置模式 (主动发送, 收到返回或超时停止)		
0x743	0x0C	0x7203	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x743	0x16	0x7308	设置参数 (主动发送, 收到返回或超时停止)	自动关机时间:1byte 预留:7bytes	单位: min, 0 为不自动关机 填充 0x00
0x743	0x16	0x7407	密码设置指令 (主动发送, 收到返回或超时停止)	ASCII 字符	PSW_SET
0x743	0x11	0x7500	查询密码状态指令 (主动发送, 收到返回或超时停止)		
0x743	0x16	0x7608	密码输入指令 (主动发送, 收到返回或超时停止)	密码:4bytes 密码模式:1byte 预留:3bytes	十进制 0x01:开机验证 0x02:密码关闭 0x03:密码修改 0x04:密码开启 填充 0x00
发送给 CDL					
0x745	0x0C	0xA010	用户参数默认配置 (返回指令)	额定电压:1byte 预留:1byte 启动模式:1byte 显示续航:1byte 显示电量百分比:1byte 显示大灯图标:1byte 亮度等级:1byte 显示单位:1byte 启动模式设置:1byte 语言:1byte 预留:6bytes	0-24V, 1-36V, 2-48V 填充 0x00 1-柔和, 2-正常, 3-强劲 0-不显示, 1-显示 0-不显示, 1-显示 0-不显示, 1-显示 1~5 0-km/h, 1-mph 0-不支持, 1-支持 0-中文, 1-英文 填充 0x00
0x745	0x0C	0xA110	运行历史信息	运行时间:4bytes	1min

			(返回指令)	开机次数:4bytes 预留:8bytes	次 填充 0x00
0x745	0x0C	0xA220	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x745	0x0C	0xA310	自定义可存储 字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x745	0x0C	0xA410	自定义可存储 字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x745	0x0C	0xA510	自定义可存储 字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x745	0x0C	0xA603	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x745	0x0C	0xA788	返回存储器指 定起始和结束 地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF

4.5 CDL 命令字定义

表10 CDL 命令字定义

ID	模式	命令字	功能	数据段	备注
发送给 MC					
0x751	0x11	0x1000	查询电机校验 码密钥		
0x751	0x16	0x1108	写入电机校验 码密钥		
0x751	0x11	0x1200	查询电机版本 信息		
0x751	0x11	0x1300	查询自定义可 存储字符串 1		
0x751	0x16	0x1410	写入自定义可 存储字符串 1	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1500	查询自定义可 存储字符串 2		
0x751	0x16	0x1610	写入自定义可 存储字符串 2	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1700	查询自定义可 存储字符串 3		

0x751	0x16	0x1810	写入自定义可 存储字符串 3	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x16	0x1901	写入电机工作 模式	工作模式:1byte	0-运行, 1-配置
0x751	0x11	0x1A00	查询电机用户 参数 1		
0x751	0x16	0x1B20	写入电机用户 参数 1	指拨模式:1byte 启动模式:1byte 停机时间:2bytes 限速值:1byte 下降速度:1byte 前飞:1byte 后飞:1byte 限流:1byte 温度预警:1byte 温度保护:1byte 无码表支持:1byte 轮胎周长:1byte 电机系列号:1byte ECO 助力比增 益:1byte ECO 加速度增 益:1byte NOMA 助力比增 益:1byte NOMA 加速度增 益:1byte SPORT 助力比增 益:1byte SPORT 加速度增 益:1byte TURBO 助力比增 益:1byte TURBO 加速度增 益:1byte SMART 助力比增 益:1byte SMART 加速度增 益:1byte 速度检测信号个 数:1byte	0x55-不支持 0xAA- 支持 1-柔和, 2-正常, 3-强劲 1ms 1km/h 1~5 个等级 T T 1A +40℃ +40℃ 0x55-不支持 0xAA- 支持 1cm 1~15 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 0.01 倍, 50~150 1~100

				踏频启动信号个数:1byte 速度传感器信号来源:1byte 周长微调值:1byte 低压保护阈值:2bytes 推行限速:1byte 推行马达转速:1byte	1~100 0x55-车轮触发 0xAA-踏频估算 0xEE-通讯获取 1cm, 有符型 mV 0.1km/h 1rpm
0x751	0x11	0x1C00	查询马达配置参数		
0x751	0x16	0x1D10	写入马达配置参数	额定功率:2bytes 额定转速:2bytes 定子电阻:2bytes 定子 Lq:2bytes 定子 Ld:2bytes 反电动势:2bytes 额定电压:1bytes 预留:3bytes	1W 1rpm 1mOhm 1mH 1mH 1mV 1V 填充 0x00
0x751	0x11	0x1E00	查询电机历史信息		
0x751	0x11	0x1F00	查询电机生产信息		
0x751	0x11	0x2000	查询力矩传感器零点信息 [单力矩]		
0x751	0x11	0x2100	查询设备在线检测结果		
0x751	0x16	0x2210	写入电机 Mode (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x751	0x16	0x2310	写入电机 SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x751	0x16	0x2420	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 产品标识:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD MM_VP1.
0x751	0x16	0x2505	复位指令	ASCII 字符串	RESET

0x751	0x16	0x2605	数据清除	ASCII 字符串	CLEAR
0x751	0x16	0x2708	参数还原	ASCII 字符串	RECOVERY
0x751	0x16	0x2802	控制指令	助力档位:1byte 大灯状态:1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x751	0x11	0x2900	查询力矩传感器校正信息 [单力矩]		
0x751	0x16	0x2A01	写入力矩传感器标定系数 [单力矩]	标定系数:1byte	0~100
0x751	0x16	0x2B02	写入力矩传感器启动值	启动值:2bytes	
0x751	0x16	0x2C01	设置电机转速百分比	百分比:1byte	5~100
0x751	0x11	0x2D08	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤ 128Bytes
0x751	0x11	0x2E00	查询骑行历史信息		
0x751	0x11	0x2F00	查询角度传感器数值		
0x751	0x11	0x3000	查询电机用户参数 2		
0x751	0x16	0x3120	写入电机用户参数 2	俯仰角零度偏移:2bytes 横滚角零度偏移:2bytes 支持角度传感器标志:1byte 尾灯工作模式:1byte	0.1°, ±900 0.1°, ±900 0x55-不支持 0xAA-支持 0x01: 模式 1, 连接尾灯, 开灯时低亮, 刹车时高亮 0x02: 模式 2, 连接尾灯, 开灯时高亮, 刹车时闪烁 0x03: 模式 3, 连接刹车灯, 刹车时高亮 0x04: 模式 4, 连接

				前灯电压:1byte 尾灯电压:1byte 预留:24bytes	刹车灯, 刹车时闪烁 0x06: 6V 0x0C: 12V 0xFF: 随电池电压 0x06: 6V 0x0C: 12V 0xFF: 随电池电压 填充 0x00
0x751	0x11	0x3201	查询 3 个力矩 传感器零点信息 [三力矩]	传感器序号:1byte	
0x751	0x11	0x3301	查询 3 个力矩 传感器标定系数 [三力矩]	传感器序号:1byte	
0x751	0x16	0x3402	写入 3 个力矩 传感器标定系数 [三力矩]	传感器序号:1byte 标定系数:1byte	0~100
0x751	0x16	0x3500	查询启动值	启动值:2bytes	
0x751	0x11	0x3601	查询力矩传感器 线性校正参数 [三力矩 J01]	力矩值序号:1byte	1~15
0x751	0x16	0x3702	设定加载力矩 值, 用于线性 校正 [三力矩 J01]	力矩值序号:1byte 力矩输入值:1byte	1~15 0~75N.m
0x751	0x16	0x3805	强制写入校正 参数 [三力矩 J01]	序号:1byte AD 采集值:2byte 力矩值:1byte 校正参数:1byte	1~15 N.m 放大 128 倍
0x751	0x16	0x3909	历史信息清除	ASCII	LOG CLEAR
发送给 BMS					
0x752	0x11	0x3000	查询电池物理 ID		
0x752	0x11	0x3100	查询电池校验 码		
0x752	0x16	0x320C	写入电池校验 码	校验码:12bytes	
0x752	0x11	0x3300	查询电池版本 信息		

0x752	0x11	0x3400	查询电池运行信息		
0x752	0x11	0x3500	查询电芯电压		
0x752	0x11	0x3600	查询电池设计信息		
0x752	0x11	0x3700	查询电池生产信息		
0x752	0x11	0x3800	查询电池历史信息		
0x752	0x11	0x3900	查询自定义可存储字符串 1		
0x752	0x16	0x3A10	写入自定义可存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x11	0x3B00	查询自定义可存储字符串 2		
0x752	0x16	0x3C10	写入自定义可存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x11	0x3D00	查询自定义可存储字符串 3		
0x752	0x16	0x3E10	写入自定义可存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x16	0x3F20	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x752	0x16	0x4010	写入 BMS Mode (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x752	0x16	0x4110	写入 BMS SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x752	0x16	0x4205	复位指令	ASCII 字符串	RESET
0x752	0x11	0x4308	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤128Bytes
发送给 PBU/OBC (未注明为 PBU/OBC 通用)					
0x753	0x11	0x5000	查询 PBU/OBC 物理 ID		
0x753	0x11	0x5100	查询 PBU/OBC 校验码		
0x753	0x16	0x520C	写入 PBU/OBC 校验码	校验码:12bytes	

0x753	0x16	0x5300	查询 PBU/OBC 生产信息		
0x753	0x11	0x5400	查询 PBU/OBC 版本信息		
0x753	0x16	0x5504	设置进入/退出 按键测试模式	ASCII 字符	STAR: 进入 STOP: 退出
0x753	0x11	0x5600	查询 PBU/OBC 历史信息		
0x753	0x11	0x5700	查询 PBU 出厂 配置参数		
0x753	0x16	0x5810	写入 PBU 出厂 配置参数	额定电压: 1byte 助力档位: 1byte 支持无 HMI: 1byte 自动关机时间: 1byte 预留 12bytes	0-24V, 1-36V, 2-48V 3~5 档 0x55-支持, 0xAA 不 支持 单位: min, 0xFF 为 不自动关机 填充 0x00
0x753	0x11	0x5900	查询自定义可 存储字符串 1		
0x753	0x16	0x5A10	写入自定义可 存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x11	0x5B00	查询自定义可 存储字符串 2		
0x753	0x16	0x5C10	写入自定义可 存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x11	0x5D00	查询自定义可 存储字符串 3		
0x753	0x16	0x5E10	写入自定义可 存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x16	0x5F20	写入 PBU/OBC 生产信息 (可选, 仅供 生产商写入)	生产商: 8bytes 生产地: 8bytes 生产日期: 8bytes 预留: 8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x753	0x16	0x6010	写入 PBU/OBC Mode (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x753	0x16	0x6110	写入 PBU/OBC SN (可选, 仅供	ASCII 字符	结束符为 0x2E, 无 效填充 0x20

			生产商写入)		
0x753	0x16	0x6205	复位指令	ASCII 字符串	RESET
0x753	0x11	0x6308	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤128Bytes
0x753	0x11	0x6400	查询 OBC 出厂默认配置		
0x753	0x16	0x6510	写入 OBC 出厂默认配置	额定电压:1byte 预留:1byte 启动模式:1byte 显示续航:1byte 显示电量百分比:1byte 显示大灯图标:1byte 亮度等级:1byte 显示单位:1byte 启动模式设置:1byte 语言:1byte 预留:6bytes	0-24V, 1-36V, 2-48V 填充 0x00 1-柔和, 2-正常, 3-强劲 0-不显示, 1-显示 0-不显示, 1-显示 0-不显示, 1-显示 1~5 0-km/h, 1-mpH 0-不支持, 1-支持 0-中文, 1-英文 填充 0x00
0x753	0x16	0x6609	清除 PBU/OBC 开机密码	ASCII 字符	PSW_CLEAR
发送给 HMI					
0x754	0x11	0x7000	查询 HMI 物理 ID		
0x754	0x11	0x7100	查询 HMI 校验码		
0x754	0x16	0x720C	写入 HMI 校验码	校验码:12bytes	
0x754	0x11	0x7300	查询 HMI 版本信息		
0x754	0x11	0x7400	查询 HMI 生产信息		
0x754	0x11	0x7500	查询用户参数默认配置		
0x754	0x16	0x7610	写入用户参数默认配置	额定电压:1byte 预留:1byte 启动模式:1byte 显示续航:1byte 显示电量百分比:1byte	0-24V, 1-36V, 2-48V 填充 0x00 1-柔和, 2-正常, 3-强劲 0-不显示, 1-显示 0-不显示, 1-显示

				显示大灯图标: 1byte 亮度等级:1byte 显示单位:1byte 启动模式设置:1byte 语言:1byte 预留:6bytes	0-不显示, 1-显示 1~5 0-km/h, 1-mpH 0-不支持, 1-支持 0-中文, 1-英文 填充 0x00
0x754	0x11	0x7700	查询运行历史信息		
0x754	0x11	0x7800	查询自定义可存储字符串 1		
0x754	0x16	0x7910	写入自定义可存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x754	0x11	0x7A00	查询自定义可存储字符串 2		
0x754	0x16	0x7B10	写入自定义可存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x754	0x11	0x7C00	查询自定义可存储字符串 3		
0x754	0x16	0x7D10	写入自定义可存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x754	0x16	0x7E20	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x754	0x16	0x7F10	写入 HMI Mode (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x754	0x16	0x8010	写入 HMI SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x754	0x16	0x8105	复位指令	ASCII 字符串	RESET
0x754	0x16	0x8204	设置进入/退出按键测试模式 (进入测试模式下, 关闭超时检查是否收到 MC 和 PBU 指令的功能)	ASCII 字符	STAR: 进入 STOP: 退出

0x754	0x11	0x8308	读取存储器指 定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤ 128Bytes
-------	------	--------	-----------------	----------------------------	---------------------

注: CDL 发送的所有指令均按照定时 200ms 发送, 其中查询指令收到返回的信息或超时 1s 停止发送, 写入指令收到通用反馈指令或超时 1s 停止发送。

5 附录 1: CRC32 计算方法

5.1 CRC32 计算多项式表

```
1. uint32_t Crc32Table[ 256 ] =
2. {
3.     0x00000000, 0x04C11DB7, 0x09823B6E, 0x0D4326D9, 0x130476DC, 0x17C56B6B,
4.     0x1A864DB2, 0x1E475005, 0x2608EDB8, 0x22C9F00F, 0x2F8AD6D6, 0x2B4BCB61,
5.     0x350C9B64, 0x31CD86D3, 0x3C8EA00A, 0x384FBDBD, 0x4C11DB70, 0x48D0C6C7,
6.     0x4593E01E, 0x4152FDA9, 0x5F15ADAC, 0x5BD4B01B, 0x569796C2, 0x52568B75,
7.     0x6A1936C8, 0x6ED82B7F, 0x639B0DA6, 0x675A1011, 0x791D4014, 0x7DDC5DA3,
8.     0x709F7B7A, 0x745E66CD, 0x9823B6E0, 0x9CE2AB57, 0x91A18D8E, 0x95609039,
9.     0x8B27C03C, 0x8FE6DD8B, 0x82A5FB52, 0x8664E6E5, 0xBE2B5B58, 0xBAEA46EF,
10.    0xB7A96036, 0xB3687D81, 0xAD2F2D84, 0xA9EE3033, 0xA4AD16EA, 0xA06C0B5D,
11.    0xD4326D90, 0xD0F37027, 0xDB056FE, 0xD9714B49, 0xC7361B4C, 0xC3F706FB,
12.    0xCEB42022, 0xCA753D95, 0xF23A8028, 0xF6FB9D9F, 0xFBB8BB46, 0xFF79A6F1,
13.    0xE13EF6F4, 0xE5FFE643, 0xE8BCCD9A, 0xEC7DD02D, 0x34867077, 0x30476DC0,
14.    0x3D044B19, 0x39C556AE, 0x278206AB, 0x23431B1C, 0x2E003DC5, 0x2AC12072,
15.    0x128E9DCF, 0x164F8078, 0x1B0CA6A1, 0x1FCD8B16, 0x018AEB13, 0x054BF6A4,
16.    0x0808D07D, 0x0CC9CDCA, 0x7897AB07, 0x7C56B6B0, 0x71159069, 0x75D48DDE,
17.    0x6B93DDDB, 0x6F52C06C, 0x6211E6B5, 0x66D0FB02, 0x5E9F46BF, 0x5A5E5B08,
18.    0x571D7DD1, 0x53DC6066, 0x4D9B3063, 0x495A2DD4, 0x44190B0D, 0x40D816BA,
19.    0xACA5C697, 0xA864DB20, 0xA527FDF9, 0xA1E6E04E, 0xBFA1B04B, 0xBB60ADFC,
20.    0xB6238B25, 0xB2E29692, 0x8AAD2B2F, 0x8E6C3698, 0x832F1041, 0x87EE0DF6,
21.    0x99A95DF3, 0x9D684044, 0x902B669D, 0x94EA7B2A, 0xE0B41DE7, 0xE4750050,
22.    0xE9362689, 0xEDF73B3E, 0xF3B06B3B, 0xF771768C, 0xFA325055, 0xFE34DE2,
23.    0xC6BCF05F, 0xC27DEDE8, 0xCF3ECB31, 0xCBFFD686, 0xD5B88683, 0xD1799B34,
24.    0xDC3ABDED, 0xD8FBA05A, 0x690CE0EE, 0x6DCDFD59, 0x608EDB80, 0x644FC637,
25.    0x7A089632, 0x7EC98B85, 0x738AAD5C, 0x774BB0EB, 0x4F040D56, 0x4BC510E1,
26.    0x46863638, 0x42472B8F, 0x5C007B8A, 0x58C1663D, 0x558240E4, 0x51435D53,
27.    0x251D3B9E, 0x21DC2629, 0x2C9F00F0, 0x285E1D47, 0x36194D42, 0x32D850F5,
28.    0x3F9B762C, 0x3B5A6B9B, 0x0315D626, 0x07D4CB91, 0x0A97ED48, 0x0E56F0FF,
29.    0x1011A0FA, 0x14D0BD4D, 0x19939B94, 0x1D528623, 0xF12F560E, 0xF5EE4BB9,
30.    0xF8AD6D0, 0xFC6C70D7, 0xE22B20D2, 0xE6EA3D65, 0xEBA91BBC, 0xEF68060B,
31.    0xD727BBB6, 0xD3E6A601, 0xDEA580D8, 0xDA649D6F, 0xC423CD6A, 0xC0E2D0DD,
32.    0xCDA1F604, 0xC960EBB3, 0xBD3E8D7E, 0xB9FF90C9, 0xB4BCB610, 0xB07DABA7,
33.    0xAE3AFBA2, 0xAAFBE615, 0xA7B8C0CC, 0xA379DD7B, 0x9B3660C6, 0x9FF77D71,
34.    0x92B45BA8, 0x9675461F, 0x8832161A, 0x8CF30BAD, 0x81B02D74, 0x857130C3,
35.    0x5D8A9099, 0x594B8D2E, 0x5408ABF7, 0x50C9B640, 0x4E8EE645, 0x4A4FFBF2,
36.    0x470CDD2B, 0x43CDC09C, 0x7B827D21, 0x7F436096, 0x7200464F, 0x76C15BF8,
37.    0x68860BFD, 0x6C47164A, 0x61043093, 0x65C52D24, 0x119B4BE9, 0x155A565E,
38.    0x18197087, 0x1CD86D30, 0x029F3D35, 0x065E2082, 0x0B1D065B, 0x0FDC1BEC,
39.    0x3793A651, 0x3352BBE6, 0x3E119D3F, 0x3AD08088, 0x2497D08D, 0x2056CD3A,
40.    0x2D15EBE3, 0x29D4F654, 0xC5A92679, 0xC1683BCE, 0xCC2B1D17, 0xC8EA00A0,
```


MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 41 页
41. 0xD6AD50A5, 0xD26C4D12, 0xDF2F6BCB, 0xDBEE767C, 0xE3A1CBC1, 0xE760D676, 42. 0xEA23F0AF, 0xEEE2ED18, 0xF0A5BD1D, 0xF464A0AA, 0xF9278673, 0xFDE69BC4, 43. 0x89B8FD09, 0x8D79E0BE, 0x803AC667, 0x84FBDDBD, 0x9ABC8BD5, 0x9E7D9662, 44. 0x933EB0BB, 0x97FFAD0C, 0xAFB010B1, 0xAB710D06, 0xA6322BDF, 0xA2F33668, 45. 0xBCB4666D, 0xB8757BDA, 0xB5365D03, 0xB1F740B4 46. };			
5.2 CRC32 计算方法 <pre>1. uint32_t CRC32_Calculate(uint8_t *pData, uint16_t Length) 2. { 3. uint32_t nReg; 4. uint32_t nTemp = 0; 5. uint16_t i, n; 6. 7. nReg = 0xFFFFFFFF; 8. for (n = 0; n < Length; n++) 9. { 10. nReg ^= (uint32_t) pData[n]; 11. for (i = 0; i < 4; i++) 12. { 13. nTemp = Crc32Table[(uint8_t)((nReg >> 24) & 0xFF)]; 14. nReg <<= 8; 15. nReg ^= nTemp; 16. } 17. } 18. return nReg; 19. }</pre>			
保密文件， 禁止外传			

6 附录 2: 警告/故障码列表

当总线上任何一个设备上报警告/故障代码时,显示设备按照下列规则进行转换后,在屏幕上依次循环显示代码,间隔周期为 1s,显示顺序以接收到代码的先后顺序为准,当超时 3s 未接受到任何警告/故障代码时恢复原显示界面。

表11 警告/故障码列表

信号来源	显示代码	类型	描述	显示内容	解决办法
MC	10	警告	过流保护	Over Current Protect	5s 后自动恢复
MC	11	警告	低压保护	Under Voltage Protect	电池充电
MC	12	警告	过压保护	Over Voltage Protect	更换正确电池
MC	13	警告	堵转保护	Rotor Locked	关机重启
MC	14	警告	过热保护	Over Heat Protect	关机静置 30min 后使用
MC	15	故障	温度传感器故障	NTC Fault	返修
MC	16	故障	速度传感器故障	Speed Sensor Fault	检查或更换速度传感器
MC	17	故障	力矩传感器故障	Torque Sensor Fault	返修
MC	18	故障	马达霍尔传感器故障	Motor Fault	返修
MC	19	警告	BMS 校验失败	BMS Check Fault	更换电池
MC	20	警告	PBU 校验失败	PBU Check Fault	更换按键
MC	21	警告	HMI 校验失败	HMI Check Fault	更换仪表
MC	22	故障	马达缺相	PhaseLine Fault	返修
MC	23	故障	踏频传感器故障	Cadence Sensor Fault	返修
MC	24	故障	指拨故障	Gas Sensor Fault	更换指拨
MC	25	故障	MOS 短路	MOS Short Circuit	返修
MC	26	故障	电压异常波动	Bus Voltage Abnormal	更换电池
MC	27	故障	处理器故障	MCU Fault	返修
MC	28	故障	电路故障	Circuit Fault	返修
MC	29	故障	TE 故障	TE MCU Fault	返修
MC	30	故障	TE 电路故障	TE Circuit Fault	返修
MC	31	警告	MC 校验失败	MC Check Fault	返修
MC	32	故障	马达失速	Motor Stall	返修
MC	33	故障	预留故障位	-	-
BMS	40	警告	预留故障位	-	-
BMS	41	警告	充电过压警告	Charge Over Voltage Alarm	停止充电或更换充电器
BMS	42	警告	放电低压警告	DisCharge Under Voltage Alarm	及时充电
BMS	43	警告	充电过流警告	Charge Over Current	更换充电器

				Alarm	
BMS	44	警告	放电过流警告	Over Current Alarm	停止骑行后故障消失
BMS	45	警告	充电高温警告	Charge Over Heat Alarm	停止充电
BMS	46	警告	充电低温警告	Charge Low Temperature Alarm	停止充电
BMS	47	警告	放电高温警告	DisCharge Over Heat Alarm	关机静置 30min 后使用
BMS	48	警告	放电低温警告	DisCharge Low Temperature Alarm	建议停止使用
BMS	49	警告	MOS 高温警告	MOS Over Heat Alarm	5s 后自动恢复
PBU/OBC	60	故障	+ 按键失效	+ Key Fault	检查或更换按键
PBU/OBC	61	故障	- 键失效	- Key Fault	检查或更换按键
PBU/OBC	62	故障	i 键失效	- Key Fault	检查或更换按键
PBU/OBC	63	故障	Light 键失效	Light Key Fault	检查或更换按键
PBU/OBC	64	故障	Walk 键失效	Walk Key Fault	检查或更换按键
PBU/OBC	65	故障	电源键失效	Power Key Fault	检查或更换按键
PBU	66	故障	处理器故障	MCU Fault	返修
PBU	67	故障	开关 MOS 短路	MOS Short Circuit	返修
PBU	68	故障	电压检测异常	Voltage Test Fault	返修
PBU	69	故障	HMI 通讯异常	HMI Communication Fault	检查线路或更换部件
PBU/OBC	70	故障	MC 通讯异常	MC Communication Fault	检查线路或更换部件
PBU	71	故障	预留故障位 1		
PBU	72	故障	预留故障位 2		
HMI	80	故障	MC 通讯异常	MC Communication Fault	检查线路或更换部件
HMI	81	故障	PBU 通讯异常	PBU Communication Fault	检查线路或更换部件

7 附录 3：升级协议及流程

7.1 升级流程图

系统 CAN 总线上除 CDL 外所有设备均需通过 CDL 完成升级，升级流程图如下图所示：

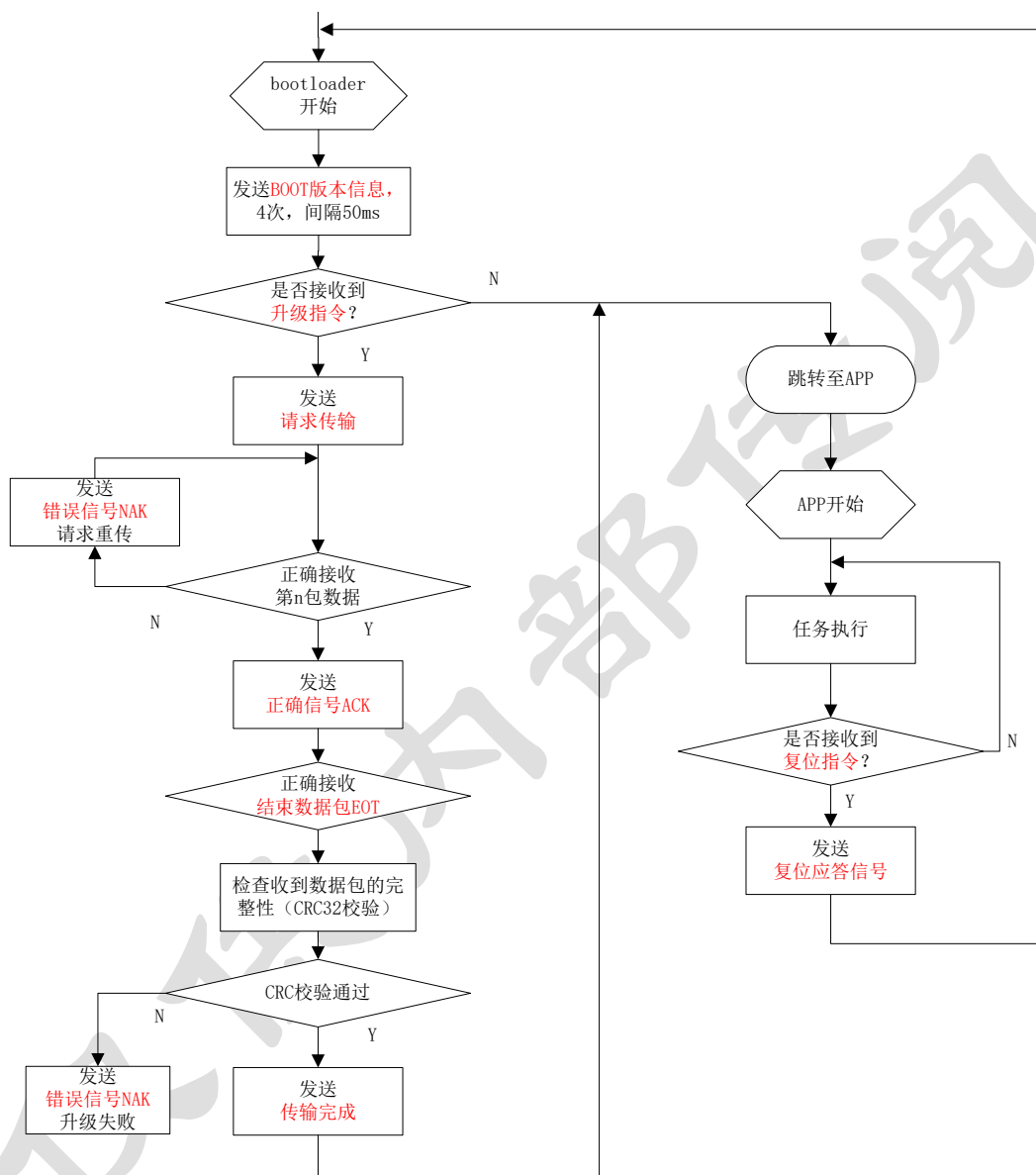


图3 升级流程图

7.2 升级协议

APP 程序中复位指令和复位应答指令参考第 3 章有关 CDL 发送给各设备的复位指令，以及各设备发送给 CDL 的通用应答指令。

Bootloader 中的协议描述如下:

表12 CDL 发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
升级指令	MC	0x751	0x16	0x0D	0xA10B	ASCII 字符: "MC_UPD" + 数据包大小 4 字节 (如 78351B, 内容为: 0F 32 01 00)
	BMS	0x752				ASCII 字符: "BMS_UPD" + 数据包大小 4 字节
	PBU/OBC	0x753				ASCII 字符: "PBU_UPD 或" OBC_UPD " + 数据包大小 4 字节
	HMI	0x754				ASCII 字符: "HMI_UPD" + 数据包大小 4 字节
数据包格式	MC	0x751	0x16	0x87	0xA385	SOH(01)+序号(1~65535)+总包数(1~65535)+数据(长度 128B, 无效填充 0xFF)
	BMS	0x752				
	PBU/OBC	0x753				
	HMI	0x754				
结束数据包	MC	0x751	0x16	0x03	0xA401	EOT(04)
	BMS	0x752				
	PBU/OBC	0x753				
	HMI	0x754				

表13 设备发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
BOOT 版本信息	MC	0x715	0x0C	0x0B	0xC109	ASCII 字符: "MC "+"Vx. x. x"
	BMS	0x725				ASCII 字符: "BMS"+"Vx. x. x"
	PBU/OBC	0x735				ASCII 字符: "PBU"/"OBC"+"Vx. x. x"
	HMI	0x745				ASCII 字符: "HMI"+"Vx. x. x"
正确信号	MC	0x715	0x0C	0x04	0xC202	当前包号, 2byte
	BMS	0x725				
	PBU/OBC	0x735				
	HMI	0x745				
错误信号	MC	0x715	0x0C	0x04	0xC302	当前包号, 2byte
	BMS	0x725				
	PBU/OBC	0x735				
	HMI	0x745				
请求传输	MC	0x715	0x0C	0x04	0xC402	0x00 0x00
	BMS	0x725				
	PBU/OBC	0x735				

	HMI	0x745				
传输完成	MC	0x715	0x0C	0x04	0xC502	0x00 0x00
	BMS	0x725				
	PBU/OBC	0x735				
	HMI	0x745				

8 附录 4: PBU 开机密码工作流程图

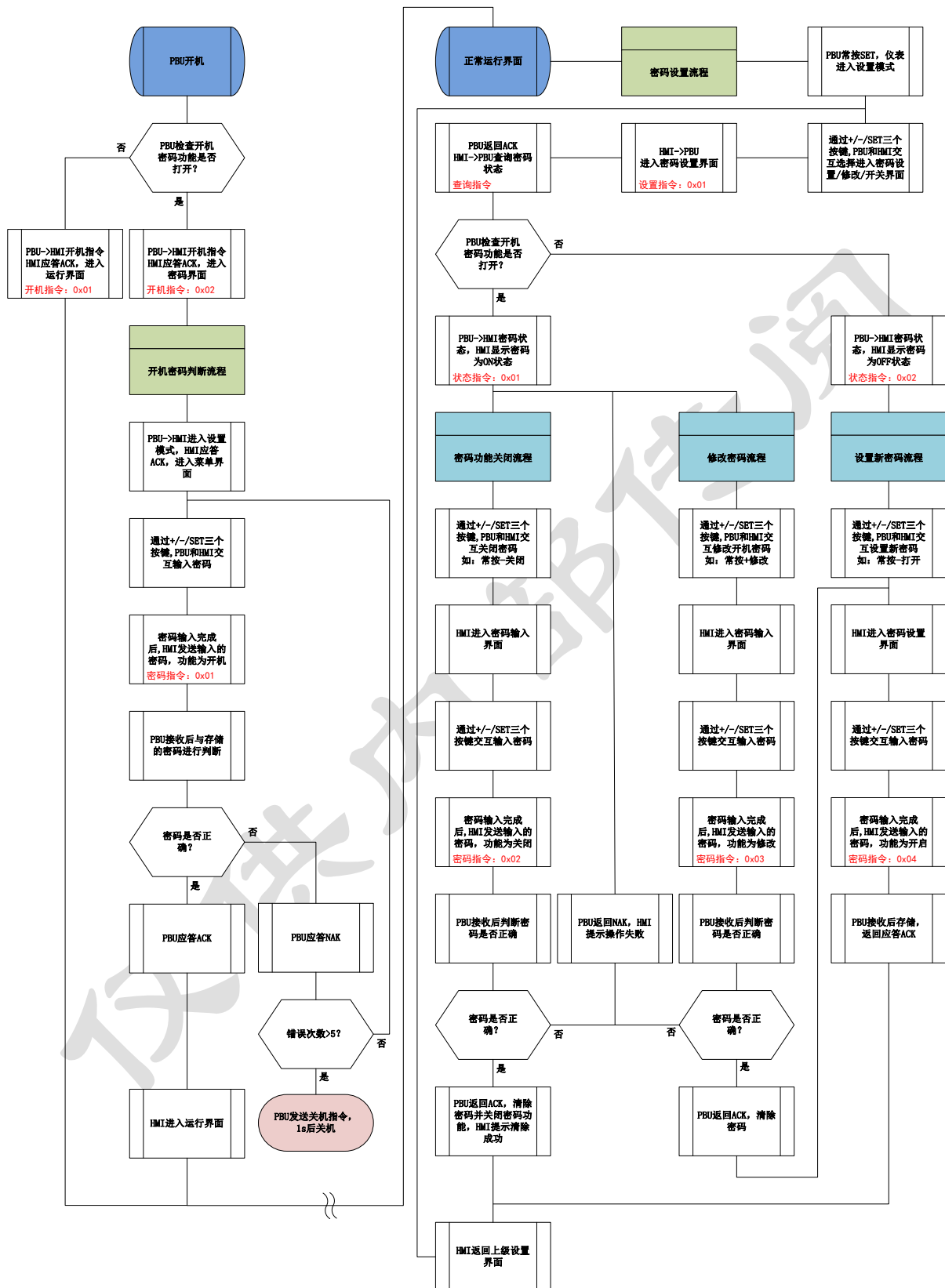


图4 PBU 开机密码工作流程


```
10.         DataLength = AddressEnd - AddressBegin+ 1;
11.         memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.         SendData(ID_MC_TO_CDL, MODE_REPORT, (0xAB08 + DataLength), (uint8_t*)Data);
13.     }
14. }while(0);
15. break;
16. }
```

9.4 数据结构定义

```
1. //故障日志记录信息 64 bytes, 1K 空间可以保存最近 16 条故障记录
2. typedef struct
3. {
4.     uint16_t Error_Index;           //故障列表索引 2 bytes, 地址偏移 0
5.     uint16_t NotesInfo1;           //备注信息 1, 地址偏移 2
6.     uint16_t NotesInfo2;           //备注信息 2, 地址偏移 4
7.     uint16_t NotesInfo3;           //备注信息 3, 地址偏移 6
8.     uint32_t ErrorCode;             //故障码 4 bytes, 地址偏移 8
9.     uint32_t RunTime;               //运行时间 4bytes, 地址偏移 12
10.    MC_RunInfo_Struct_t RunInfo;     //运行信息 32bytes, 地址偏移 16
11.    MC_AttitudeAngle_Struct_t AttitudeAngle; //车辆姿态角度 8bytes, 地址偏移 48
12.    MC_CalParam_Struct_t MC_CalParam; //马达控制参数 8bytes , 地址偏移 56
13. }MC_ErrorLogSaveInfo_Struct_t;
14.
15. //电机运行信息, 占用空间 32bytes
16. typedef struct
17. {
18.     uint16_t BikeSpeed;             //车速 0.1km/h, 地址偏移 0
19.     int16_t MotorSpeed;             //输出转速 1rpm, 地址偏移 2
20.     uint16_t Power;                 //电功率 1W, 地址偏移 4
21.     uint16_t BusVoltage;            //母线电压 1mV, 地址偏移 6
22.     uint16_t BusCurrent;            //母线电流 1mA, 地址偏移 8
23.     uint8_t Cadence;                //踏频 1rpm, 地址偏移 10
24.     uint8_t Torque;                 //踩踏力矩 1Nm, 地址偏移 11
25.     MC_CadenceDir_Struct_t CadenceDir; //踩踏方向 0-正,1-反,2-停止, 地址偏移 12
26.     uint8_t GearSt;                 //助力档位, 地址偏移 13
27.     MC_LightSwitch_Struct_t LightSwitch; //灯开关 0xF0-关, 0xF1-开, 地址偏移 14
28.     uint8_t SOC;                    //剩余电量 1%, 地址偏移 15
29.     uint16_t RemainDistance;         //续航里程 1km, 地址偏移 16
30.     uint16_t ODO_Km;                 //总里程 1km, 地址偏移 18
31.     uint8_t PowerPerKm;              //平均功耗 0.01Ah/km , 地址偏移 20
32.     uint8_t T_PCB;                  //PCB 温度 +40°C, 地址偏移 21
```

MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 50 页
33. uint8_t T_Coil; //绕组温度 +40℃, 地址偏移 22 34. uint8_t T_MCU; //MCU 温度 +40℃, 地址偏移 23 35. uint16_t Ride_Km; //开机后骑行里程 0.1km, 地址偏移 24 36. uint16_t Ride_Time; //开机后骑行时间 1s, 地址偏移 26 37. uint8_t TorqueSensorData1; //力矩传感器 1 原始值 38. uint8_t TorqueSensorData2; //力矩传感器 2 原始值 39. uint8_t TorqueSensorData3; //力矩传感器 3 原始值 40. uint8_t RS ; //预留 1byte 41. }MC_RunInfo_Struct_t; 42. 43. //姿态传感器数值 44. typedef struct 45. { 46. int16_t Angle_Pitch_Absolute; //俯仰角绝对角度 47. int16_t Angle_Roll_Absolute; //横滚角绝对角度 48. int16_t Angle_Pitch_Relative; //俯仰角相对角度 49. int16_t Angle_Roll_Relative; //横滚角相对角度 50. }MC_AttitudeAngle_Struct_t; 51. 52. //电机控制参数 53. typedef struct 54. { 55. MC_AssistRunMode_Struct_t AssistRunMode; //助力模式 56. int16_t Ref_Torque; //力矩模式输入 FOC 值 57. int16_t Ref_Speed; //速度模式输入 FOC 值 58. FlagStatus Foc_Flag; //FCO 运算标志 59. }MC_CalParam_Struct_t;			
保密文件， 禁止外传			


```
10.         DataLength = AddressEnd - AddressBegin+ 1;
11.         memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.         SendData(ID_MC_TO_CDL, MODE_REPORT, (0x5608 + DataLength), (uint8_t*)Data);
13.     }
14. }while(0);
15. break;
16. }
```

10.4 数据结构定义

```
1. //故障日志记录信息 128 bytes, 1K 空间可以保存最近 8 条故障记录
2. typedef struct
3. {
4.     uint16_t Error_Index;           //故障列表索引, 2 bytes, 地址偏移 0
5.     uint16_t NotesInfo1;           //备注信息 1, 2 Bytes, 地址偏移 2
6.     uint16_t NotesInfo2;           //备注信息 2, 2 Bytes, 地址偏移 4
7.     uint16_t NotesInfo3;           //备注信息 3, 2Bytes, 地址偏移 6
8.     uint32_t ErrorCode;             //故障码, 4 bytes, 地址偏移 8
9.     uint16_t CellVoltage_1;         //电芯 1 电压, 单位 mV, 2 Bytes, 地址偏移 12
10.    uint16_t CellVoltage_2;         //电芯 2 电压, 单位 mV, 2 Bytes, 地址偏移 14
11.    uint16_t CellVoltage_3;         //电芯 3 电压, 单位 mV, 2 Bytes, 地址偏移 16
12.    uint16_t CellVoltage_4;         //电芯 4 电压, 单位 mV, 2 Bytes, 地址偏移 18
13.    uint16_t CellVoltage_5;         //电芯 5 电压, 单位 mV, 2 Bytes, 地址偏移 20
14.    uint16_t CellVoltage_6;         //电芯 6 电压, 单位 mV, 2 Bytes, 地址偏移 22
15.    uint16_t CellVoltage_7;         //电芯 7 电压, 单位 mV, 2 Bytes, 地址偏移 24
16.    uint16_t CellVoltage_8;         //电芯 8 电压, 单位 mV, 2 Bytes, 地址偏移 26
17.    uint16_t CellVoltage_9;         //电芯 9 电压, 单位 mV, 2 Bytes, 地址偏移 28
18.    uint16_t CellVoltage_10;        //电芯 10 电压, 单位 mV, 2 Bytes, 地址偏移 30
19.    uint16_t CellVoltage_11;        //电芯 11 电压, 单位 mV, 2 Bytes, 地址偏移 32
20.    uint16_t CellVoltage_12;        //电芯 12 电压, 单位 mV, 2 Bytes, 地址偏移 34
21.    uint16_t CellVoltage_13;        //电芯 13 电压, 单位 mV, 2 Bytes, 地址偏移 36
22.    uint16_t CellVoltage_14;        //电芯 14 电压, 单位 mV, 2 Bytes, 地址偏移 38
23.    uint16_t CellVoltage_15;        //电芯 15 电压, 单位 mV, 2 Bytes, 地址偏移 40
24.    uint16_t CellVoltage_16;        //电芯 16 电压, 单位 mV, 2 Bytes, 地址偏移 42
25.    int32_t ChargeCurrent;           //充电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 44
26.    int32_t DisChargeCurrent;        //放电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 48
27.    uint32_t FullChargeCapacity;     //满充容量, 单位 mAh, 4 Bytes, 地址偏移 52
28.    uint32_t RemainCapacity;         //剩余容量, 单位 mAh, 4 Bytes, 地址偏移 56
29.    uint16_t CycleCount;             //循环次数, 2 Bytes, 地址偏移 60
30.    uint8_t Temperature_1;          //温度 1, 单位°C, 偏移 40, 1 Byte, 地址偏移 62
31.    uint8_t Temperature_2;          //温度 2, 单位°C, 偏移 40, 1 Byte, 地址偏移 63
32.    uint8_t Temperature_3;          //温度 3, 单位°C, 偏移 40, 1 Byte, 地址偏移 64
```

```
33.    uint8_t Temperature_4;           //温度 4, 单位°C, 偏移 40, 1 Byte, 地址偏移 65
34.    uint8_t SOC;                     //剩余电量, 1 Byte, 地址偏移 66
35.    uint8_t SOH;                     //电荷寿命, 1 Byte, 地址偏移 67
36.    uint8_t AFE_Status;              //AFE 状态, 1 Byte, 地址偏移 68
37.    uint8_t Working_State;           //工作状态, 1 Byte, 地址偏移 69
38.    uint16_t MaxChargeTime;           //最大充电持续时间, 单位 min, 地址偏移 70
39.    uint16_t MaxBetweenChargeTime;    //最大充电间隔时间, 单位 min, 地址偏移 72
40.    uint16_t MaxBetweenDisChargeTime; //最大放电持续时间, 单位 min, 地址偏移 74
41.    uint16_t LastBetweenDisChargeTime; //最大放电间隔时间, 单位 min, 地址偏移 76
42.    uint16_t MaxUVPTime;             //最大欠压保护间隔时间, 单位 min, 地址偏移 78
43.    RTC_Struct_t RTC;                //实时时钟, 8 Bytes, 地址偏移 80
44.    RTC_Struct_t LastChargeTime;      //最近充电时钟, 8 Bytes, 地址偏移 88
45.    RTC_Struct_t LastDisChargeTime;    //最近放电时钟, 8 Bytes, 地址偏移 96
46.    RTC_Struct_t LastUVP_RTC;         //最近欠压保护时钟, 8 Bytes, 地址偏移 104
47.    RTC_Struct_t LastUVP_Active_RTC;  //最近欠压保护激活时钟, 8 Bytes, 地址偏移 112
48.    RTC_Struct_t LastFCC_Update_RTC  //最近满充容量更新时钟, 8 Bytes, 地址偏移 120
49. }BMS_ErrorLogSaveInfo_Struct_t;
50.
51. //RTC 数据结构
52. typedef struct
53. {
54.    uint8_t RS1;                       //预留, 0x00
55.    uint8_t Year;                      //年
56.    uint8_t Mouth;                    //月
57.    uint8_t Date;                     //日
58.    uint8_t RS2;                       //预留, 0x00
59.    uint8_t Hour;                     //时
60.    uint8_t Minute;                   //分
61.    uint8_t Second;                   //秒
62. }
```

MOTINOVA 大牙盘系列中置系统通信协议

1 系统组成

MC: 电机控制器 Motor Controller

BMS: 电池管理系统 Battery Management System

OBC: 车载计算机 On Board Computer

CDL: 通讯适配器 CAN Dongle

APP: 用户程序 Application

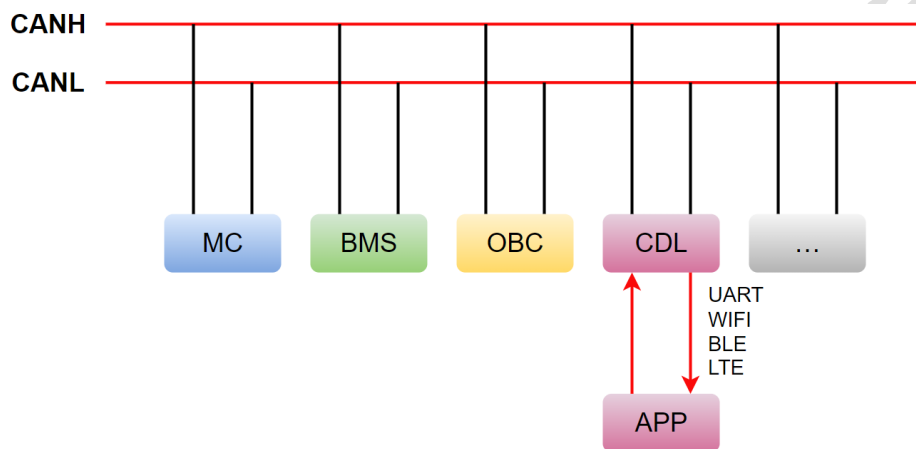


图1 系统通信接口示意图

2 硬件接口

2.1 通信接口参考电路

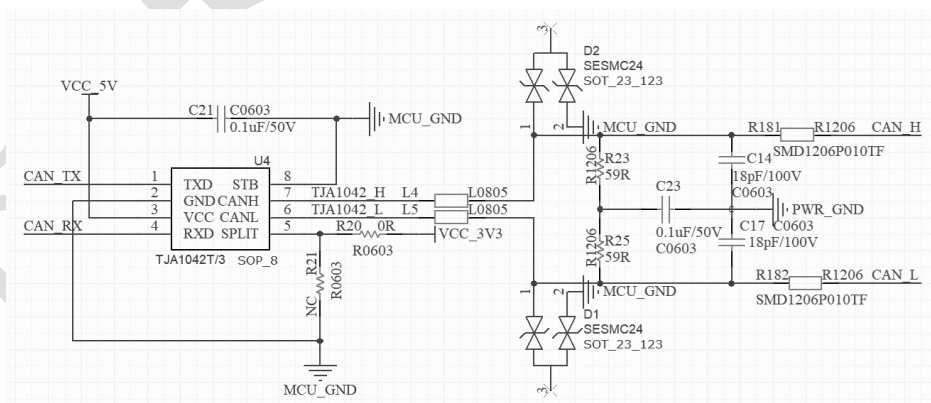


图2 通信接口参考电路

2.2 注意事项

图 2 展示的为各组件 CAN 通信接口参考电路, 设计时注意以下几点:

- 1) ESD 保护电路设计需考虑电源线与通信线发生短路时, 不会导致 ESD 保护器件

被击穿, 电源线最高电压按照 48V 系统最高工作电压 54.6V 测试;

2) 终端匹配电阻设计在 BMS 和 OBC 内部, 其它部件预留;

3) BMS 需考虑通信接口电气隔离设计。

3 通信协议规则

3.1 软件配置

CAN 控制器推荐配置如下:

时钟频率: 1500 KHz

SJW: 1

BS1: 4

BS2: 1

波特率: 250 Kbps

3.2 数据帧封装格式

所有传输的数据按照标准帧进行封装, 将传输数据按顺序填入数据帧中。以下对数据帧格式、ID 分配、分包方式进行介绍。

3.2.1 数据帧格式

协议描述了每帧数据内容, 包括帧头、帧模式、命令段长度、命令字、数据段、校验位、帧尾。每帧格式如下:

表1 数据帧格式

帧头	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中:

8) 帧头固定为 0x55 0xAA, 帧尾固定为 0xF0;

9) 帧模式包含读 0x11, 写 0x16, 和上报 0x0C, 任何设备收到写指令时, 需根据数据来源发送通用反馈指令;

10) LENGTH 命令段总长度, 占用 1 字节, 有效值为 0x02~0xFF;

11) COMMAND 为命令字, 占用 2 个字节, 第 1 字节为命令字序号, 第 2 字节为数据段长度;

12) DATA 为数据段, 长度为 LENGTH - 2;

13) CRC 为校验位, 占用 4 字节, 由帧头开始, CAN_ID 插入到帧头和帧模式之间, 计算到数据段最后一个字节, 计算方法见附录 1, 计算结果高字节在前, 如:

CAN_ID 为 0x0712, 数据帧为 55 AA 11 03 22 01 00 CRC1 CRC2 CRC3 CRC4 F0,
CRC 计算函数输入数据为 55 AA 07 12 11 03 22 01 00, 计算结果依次由高到低写入 CRC1、CRC2、CRC3、CRC4;

14) 数据段发送时, 采用小端模式。

3.2.2 ID 分配

表2 ID 分配

MC	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x710		0x712	0x713	0x715
BMS	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x720	0x721		0x723	0x725
OBC	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x730	0x731	0x732		0x735
CDL	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x750	0x751	0x752	0x753	

3.2.3 分包方式

对于长度超过 8bytes 的数据帧, 按照 8+N 的方式分包, 每个数据包填入相同的 ID 号, 如下表所示:

表3 封装方式

包序号	1				N	
内容	ID	Byte1~Byte8	ID	Byte1~Byte8	ID	Byte1~ByteN

3.3 CDL 数据格式转换规则

3.3.1 接收处理

CDL 接收到 CAN 总线数据后, 按照下表格式插入 ID, 完成转发:

表4 CDL 数据转换格式

帧头	CAN ID	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	ID	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中 ID 占用 2 字节。

3.3.2 发送处理

CDL 接收到其它设备按照上表格式发送的数据后, 取数据包中的 ID 作为当前发送数据帧的 CAN ID, 并以表 1 格式完成分包转发。

3.3.3 CDL 专用指令

用于 CDL 与 APP 之间的数据交互, 此部分指令不需要通过 CDL 发送到 CAN 总线。

3.3.3.1 CDL 在线检测

在其它 UART、WIFI、BLE 等接口方式的设备 APP 通过 CDL 访问 CAN 总线上 MC、BMS、OBC、OBC 等设备时, 需要先确认 CDL 连接是否正常。

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 收到 CDL 返回表示 CDL 在线, 超时 2s 未收到 CDL 返回表示 CDL 离线:

[55 AA 07 FF 11 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 返回:

[55 AA 07 FF 0C 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.2 设备开关机

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 并显示开机结果:

[55 AA 07 FF 16 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

2) CDL 接收到指令后使能开机信号, 同时返回:

[55 AA 07 FF 0C 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

3.3.3.3 上报系统供电方式

CDL 定时 2s 发送一次系统供电方式, APP 显示供电方式, 数据如下:

[55 AA 07 FF 0C 03 33 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下:

0x00: 适配器未接入、外部设备无电池

0x01: 适配器未接入、外部设备含电池

0x11: 适配器接入、外部设备含电池

0x10: 适配器接入、外部设备无电池

3.3.3.4 CDL 复位

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 并显示复位结果:

[55 AA 07 FF 11 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 收到指令后返回指令, 然后复位:

[55 AA 07 FF 0C 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.5 设备校验

通过其它 UART、WIFI、BLE 等接口方式的设备 APP 连接 CDL, 需要对设备与 CDL 进

行双向校验。

1) APP 发送随机码和校验密钥给到 CDL:

[55 AA 07 FF 16 16 55 14 随机码 密钥 CRC1 CRC2 CRC3 CRC4 F0]

其中随机码长度为 12Bytes, 密钥长度为 8Bytes。

2) CDL 根据随机码和密钥进行计算后, 将校验码发送给 APP, 由 APP 判断校验是否成功:

[55 AA 07 FF 0C 1D 55 1B 校验结果 版本号 CRC1 CRC2 CRC3 CRC4 F0]

其中校验结果长度为 12Bytes, 版本号为 15Bytes, 格式为 Vx.x.x_YYYYMMDD。

4 通信内容

4.1 MC 命令字定义

表5 MC 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x710	0x0C	0x1020	MC 运行信息 (收到 OBC 控制 指令时返回)	车速:2bytes 输出转速:2bytes 电功率:2bytes 母线电压:2bytes 母线电流:2bytes 踢频:1byte 踩踏力矩:1byte 踩踏方向:1byte 助力档位:1byte 大灯状态:1byte 剩余电量:1byte 续航里程:2bytes 预留:2bytes 平均功耗:1byte PCB 温度:1bytes 绕组温度:1bytes MCU 温度:1bytes	0.1km/h 1rpm 1W 1mV 1mA 1rpm 1N.m 0-正, 1-反, 2-停止 0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开 1%, 无效为 0xFF 1km, 无效为 0xFFFF 0x00 0.01Ah/km, 无效为 0xFF +40℃ +40℃ +40℃

				开机后骑行里程:2bytes 开机后骑行时间:2byte 预留:4byte	0.1km 1s 填充 0x00
0x710	0x0C	0x1104	MC 故障码 (存在故障时 200ms 自动发送, 故障消失后停止发送)	0:无故障 低 16 位: 0x0001:过流保护 0x0002:低压保护 0x0004:过压保护 0x0008:堵转保护 0x0010:过热保护 0x0020:SPS 故障 0x0040:TQS 故障 0x0080:霍尔故障 0x0100:马达缺相 0x0200:NTC 故障 0x0400:BMS 校验失败 0x1000:OBC 校验失败 0x2000:MCU 故障 0x4000:踏频故障 0x8000:指拨故障 高 16 位: 0x0001:MOS 短路 0x0002:电压异常 0x0004:电路故障 0x0040:马达失速故障 0x0080:预留故障	按位或输出, 0-正常, 1-故障
0x710	0x0C	0x1240	电机版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20 FW 命名格式为 Vxrxxr_x_YYYYMMDD
0x710	0x0C	0x1305	关机就绪 (返回指令)	ASCII 字符	READY
0x710	0x0C	0x1401	在线检测结果 (返回指令)	0x00:系统正常 0x01:BMS 离线 0x02:OBC 离线 0x04:预留	按位或输出, 0-在线, 1-离线

				0x08: 预留 0x10: 预留 0x20: 预留 0x40: 预留 0x80: 预留	
0x710	0x0C	0x1510	骑行历史信息 (返回指令)	ODO 里程: 4bytes ODO 时间: 4bytes TRIP 里程: 4bytes TRIP 时间: 4bytes	0.1km 1min 0.1km 1min
0x710	0x0C	0x1720	特殊性说明	ASCII 字符	结束符为',', 无效填充 0x20
0x710	0x0C	0x1808	关机指令 (控制器关机前发送) [适用 V2 电机]	ASCII 字符	SHUTDOWN
发送给 BMS					
0x712	0x11	0x3009	BMS 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x712	0x11	0x3100	查询 BMS 物理 ID (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3200	查询 BMS 校验码 (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3300	查询 BMS 设计信息 (主动发送, 收到返回或超时停止)		
发送给 OBC					
0x713	0x11	0x5009	OBC 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x713	0x11	0x5100	查询 OBC 物理 ID (主动发送, 收到返回或超时停止)		

0x713	0x11	0x5200	查询OBC校验码 (主动发送, 收到返回或超时停止)		
0x713	0x0C	0x5303	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x713	0x0C	0x5408	返回OBC用户参数	轮胎默认周长:1byte 启动模式:1byte 限速:1byte 周长微调:1byte 助力方案:1byte 自动关机时间:1byte 预留:2bytes	1cm 1-柔和, 2-正常, 3-强劲 1km/h ±10cm, 有符型 1-2 1min 填充 0x00
发送给 CDL					
0x715	0x0C	0xA408	校验码计算密钥 (返回指令)	密钥:8bytes	ASCII
0x715	0x0C	0xA520	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 产品标识:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD MM_MC1. CITY MM_MT1. MTB MM_MG1. CARGO
0x715	0x0C	0xA610	自定义可存储字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA710	自定义可存储字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA810	自定义可存储字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA903	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x715	0x0C	0xAB88	返回存储器指定起始和结束地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 填充 0xFF
0x715	0x0C	0xB226	马达信息	极对数:2bytes	1-65535

			(返回指令)	电阻:2bytes d 轴电感:2bytes q 轴电感:2bytes 永磁体磁链:2bytes Id 最大值:2bytes Id 最小值:2bytes 额定转速:2bytes 额定功率:2bytes 额定电流:2bytes 额定电压:2bytes 惯量:2bytes 最大转矩:2bytes 预留:12byte	0.1 mOhm 0.01 uH 0.01 uH 0.001 mWb 0.01 A 0.01 A 1 rpm 1 W 0.01 A 0.1 V $10^{-7} \text{ kg}\cdot\text{m}^2$ 0.1 Nm 填充 0x00
0x715	0x0C	0xB31A	整车信息	轮胎周长:2bytes 电控传动比:2bytes 助力最大限速:2bytes 推行模式限速:2bytes 前牙盘 T 数:2bytes 后牙盘 T 数:2bytes 助力方案 1 编号:2bytes 助力方案 2 编号:2bytes 前后灯电压:2bytes 周长微调值:2bytes 启动模式: 2bytes 自动关机时间:2bytes 预留: 2bytes	1 cm 0.1 1 km/h 1 km/h 1 1 每 2 bit 代表 1 个档位的助力曲线, 由低到高依次表示 1-5 档 2 bit 代表 1 个档位的助力曲线, 由低到高依次表示 1-5 档 高位: 前灯, 6:6V, 12:12V 低位: 后灯, 6:6V, 12:12V $\pm 10\text{cm}$, 有符型 1-柔和, 2-正常, 3-强劲 1min, 设置 0 为不自动关机 0x00
0x715	0x0C	0xB420	控制器参数	位置传感器零点:2bytes 位置传感器当前零点:2bytes	Q14 Q14

MOTINOVA				文件名称：MOTINOVA E-Bike 电气系统通信协议		密级：秘密
				文件编号：[填入文件编号]	版本：V4.2	共 133 页 第 63 页
					峰值电流:2bytes	0.01 A
					电流保护阈	0.01 A
					值:2bytes	
					最高档电压保护阈	0.1 V
					值:2bytes	
					最高档欠压保护阈	0.1 V
					值:2bytes	
					超速保护阈	1 rpm
					值:2bytes	
					温度保护阈	1 Ce
					值:2bytes	
					温度保护恢复阈	1 Ce
					值:2bytes	
					温度降额启动阈	1 Ce
					值:2bytes	
					预留:12bytes	填充 0x00
0x715	0x0C	0xB528	传感器参数	力矩传感器出厂零	Q12	
				点值:2bytes		
				力矩传感器历史零	Q12	
				点值 1:2bytes		
				力矩传感器历史零	Q12	
				点值 2:2bytes		
				力矩传感器历史零	Q12	
				点值 3:2bytes		
				力矩传感器最新零	Q12	
				点值:2bytes		
				力矩传感器最大转	0.1 Nm	
				矩:2bytes		
				第一阶段转矩点加	0.1 Nm	
				载值:2bytes		
				第一阶段转矩点采	Q12	
				集值:2bytes		
				第二阶段转矩点加	0.1 Nm	
				载值:2bytes		
				第二阶段转矩点采	Q12	
				集值:2bytes		
				第三阶段转矩点加	0.1 Nm	
				载值:2bytes		
				第三阶段转矩点采	Q12	
				集值:2bytes		
				第四阶段转矩点加	0.1 Nm	
				载值:2bytes		
				第四阶段转矩点采	Q12	
				集值:2bytes		

				踏频传感器一圈脉冲数:2bytes	1
				车速传感器一圈脉冲数:2bytes	1
				预留:8bytes	填充 0x00
				零速启动增益:2bytes	0.1
				巡航启动增益:2bytes	0.1
				助力转矩曲线编号:2bytes	助力转矩曲线编号 1-15
				助力踏频曲线编号:2bytes	助力踏频曲线编号 1-5
				转矩曲线.a:4bytes	
				转矩曲线.b:4bytes	
				转矩曲线.c:4bytes	
				转矩曲线.d:4bytes	
				踏频曲线.a:4bytes	
				踏频曲线.b:4bytes	
				踏频曲线.c:4bytes	
				踏频曲线.d:4bytes	
				助力启动阈值:2bytes	0.1 Nm
				助力停止阈值:2bytes	0.1 Nm
				启动时电流增长阶梯:2bytes	Q14
				启动对应踏频脉冲数:2bytes	1
				转矩滤波对应踏频脉冲数:2bytes	1
				待速转速:2bytes	1 rpm
				待速最大电流:2bytes	0.01 A
				车速限幅启动阈值:2bytes	0.1km

MOTINOVA		文件名称：MOTINOVA E-Bike 电气系统通信协议				密级：秘密	
		文件编号：[填入文件编号]			版本：V4.2	共 133 页 第 65 页	
	0x715	0x0C	0xB74C	历史信息	车速限幅停止阈值:2bytes	0.1 km	
					踏频占比:2bytes		
					预留:16bytes	填充 0x00	
					开机次数:2bytes	1	
					使用时间 H:2bytes	1 min	
					使用时间 L:2bytes	1 min	
					NTC 温度传感器最高温:2bytes	1 Ce	
					NTC 温度传感器最高温:2bytes	1 Ce	
					NTC 温度传感器最低温:2bytes		
					硬件过流保护次数:2bytes	1	
					软件过流保护次数:2bytes	1	
					过温保护次数:2bytes	1	
					堵转次数:2bytes	1	
					缺相次数:2bytes	1	
					过压次数:2bytes	1	
					欠压次数:2bytes	1	
					通讯超时次数:2bytes	1	
					平均功耗 1:2bytes	0.01 Ah/km	
					平均功耗 2:2bytes	0.01 Ah/km	
					平均功耗 3:2bytes	0.01 Ah/km	
					平均功耗 4:2bytes	0.01 Ah/km	
					平均功耗 5:2bytes	0.01 Ah/km	
					ODO 里程 H:2bytes	0.1 km	
					ODO 里程 L:2bytes	0.1 km	
					ODO 时间 H:2bytes	1 min	
					ODO 时间 L:2bytes	1 min	
					TRIP 里程 H:2bytes	0.1 km	
					TRIP 里程 L:2bytes	0.1 km	
					TRIP 时间 H:2bytes	1 min	
					TRIP 时间 L:2bytes	1 min	
					力矩传感器故障次数:2bytes	1	
					踏频传感器故障次数:2bytes	1	
					车速传感器故障次数:2bytes	1	
					位置传感器故障次数:2bytes	1	

				预留:16bytes	填充 0x00
0x715	0x0C	0xB83A	调试参数	整体运行模 式:2bytes 位置获取模 式:2bytes 采样模式:2bytes 旋转方向:2bytes 定位电流:2bytes 拖拽电压:2bytes 拖拽电流:2bytes 拖拽频率:2bytes 加速斜率:2bytes 减速斜率:2bytes 转速环控制器带 宽:2bytes 转速环控制器 m:2bytes 电流环控制器带 宽:2bytes 电流环控制器 m:2bytes 磁链观测器带 宽:2bytes 磁链观测器 m:2bytes 锁相环带 宽:2bytes 锁相环 m:2bytes 惯量:2bytes PWM 最大占空 比:2bytes PWM 七段式切五段 式阈值:2bytes 功率限幅 值:2bytes 功率 Error:2bytes 功率限幅 Kp:2bytes 功率限幅 Ki:2bytes 预留:8bytes	填充 0x00
0x715	0x0C	0xB904	存储标志	EEPROM 存储 FLAG:2bytes SIP 偏置校准	0-未存储, 1-已存储 0-未存储, 1-已存储

FLAG: 2bytes

4.2 BMS 命令字定义

表6 BMS 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x720	0x0C	0x1010	电池运行信息 (返回指令)	电压: 2bytes 平均电流: 2bytes 剩余容量: 2bytes 满充容量: 2bytes 电芯温度: 1byte 剩余电量: 1byte 运行状态: 1byte (按位或输出) SOH: 1byte 预留: 4bytes	1mV 1mA, 有符型, 放电 为负, 充电为正 1mAh 1mAh +40℃ 0~100% 0x00: 休眠 0x01: 充电器接入 0x02: 预留 0x04: 预留 0x08: 预留 0x10: 预留 0x20: 预留 0x40: 预留 0x80: 预留 0~100% 填充 0x00
0x720	0x0C	0x1120	电芯电压 (返回指令)	Cell_1: 2bytes Cell_16: 2bytes 不足部分填充 0x00	1mV 1mV
0x720	0x0C	0x1204	BMS 故障码 (存在故障时 200ms 自动发 送, 故障消失后 停止发送)	高 16 位: 0x0001: 充电过压 警告 0x0002: 放电低压 警告 0x0004: 充电过流 警告 0x0008: 放电过流 警告 0x0010: 充电高温 警告 0x0020: 充电低温 警告 0x0040: 放电高温 警告	按位或输出, 0-正 常, 1-故障

				0x0080:放电低温警告 0x0100:MOS 高温警告 低 16 位: 0x0001:二级放电过流保护 0x0002:充电过流保护 0x0004:短路保护 0x0008:过放保护 0x0010:过充保护 0x0020:放电低温保护 0x0040:放电高温保护 0x0080:充电低温保护 0x0100:充电高温保护 0x0200:放电 MOS 故障 0x0400:充电 MOS 故障 0x0800:温度传感器故障 0x1000:预留 0x2000:一级过流保护 0x4000:AFE 故障 0x8000:MCU 故障	
0x720	0x0C	0x1308	关机指令 (主动发送,收到返回或超时停止)	ASCII 字符	常按开关键 3s 或监测到母线电流小于 50mA 且 CAN 总线空闲持续 30min 后,执行发送 SHUTDOWN,延时 1s 后,关闭放电开关
0x720	0x0C	0x1410	电池设计信息 (返回指令)	设计容量:2bytes 设计电压:1byte 电芯型号:8bytes 预留:5bytes	1mAh 1V ASCII, 0x2E 结束, 无效填充 0x20 填充 0x00
0x720	0x0C	0x1540	电池版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、 SN、HW、FW;

保密文件， 禁止外传

				SOH:1byte 预留:5bytes	0~100% 填充 0x00
0x725	0x0C	0x5120	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x725	0x0C	0x5210	自定义可存储 字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x725	0x0C	0x5310	自定义可存储 字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x725	0x0C	0x5410	自定义可存储 字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x725	0x0C	0x5503	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x725	0x0C	0x5688	存储器指定起 始和结束地址 的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF

4.3 OBC 命令字定义

表7 OBC 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x730	0x0C	0x1140	版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、 SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20 FW 命名格式为 Vxrxrx_YYYYMMDD
0x730	0x0C	0x120C	物理 ID (返回指令)	ID:12bytes	不足位填充 1
0x730	0x0C	0x130C	校验码 (返回指令)	校验码:12bytes	
0x730	0x0C	0x1405	关机就绪 (返回指令)	ASCII 字符	READY
0x730	0x0C	0x1504	故障代码 (存在故障时 200ms 自动发	高 16 位:0x0000 低 16 位 0x0000:无故障	按位或输出, 0-正 常, 1-故障

			送,故障消失后 停止发送)	0x0001:+键失效 0x0002:-键失效 0x0004:i 键失效 0x0008:灯键失效 0x0010:Walk 键失 效 0x0020:电源键失 效 0x0400:MC 通讯异 常	
发送给 MC					
0x731	0x0C	0x3002	控制电机指令 (定时上传)	助力档位:1byte 0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 大灯状态:1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x731	0x0C	0x3105	在线检测反馈 (返回指令)	ASCII 字符	READY
0x731	0x11	0x3300	查询用户参数 (主动发送,收 到返回或超时 停止)		
0x731	0x16	0x3408	设置用户参数 (主动发送,收 到返回或超时 停止)	周长微调值:1byte 启动模式:1byte 助力方案:1byte 自动关机时 间:1byte 预留:6bytes	有符型, 单位 1cm, ±10.0cm 1-柔和, 2-正常, 3- 强劲 1-2 1min, 设置为 0 时控 制器不自动关机 填充 0x00
0x731	0x11	0x3500	查询骑行历史 信息 (主动发送,收 到返回或超时 停止)		
0x731	0x16	0x3605	清除电机 TRIP 信息(主动发 送,收到返回或 超时停止)	ASCII 字符	CLEAR

0x731	0x11	0x3900	OBC 读取电机版本信息		
0x731	0x0C	0x3A02	OBC 发送按键状态 (主动发送, 按键按下触发发送一次)	Byte1: 预留 Byte2: Bit7:0 Bit6:0 Bit5: 电源键 Bit4: +键 Bit3: -键 Bit2: Walk 键 Bit1: Light 键 Bit0: M/Set 键	0x00 按键状态按位或填充, 1-按下, 0-弹起
发送给 BMS					
0x732	0x11	0x5000	查询 BMS 运行信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5100	查询 BMS 版本信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5200	查询 BMS 设计信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5300	查询 BMS 电芯电压 (主动发送, 收到返回或超时停止)		
发送给 CDL					
0x735	0x0C	0x9003	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x735	0x0C	0x9101	发送按键状态 (定时上传)	+键: Bit0 -键: Bit1 i 键: Bit2 灯键: Bit3 Walk 键: Bit4 电源键: Bit5	0-松开 1-按下 多余位填充 0
0x735	0x0C	0x9310	运行历史信息 (返回指令)	运行时间: 4bytes 开机次数: 4bytes 预留: 8bytes	1min 次 填充 0x00

0x735	0x0C	0x9420	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x735	0x0C	0x9510	自定义可存储 字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9610	自定义可存储 字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9710	自定义可存储 字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9888	存储器指定起 始和结束地址 的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF

4.4 CDL 命令字定义

表8 CDL 命令字定义

ID	模式	命令字	功能	数据段	备注
发送给 MC					
0x751	0x11	0x1000	查询电机校验 码密钥		
0x751	0x16	0x1108	写入电机校验 码密钥		
0x751	0x11	0x1200	查询电机版本 信息		
0x751	0x11	0x1300	查询自定义可 存储字符串 1		
0x751	0x16	0x1410	写入自定义可 存储字符串 1	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1500	查询自定义可 存储字符串 2		
0x751	0x16	0x1610	写入自定义可 存储字符串 2	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1700	查询自定义可 存储字符串 3		
0x751	0x16	0x1810	写入自定义可 存储字符串 3	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x16	0x1901	写入电机工作 模式	工作模式:1byte	0-运行, 1-配置

0x751	0x11	0x1E00	查询电机历史信息		
0x751	0x11	0x1F00	查询电机生产信息		
0x751	0x11	0x2100	查询设备在线检测结果		
0x751	0x16	0x2210	写入电机 Mode (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x751	0x16	0x2310	写入电机 SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x751	0x16	0x2420	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 产品标识:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD MM_MC1. CITY MM_MT1. MTB MM_MG1. CARGO
0x751	0x16	0x2505	复位指令	ASCII 字符串	RESET
0x751	0x16	0x2605	数据清除	ASCII 字符串	CLEAR
0x751	0x16	0x2708	参数还原	ASCII 字符串	RECOVERY
0x751	0x16	0x2802	控制指令	助力档位:1byte 大灯状态:1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x751	0x16	0x2C01	设置电机转速百分比	百分比:1byte	5~100
0x751	0x11	0x2D08	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤128Bytes
0x751	0x11	0x2E00	查询骑行历史信息		
0x751	0x16	0x3909	历史信息清除	ASCII	LOG CLEAR
0x751	0x11	0x3A00	读取马达信息		
0x751	0x16	0x3B28	写入马达信息	存储标志:2bytes 极对数:2bytes 电阻:2bytes d 轴电感:2bytes	0-不存储, 1-存储 1-65535 0.1 mOhm 0.01 uH

				q 轴电感:2bytes 永磁体磁链:2bytes Id 最大值:2bytes Id 最小值:2bytes 额定转速:2bytes 额定功率:2bytes 额定电流:2bytes 额定电压:2bytes 惯量:2bytes 最大扭矩:2bytes 预留 0:12byte	0.01 uH 0.001 mWb 0.01 A 0.01 A 1 rpm 1 W 0.01 A 0.1 V $10^{-7} \text{ kg}\cdot\text{m}^2$ 0.1 Nm 填充 0x00
0x751	0x11	0x3C00	读取整车信息		
0x751	0x16	0x3D1C	写入整车信息	存储标志:2bytes 轮径:2bytes 电控传动比:2bytes 助力最大限 速:2bytes 推行模式限 速:2bytes 前牙盘 T 数:2bytes 后牙盘 T 数:2bytes 助力方案 1 编 号:2bytes 助力方案 2 编 号:2bytes 前后灯电压:2bytes 周长微调值:2bytes 启动模式: 2bytes 自动关机时 间:2bytes 预留:2bytes	0-不存储, 1-存储 1 cm 0.1 1 km/h 1 km/h 1 1 每 2 bit 代表 1 个档 位的助力曲线, 由低 到高依次表示 1-5 档 2 bit 代表 1 个档位 的助力曲线, 由低到 高依次表示 1-5 档 高位: 前灯, 6:6V, 12:12V 低位: 后灯, 6:6V, 12:12V $\pm 10\text{cm}$, 有符型 1-柔和, 2-正常, 3- 强劲 1min, 设置 0 为不自 动关机 0x00
0x751	0x11	0x3E00	查询控制器参 数		
0x751	0x16	0x3F22	写入控制器参 数	存储标志:2bytes 位置传感器零	0-不存储, 1-存储 Q14

				点:2bytes 位置传感器当前零 点:2bytes 峰值电流:2bytes 电流保护阈 值:2bytes 最高档电压保护阈 值:2bytes 最高档欠压保护阈 值:2bytes 超速保护阈 值:2bytes 温度保护阈 值:2bytes 温度保护恢复阈 值:2bytes 温度降额启动阈 值:2bytes 预留:12bytes	Q14 0.01 A 0.01 A 0.1 V 0.1 V 1 rpm 1 Ce 1 Ce 1 Ce 填充 0x00
0x751	0x11	0x4000	查询传感器参 数		
0x751	0x16	0x4104	写入力矩传感 器加载值	标定点序号:2ytes 加载值:2bytes	1 0.1 Nm
0x751	0x16	0x420E	写入其它传感 器参数	存储标志:2bytes 踏频传感器一圈脉冲 数:2bytes 车速传感器一圈脉冲 数:2bytes 预留:8bytes	0-不存储, 1-存储 1 1 填充 0x00
0x751	0x11	0x4304	查询助力参数	助力转矩曲线编 号:2bytes 助力踏频曲线编 号:2bytes	1 1
0x751	0x16	0x4450	写入助力参数	存储标志:2bytes 填充:2bytes 零速启动增 益:2bytes 巡航启动增 益:2bytes 助力转矩曲线编 号:2bytes 助力踏频曲线编 号:2bytes 转矩曲线. a:4bytes	0-不存储, 1-存储 0x00 0.1 0.1 助力转矩曲线编号 1-15 助力踏频曲线编号 1-5

				转矩曲线. b:4bytes 转矩曲线. c:4bytes 转矩曲线. d:4bytes 踏频曲线. a:4bytes 踏频曲线. b:4bytes 踏频曲线. c:4bytes 踏频曲线. d:4bytes 助力启动阈 值:2bytes 助力停止阈 值:2bytes 启动时电流增长阶 梯:2bytes 启动对应踏频脉冲 数:2bytes 转矩滤波对应踏频脉 冲数:2bytes 待速转速:2bytes 待速最大电 流:2bytes 车速限幅启动阈 值:2bytes 车速限幅停止阈 值:2bytes 踏频占比:2bytes 预留:16bytes	0.1 Nm 0.1 Nm Q14 1 1 1 rpm 0.01 A 0.1km 0.1 km 填充 0x00
0x751	0x11	0x4500	查询调试参数		
0x751	0x16	0x463C	写入调试参数	存储标志:2bytes 整体运行模 式:2bytes 位置获取模 式:2bytes 采样模式:2bytes 旋转方向:2bytes 定位电流:2bytes 拖拽电压:2bytes	0- 不存储, 1-存储

MOTINOVA		文件名称: MOTINOVA E-Bike 电气系统通信协议			密级: 秘密
		文件编号: [填入文件编号]		版本: V4.2	共 133 页 第 78 页
				拖拽电流:2bytes 拖拽频率:2bytes 加速斜率:2bytes 减速斜率:2bytes 转速环控制器带 宽:2bytes 转速环控制器 m:2bytes 电流环控制器带 宽:2bytes 电流环控制器 m:2bytes 磁链观测器带 宽:2bytes 磁链观测器 m:2bytes 锁相环带宽:2bytes 锁相环 m:2bytes 惯量:2bytes PWM 最大占空 比:2bytes PWM 七段式切五段式 阈值:2bytes 功率限幅值:2bytes 功率 Error:2bytes 功率限幅 Kp:2bytes 功率限幅 Ki:2bytes 预留:8bytes	
					填充 0x00
0x751	0x16	0x472E	写入力矩传感器参数	存储标志:2bytes 出厂零点:2bytes 历史零点 1:2bytes 历史零点 2:2bytes 历史零点 3:2bytes 最新零点:2bytes 最大扭矩:2bytes 第一阶段加载 值:2bytes 第一阶段采集 值:2bytes 第二阶段加载	0-不存储, 1-存储 Q12 Q12 Q12 Q12 Q12 0. 1N. m 0. 1N. m Q12 0. 1N. m

				值:2bytes 第二阶段采集	Q12
				值:2bytes 第三阶段加载	0. 1N. m
				值:2bytes 第三阶段采集	Q12
				值:2bytes 第四阶段加载	0. 1N. m
				值:2bytes 第四阶段采集	Q12
				值:2bytes 预留:16bytes	填充 0x00
0x751	0x11	0x4800	读取存储标志		
0x751	0x16	0x4906	写入存储标志	存储标志:2bytes EEPROM 存储 FLAG:2bytes SIP 偏置校准存储 FLAG:2bytes	0-不存储, 1-存储 0-未存储, 1-已存储 0-未存储, 1-已存储
发送给 BMS					
0x752	0x11	0x3000	查询电池物理 ID		
0x752	0x11	0x3100	查询电池校验码		
0x752	0x16	0x320C	写入电池校验码	校验码:12bytes	
0x752	0x11	0x3300	查询电池版本信息		
0x752	0x11	0x3400	查询电池运行信息		
0x752	0x11	0x3500	查询电芯电压		
0x752	0x11	0x3600	查询电池设计信息		
0x752	0x11	0x3700	查询电池生产信息		
0x752	0x11	0x3800	查询电池历史信息		
0x752	0x11	0x3900	查询自定义可存储字符串 1		
0x752	0x16	0x3A10	写入自定义可存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x11	0x3B00	查询自定义可存储字符串 2		
0x752	0x16	0x3C10	写入自定义可	ASCII 字符串	以 0x2E 结束, 无效

			存储字符串 2		填充 0x20
0x752	0x11	0x3D00	查询自定义可 存储字符串 3		
0x752	0x16	0x3E10	写入自定义可 存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x752	0x16	0x3F20	写入生产信息 (可选, 仅供 生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x752	0x16	0x4010	写入 BMS Mode (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x752	0x16	0x4110	写入 BMS SN (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x752	0x16	0x4205	复位指令	ASCII 字符串	RESET
0x752	0x11	0x4308	读取存储器指 定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤ 128Bytes
发送给 OBC					
0x753	0x11	0x5000	查询物理 ID		
0x753	0x11	0x5100	查询校验码		
0x753	0x16	0x520C	写入校验码	校验码:12bytes	
0x753	0x16	0x5300	查询生产信息		
0x753	0x11	0x5400	查询版本信息		
0x753	0x16	0x5504	设置进入/退 出按键测试模 式	ASCII 字符	STAR:进入 STOP:退出
0x753	0x11	0x5600	查询历史信息		
0x753	0x11	0x5900	查询自定义可 存储字符串 1		
0x753	0x16	0x5A10	写入自定义可 存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x11	0x5B00	查询自定义可 存储字符串 2		
0x753	0x16	0x5C10	写入自定义可 存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x11	0x5D00	查询自定义可 存储字符串 3		
0x753	0x16	0x5E10	写入自定义可 存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效 填充 0x20
0x753	0x16	0x5F20	写入生产信息	生产商:8bytes	ASCII, 0x2E 结束,

			(可选, 仅供 生产商写入)	生产地:8bytes 生产日期:8bytes 预留:8bytes	无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x753	0x16	0x6010	写入 Model (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x753	0x16	0x6110	写入 SN (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x753	0x16	0x6205	复位指令	ASCII 字符串	RESET
0x753	0x11	0x6308	读取存储器指 定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小 $\leq$ 128Bytes
0x753	0x16	0x6609	清除开机密码	ASCII 字符	PSW_CLEAR

注: CDL 发送的所有指令均按照定时 200ms 发送, 其中查询指令收到返回的信息或超时 1s 停止发送, 写入指令收到通用反馈指令或超时 1s 停止发送。

5 附录 1: CRC32 计算方法

5.1 CRC32 计算多项式表

```
47. uint32_t Crc32Table[ 256 ] =  
48. {  
49.   0x00000000, 0x04C11DB7, 0x09823B6E, 0x0D4326D9, 0x130476DC, 0x17C56B6B,  
50.   0x1A864DB2, 0x1E475005, 0x2608EDB8, 0x22C9F00F, 0x2F8AD6D6, 0x2B4BCB61,  
51.   0x350C9B64, 0x31CD86D3, 0x3C8EA00A, 0x384FBDDB, 0x4C11DB70, 0x48D0C6C7,  
52.   0x4593E01E, 0x4152FDA9, 0x5F15ADAC, 0x5BD4B01B, 0x569796C2, 0x52568B75,  
53.   0x6A1936C8, 0x6ED82B7F, 0x639B0DA6, 0x675A1011, 0x791D4014, 0x7DDC5DA3,  
54.   0x709F7B7A, 0x745E66CD, 0x9823B6E0, 0x9CE2AB57, 0x91A18D8E, 0x95609039,  
55.   0x8B27C03C, 0x8FE6DD8B, 0x82A5FB52, 0x8664E6E5, 0xBE2B5B58, 0xBAEA46EF,  
56.   0xB7A96036, 0xB3687D81, 0xAD2F2D84, 0xA9EE3033, 0xA4AD16EA, 0xA06C0B5D,  
57.   0xD4326D90, 0xD0F37027, 0xDB056FE, 0xD9714B49, 0xC7361B4C, 0xC3F706FB,  
58.   0xCEB42022, 0xCA753D95, 0xF23A8028, 0xF6FB9D9F, 0xFBB8BB46, 0xFF79A6F1,  
59.   0xE13EF6F4, 0xE5FFEB43, 0xE8BCCD9A, 0xEC7DD02D, 0x34867077, 0x30476DC0,  
60.   0x3D044B19, 0x39C556AE, 0x278206AB, 0x23431B1C, 0x2E003DC5, 0x2AC12072,  
61.   0x128E9DCF, 0x164F8078, 0x1B0CA6A1, 0x1FCD8B16, 0x018AEB13, 0x054BF6A4,  
62.   0x0808D07D, 0x0CC9CDCA, 0x7897AB07, 0x7C56B6B0, 0x71159069, 0x75D48DDE,  
63.   0x6B93DDDB, 0x6F52C06C, 0x6211E6B5, 0x66D0FB02, 0x5E9F46BF, 0x5A5E5B08,  
64.   0x571D7DD1, 0x53DC6066, 0x4D9B3063, 0x495A2DD4, 0x44190B0D, 0x40D816BA,  
65.   0xACA5C697, 0xA864DB20, 0xA527FDF9, 0xA1E6E04E, 0xBFA1B04B, 0xBB60ADFC,  
66.   0xB6238B25, 0xB2E29692, 0x8AAD2B2F, 0x8E6C3698, 0x832F1041, 0x87EE0DF6,  
67.   0x99A95DF3, 0x9D684044, 0x902B669D, 0x94EA7B2A, 0xE0B41DE7, 0xE4750050,  
68.   0xE9362689, 0xEDF73B3E, 0xF3B06B3B, 0xF771768C, 0xFA325055, 0xFE34DE2,  
69.   0xC6BCF05F, 0xC27DEDE8, 0xCF3ECB31, 0xCBFFD686, 0xD5B88683, 0xD1799B34,  
70.   0xDC3ABDED, 0xD8FBA05A, 0x690CE0EE, 0x6DCDFD59, 0x608EDB80, 0x644FC637,  
71.   0x7A089632, 0x7EC98B85, 0x738AAD5C, 0x774BB0EB, 0x4F040D56, 0x4BC510E1,  
72.   0x46863638, 0x42472B8F, 0x5C007B8A, 0x58C1663D, 0x558240E4, 0x51435D53,  
73.   0x251D3B9E, 0x21DC2629, 0x2C9F00F0, 0x285E1D47, 0x36194D42, 0x32D850F5,  
74.   0x3F9B762C, 0x3B5A6B9B, 0x0315D626, 0x07D4CB91, 0x0A97ED48, 0x0E56F0FF,  
75.   0x1011A0FA, 0x14D0BD4D, 0x19939B94, 0x1D528623, 0xF12F560E, 0xF5EE4BB9,  
76.   0xF8AD6D60, 0xFC6C70D7, 0xE22B20D2, 0xE6EA3D65, 0xEBA91BBC, 0xEF68060B,  
77.   0xD727BBB6, 0xD3E6A601, 0xDEA580D8, 0xDA649D6F, 0xC423CD6A, 0xC0E2D0DD,  
78.   0xCDA1F604, 0xC960EBB3, 0xBD3E8D7E, 0xB9FF90C9, 0xB4BCB610, 0xB07DABA7,  
79.   0xAE3AFBA2, 0xAAFBE615, 0xA7B8C0CC, 0xA379DD7B, 0x9B3660C6, 0x9FF77D71,  
80.   0x92B45BA8, 0x9675461F, 0x8832161A, 0x8CF30BAD, 0x81B02D74, 0x857130C3,  
81.   0x5D8A9099, 0x594B8D2E, 0x5408ABF7, 0x50C9B640, 0x4E8EE645, 0x4A4FFBF2,  
82.   0x470CDD2B, 0x43CDC09C, 0x7B827D21, 0x7F436096, 0x7200464F, 0x76C15BF8,  
83.   0x68860BFD, 0x6C47164A, 0x61043093, 0x65C52D24, 0x119B4BE9, 0x155A565E,  
84.   0x18197087, 0x1CD86D30, 0x029F3D35, 0x065E2082, 0x0B1D065B, 0x0FDC1BEC,  
85.   0x3793A651, 0x3352BBE6, 0x3E119D3F, 0x3AD08088, 0x2497D08D, 0x2056CD3A,  
86.   0x2D15EBE3, 0x29D4F654, 0xC5A92679, 0xC1683BCE, 0xCC2B1D17, 0xC8EA00A0,
```

MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 83 页
87. 0xD6AD50A5, 0xD26C4D12, 0xDF2F6BCB, 0xDBEE767C, 0xE3A1CBC1, 0xE760D676, 88. 0xEA23F0AF, 0xEEE2ED18, 0xF0A5BD1D, 0xF464A0AA, 0xF9278673, 0xFDE69BC4, 89. 0x89B8FD09, 0x8D79E0BE, 0x803AC667, 0x84FBDDBD, 0x9ABC8BD5, 0x9E7D9662, 90. 0x933EB0BB, 0x97FFAD0C, 0xAFB010B1, 0xAB710D06, 0xA6322BDF, 0xA2F33668, 91. 0xBCB4666D, 0xB8757BDA, 0xB5365D03, 0xB1F740B4 92. };			
5.2 CRC32 计算方法 20. uint32_t CRC32_Calculate(uint8_t *pData, uint16_t Length) 21. { 22. uint32_t nReg; 23. uint32_t nTemp = 0; 24. uint16_t i, n; 25. 26. nReg = 0xFFFFFFFF; 27. for (n = 0; n < Length; n++) 28. { 29. nReg ^= (uint32_t) pData[n]; 30. for (i = 0; i < 4; i++) 31. { 32. nTemp = Crc32Table[(uint8_t)((nReg >> 24) & 0xFF)]; 33. nReg <<= 8; 34. nReg ^= nTemp; 35. } 36. } 37. return nReg; 38. }			
保密文件， 禁止外传			

6 附录 2: 警告/故障码列表

当总线上任何一个设备上报警告/故障代码时,显示设备按照下列规则进行转换后,在屏幕上依次循环显示代码,间隔周期为 1s,显示顺序以接收到代码的先后顺序为准,当超时 3s 未接受到任何警告/故障代码时恢复原显示界面。

表9 警告/故障码列表

信号来源	显示代码	类型	描述	显示内容	解决办法
MC	10	警告	过流保护	Over Current Protect	5s 后自动恢复
MC	11	警告	低压保护	Under Voltage Protect	电池充电
MC	12	警告	过压保护	Over Voltage Protect	更换正确电池
MC	13	警告	堵转保护	Rotor Locked	关机重启
MC	14	警告	过热保护	Over Heat Protect	关机静置 30min 后使用
MC	15	故障	温度传感器故障	NTC Fault	返修
MC	16	故障	速度传感器故障	Speed Sensor Fault	检查或更换速度传感器
MC	17	故障	力矩传感器故障	Torque Sensor Fault	返修
MC	18	故障	马达霍尔传感器故障	Motor Fault	返修
MC	19	警告	BMS 校验失败	BMS Check Fault	更换电池
MC	20	警告	OBC 校验失败	OBC Check Fault	更换按键
MC	22	故障	马达缺相	PhaseLine Fault	返修
MC	23	故障	踏频传感器故障	Cadence Sensor Fault	返修
MC	24	故障	指拨故障	Gas Sensor Fault	更换指拨
MC	25	故障	MOS 短路	MOS Short Circuit	返修
MC	26	故障	电压异常波动	Bus Voltage Abnormal	更换电池
MC	27	故障	处理器故障	MCU Fault	返修
MC	28	故障	电路故障	Circuit Fault	返修
MC	31	警告	MC 校验失败	MC Check Fault	返修
MC	32	故障	马达失速	Motor Stall	返修
MC	33	故障	预留故障位	-	-
BMS	40	警告	预留故障位	-	-
BMS	41	警告	充电过压警告	Charge Over Voltage Alarm	停止充电或更换充电器
BMS	42	警告	放电低压警告	DisCharge Under Voltage Alarm	及时充电
BMS	43	警告	充电过流警告	Charge Over Current Alarm	更换充电器
BMS	44	警告	放电过流警告	Over Current Alarm	停止骑行后故障消失

BMS	45	警告	充电高温警告	Charge Over Heat Alarm	停止充电
BMS	46	警告	充电低温警告	Charge Low Temperature Alarm	停止充电
BMS	47	警告	放电高温警告	DisCharge Over Heat Alarm	关机静置 30min 后使用
BMS	48	警告	放电低温警告	DisCharge Low Temperature Alarm	建议停止使用
BMS	49	警告	MOS 高温警告	MOS Over Heat Alarm	5s 后自动恢复
OBC	60	故障	+按键失效	+ Key Fault	检查或更换按键
OBC	61	故障	-键失效	- Key Fault	检查或更换按键
OBC	62	故障	i 键失效	- Key Fault	检查或更换按键
OBC	63	故障	Light 键失效	Light Key Fault	检查或更换按键
OBC	64	故障	Walk 键失效	Walk Key Fault	检查或更换按键
OBC	65	故障	电源键失效	Power Key Fault	检查或更换按键
OBC	70	故障	MC 通讯异常	MC Communication Fault	检查线缆或更换部件
OBC	71	故障	预留故障位 1		
OBC	72	故障	预留故障位 2		

7 附录 3: 升级协议及流程

7.1 升级流程图

系统 CAN 总线上除 CDL 外所有设备均需通过 CDL 完成升级, 升级流程图如下图所示:

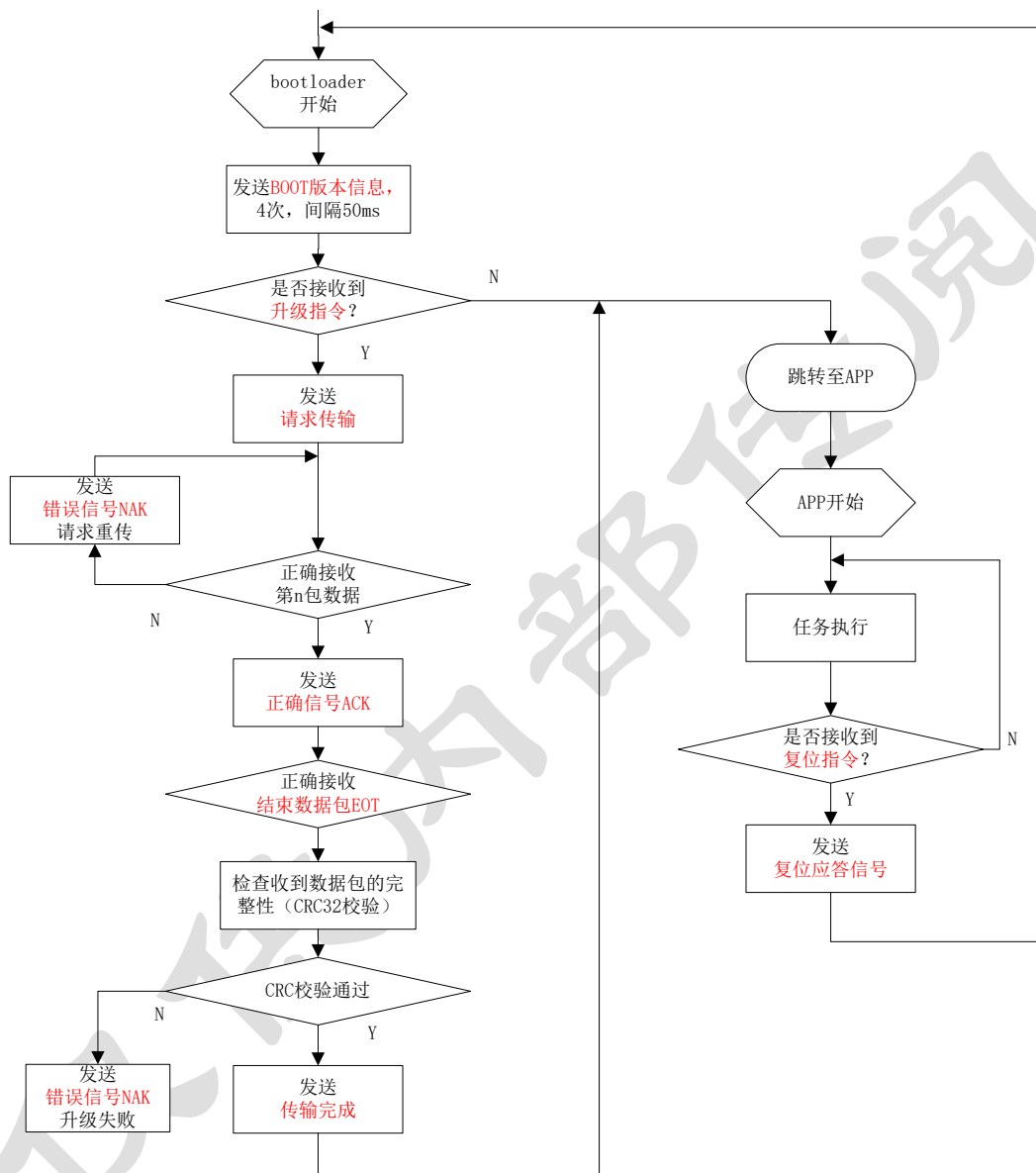


图3 升级流程图

7.2 升级协议

APP 程序中复位指令和复位应答指令参考第 3 章有关 CDL 发送给各设备的复位指令, 以及各设备发送给 CDL 的通用应答指令。

Bootloader 中的协议描述如下:

表10 CDL 发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
升级指令	MC	0x751	0x16	0x0D	0xA10B	ASCII 字符: "MC_UPD" + 数据包大小 4 字节 (如 78351B, 内容为: 0F 32 01 00)
	BMS	0x752				ASCII 字符: "BMS_UPD" + 数据包大小 4 字节
	OBC	0x753				ASCII 字符: "OBC_UPD" + 数据包大小 4 字节
数据包格式	MC	0x751	0x16	0x87	0xA385	SOH(01)+序号(1~65535)+总包数(1~65535)+数据(长度 128B, 无效填充 0xFF)
	BMS	0x752				
	OBC	0x753				
结束数据包	MC	0x751	0x16	0x03	0xA401	EOT(04)
	BMS	0x752				
	OBC	0x753				

表11 设备发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
BOOT 版本信息	MC	0x715	0x0C	0x0B	0xC109	ASCII 字符: "MC "+"Vx.x.x"
	BMS	0x725				ASCII 字符: "BMS"+"Vx.x.x"
	OBC	0x735				ASCII 字符: "OBC"+"Vx.x.x"
正确信号	MC	0x715	0x0C	0x04	0xC202	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
错误信号	MC	0x715	0x0C	0x04	0xC302	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
请求传输	MC	0x715	0x0C	0x04	0xC402	0x00 0x00
	BMS	0x725				
	OBC	0x735				
传输完成	MC	0x715	0x0C	0x04	0xC502	0x00 0x00
	BMS	0x725				
	OBC	0x735				


```
8.     if(AddressBegin <= AddressEnd)
9.     {
10.        DataLength = AddressEnd - AddressBegin+ 1;
11.        memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.        SendData(ID_MC_TO_CDL, MODE_REPORT, (0xAB08 + DataLength), (uint8_t*)Data);
13.    }
14.    }while(0);
15.    break;
16. }
```

8.4 数据结构定义

```
1. //故障日志记录信息 64 bytes, 1K 空间可以保存最近 16 条故障记录
2. typedef struct
3. {
4.     UWORD      ErrorIndex; // 故障序列号, 2bytes, 地址偏移 0
5.     UWORD      NotesInfo1; // 备注信息 1, 2bytes, 地址偏移 2
6.     UWORD      NotesInfo2; // 备注信息 2, 2bytes, 地址偏移 4
7.     UWORD      NotesInfo3; // 备注信息 3, 2bytes, 地址偏移 6
8.     ULONG      ErrorCode;  // 故障码, 4 bytes, 地址偏移 8
9.     ULONG      RunTime;    // 运行时间, 4bytes, 地址偏移 12
10.    MC_RunInfo_Struct_t RunInfo; // 运行信息 32bytes, 地址偏移 16
11.    SWORD      IqCurrentPu; // Iq 电流, 2bytes, 地址偏移 48
12.    SWORD      IqVoltagePu; // Iq 电压, 2bytes, 地址偏移 50
13.    SWORD      IdCurrentPu; // Id 电流, 2bytes, 地址偏移 52
14.    SWORD      IdVoltagePu; // Id 电压, 2bytes, 地址偏移 54
15.    UWORD      RS1;         // 预留信息 1, 2bytes, 地址偏移 56
16.    UWORD      RS2;         // 预留信息 1, 2bytes, 地址偏移 58
17.    UWORD      RS3;         // 预留信息 1, 2bytes, 地址偏移 60
18.    UWORD      RS4;         // 预留信息 1, 2bytes, 地址偏移 62
19. } ERROR_LOG; //共 64 字节
20.
21. //电机运行信息, 占用空间 32bytes
22. typedef struct
23. {
24.     uint16_t BikeSpeed; //车速 0.1km/h, 地址偏移 0
25.     int16_t MotorSpeed; //输出转速 1rpm, 地址偏移 2
26.     uint16_t Power;     //电功率 1W, 地址偏移 4
27.     uint16_t BusVoltage; //母线电压 1mV, 地址偏移 6
28.     uint16_t BusCurrent; //母线电流 1mA, 地址偏移 8
29.     uint8_t Cadence;     //踏频 1rpm, 地址偏移 10
30.     uint8_t Torque;      //踩踏力矩 1Nm, 地址偏移 11
31.    MC_CadenceDir_Struct_t CadenceDir; //踩踏方向 0-正,1-反,2-停止, 地址偏移 12
```

MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 90 页
32. uint8_t GearSt; //助力档位, 地址偏移 13 33. MC_LightSwitch_Struct_t LightSwitch; //灯开关 0xF0-关, 0xF1-开, 地址偏移 14 34. uint8_t SOC; //剩余电量 1%, 地址偏移 15 35. uint16_t RemainDistance; //续航里程 1km, 地址偏移 16 36. uint16_t ODO_Km; //总里程 1km, 地址偏移 18 37. uint8_t PowerPerKm; //平均功耗 0.01Ah/km , 地址偏移 20 38. uint8_t T_PCB; //PCB 温度 +40℃, 地址偏移 21 39. uint8_t T_Coil; //绕组温度 +40℃, 地址偏移 22 40. uint8_t T_MCU; //MCU 温度 +40℃, 地址偏移 23 41. uint16_t Ride_Km; //开机后骑行里程 0.1km, 地址偏移 24 42. uint16_t Ride_Time; //开机后骑行时间 1s, 地址偏移 26 43. uint8_t TorqueSensorData1; //力矩传感器 1 原始值 44. uint8_t TorqueSensorData2; //力矩传感器 2 原始值 45. uint8_t TorqueSensorData3; //力矩传感器 3 原始值 46. uint8_t RS ; //预留 1byte 47. }MC_RunInfo_Struct_t;			
仅供内部			
保密文件, 禁止外传			


```
10.         DataLength = AddressEnd - AddressBegin+ 1;
11.         memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.         SendData(ID_MC_TO_CDL, MODE_REPORT, (0x5608 + DataLength), (uint8_t*)Data);
13.     }
14. }while(0);
15. break;
16. }
```

9.4 数据结构定义

```
1. //故障日志记录信息 128 bytes, 2K 空间可以保存最近 16 条故障记录
2. typedef struct
3. {
4.     uint16_t Error_Index;           //故障列表索引, 2 bytes, 地址偏移 0
5.     uint16_t NotesInfo1;           //备注信息 1, 2 Bytes, 地址偏移 2
6.     uint16_t NotesInfo2;           //备注信息 2, 2 Bytes, 地址偏移 4
7.     uint16_t NotesInfo3;           //备注信息 3, 2Bytes, 地址偏移 6
8.     uint32_t ErrorCode;             //故障码, 4 bytes, 地址偏移 8
9.     uint16_t CellVoltage_1;         //电芯 1 电压, 单位 mV, 2 Bytes, 地址偏移 12
10.    uint16_t CellVoltage_2;         //电芯 2 电压, 单位 mV, 2 Bytes, 地址偏移 14
11.    uint16_t CellVoltage_3;         //电芯 3 电压, 单位 mV, 2 Bytes, 地址偏移 16
12.    uint16_t CellVoltage_4;         //电芯 4 电压, 单位 mV, 2 Bytes, 地址偏移 18
13.    uint16_t CellVoltage_5;         //电芯 5 电压, 单位 mV, 2 Bytes, 地址偏移 20
14.    uint16_t CellVoltage_6;         //电芯 6 电压, 单位 mV, 2 Bytes, 地址偏移 22
15.    uint16_t CellVoltage_7;         //电芯 7 电压, 单位 mV, 2 Bytes, 地址偏移 24
16.    uint16_t CellVoltage_8;         //电芯 8 电压, 单位 mV, 2 Bytes, 地址偏移 26
17.    uint16_t CellVoltage_9;         //电芯 9 电压, 单位 mV, 2 Bytes, 地址偏移 28
18.    uint16_t CellVoltage_10;        //电芯 10 电压, 单位 mV, 2 Bytes, 地址偏移 30
19.    uint16_t CellVoltage_11;        //电芯 11 电压, 单位 mV, 2 Bytes, 地址偏移 32
20.    uint16_t CellVoltage_12;        //电芯 12 电压, 单位 mV, 2 Bytes, 地址偏移 34
21.    uint16_t CellVoltage_13;        //电芯 13 电压, 单位 mV, 2 Bytes, 地址偏移 36
22.    uint16_t CellVoltage_14;        //电芯 14 电压, 单位 mV, 2 Bytes, 地址偏移 38
23.    uint16_t CellVoltage_15;        //电芯 15 电压, 单位 mV, 2 Bytes, 地址偏移 40
24.    uint16_t CellVoltage_16;        //电芯 16 电压, 单位 mV, 2 Bytes, 地址偏移 42
25.    int32_t ChargeCurrent;           //充电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 44
26.    int32_t DisChargeCurrent;        //放电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 48
27.    uint32_t FullChargeCapacity;     //满充容量, 单位 mAh, 4 Bytes, 地址偏移 52
28.    uint32_t RemainCapacity;         //剩余容量, 单位 mAh, 4 Bytes, 地址偏移 56
29.    uint16_t CycleCount;             //循环次数, 2 Bytes, 地址偏移 60
30.    uint8_t Temperature_1;          //温度 1, 单位°C, 偏移 40, 1 Byte, 地址偏移 62
31.    uint8_t Temperature_2;          //温度 2, 单位°C, 偏移 40, 1 Byte, 地址偏移 63
32.    uint8_t Temperature_3;          //温度 3, 单位°C, 偏移 40, 1 Byte, 地址偏移 64
```

```
33.    uint8_t Temperature_4;           //温度 4, 单位°C, 偏移 40, 1 Byte, 地址偏移 65
34.    uint8_t SOC;                     //剩余电量, 1 Byte, 地址偏移 66
35.    uint8_t SOH;                     //电荷寿命, 1 Byte, 地址偏移 67
36.    uint8_t AFE_Status;              //AFE 状态, 1 Byte, 地址偏移 68
37.    uint8_t Working_State;           //工作状态, 1 Byte, 地址偏移 69
38.    uint16_t MaxChargeTime;           //最大充电持续时间, 单位 min, 地址偏移 70
39.    uint16_t MaxBetweenChargeTime;    //最大充电间隔时间, 单位 min, 地址偏移 72
40.    uint16_t MaxBetweenDisChargeTime; //最大放电持续时间, 单位 min, 地址偏移 74
41.    uint16_t LastBetweenDisChargeTime; //最大放电间隔时间, 单位 min, 地址偏移 76
42.    uint16_t MaxUVPTime;             //最大欠压保护间隔时间, 单位 min, 地址偏移 78
43.    RTC_Struct_t RTC;                //实时时钟, 8 Bytes, 地址偏移 80
44.    RTC_Struct_t LastChargeTime;      //最近充电时钟, 8 Bytes, 地址偏移 88
45.    RTC_Struct_t LastDisChargeTime;    //最近放电时钟, 8 Bytes, 地址偏移 96
46.    RTC_Struct_t LastUVP_RTC;         //最近欠压保护时钟, 8 Bytes, 地址偏移 104
47.    RTC_Struct_t LastUVP_Active_RTC;  //最近欠压保护激活时钟, 8 Bytes, 地址偏移 112
48.    RTC_Struct_t LastFCC_Update_RTC   //最近满容量更新时钟, 8 Bytes, 地址偏移 120
49. }BMS_ErrorLogSaveInfo_Struct_t;
50.
51. //RTC 数据结构
52. typedef struct
53. {
54.    uint8_t RS1;                      //预留, 0x00
55.    uint8_t Year;                     //年
56.    uint8_t Mouth;                   //月
57.    uint8_t Date;                    //日
58.    uint8_t RS2;                      //预留, 0x00
59.    uint8_t Hour;                    //时
60.    uint8_t Minute;                  //分
61.    uint8_t Second;                  //秒
62. }
```

MOTINOVA 轮毂控制器系统通信协议

1 系统组成

MC: 电机控制器 Motor Controller

BMS: 电池管理系统 Battery Management System

OBC: 车载计算机 On Board Computer

CDL: 通讯适配器 CAN Dongle

APP: 用户程序 Application

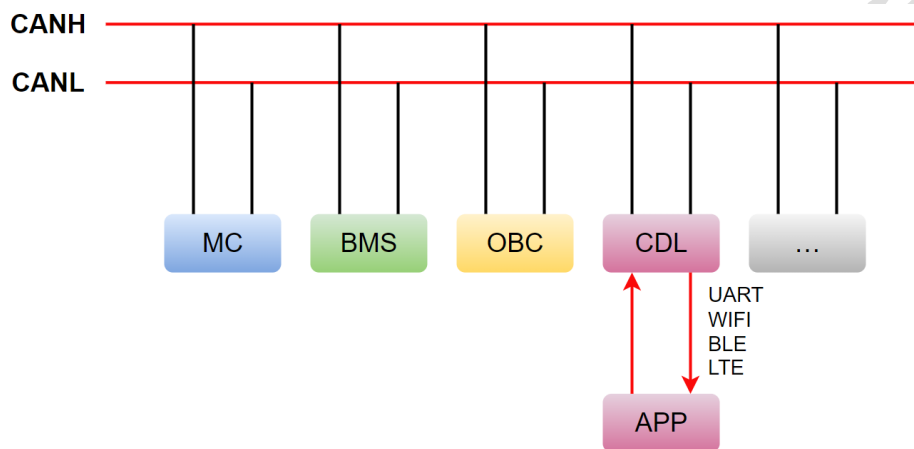


图1 系统通信接口示意图

2 硬件接口

2.1 通信接口参考电路

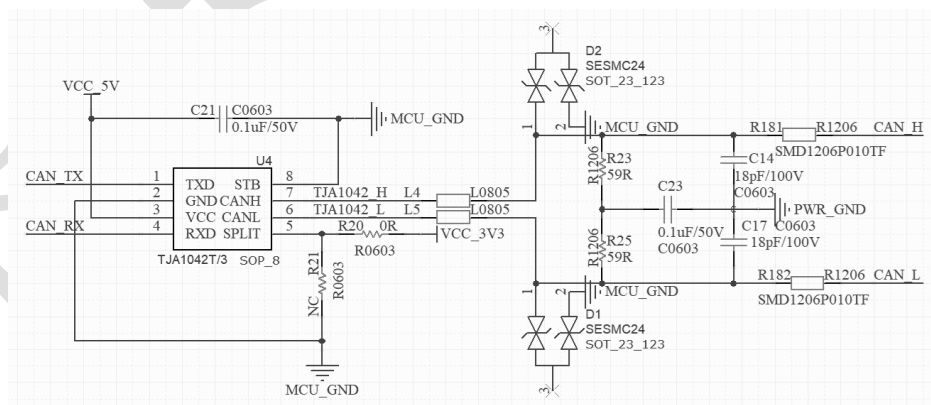


图2 通信接口参考电路

2.2 注意事项

图 2 展示的为各组件 CAN 通信接口参考电路，设计时注意以下几点：

4) ESD 保护电路设计需考虑电源线与通信线发生短路时，不会导致 ESD 保护器件

被击穿, 电源线最高电压按照 48V 系统最高工作电压 54.6V 测试;

5) 终端匹配电阻设计在 BMS 和 OBC 内部, 其它部件预留;

6) BMS 需考虑通信接口电气隔离设计。

3 通信协议规则

3.1 软件配置

CAN 控制器推荐配置如下:

时钟频率: 1500 KHz

SJW: 1

BS1: 4

BS2: 1

波特率: 250 Kbps

3.2 数据帧封装格式

所有传输的数据按照标准帧进行封装, 将传输数据按顺序填入数据帧中。以下对数据帧格式、ID 分配、分包方式进行介绍。

3.2.1 数据帧格式

协议描述了每帧数据内容, 包括帧头、帧模式、命令段长度、命令字、数据段、校验位、帧尾。每帧格式如下:

表1 数据帧格式

帧头	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中:

15) 帧头固定为 0x55 0xAA, 帧尾固定为 0xF0;

16) 帧模式包含读 0x11, 写 0x16, 和上报 0x0C, 任何设备收到写指令时, 需根据数据来源发送通用反馈指令;

17) LENGTH 命令段总长度, 占用 1 字节, 有效值为 0x02~0xFF;

18) COMMAND 为命令字, 占用 2 个字节, 第 1 字节为命令字序号, 第 2 字节为数据段长度;

19) DATA 为数据段, 长度为 LENGTH - 2;

20) CRC 为校验位, 占用 4 字节, 由帧头开始, CAN_ID 插入到帧头和帧模式之间, 计算到数据段最后一个字节, 计算方法见附录 1, 计算结果高字节在前, 如:

CAN_ID 为 0x0712, 数据帧为 55 AA 11 03 22 01 00 CRC1 CRC2 CRC3 CRC4 F0,
CRC 计算函数输入数据为 55 AA 07 12 11 03 22 01 00, 计算结果依次由高到低写入 CRC1、CRC2、CRC3、CRC4;

21) 数据段发送时, 采用小端模式。

3.2.2 ID 分配

表2 ID 分配

MC	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x710		0x712	0x713	0x715
BMS	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x720	0x721		0x723	0x725
OBC	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x730	0x731	0x732		0x735
CDL	Target	广播	MC	BMS	OBC	CDL
	CAN ID	0x750	0x751	0x752	0x753	

3.2.3 分包方式

对于长度超过 8bytes 的数据帧, 按照 8+N 的方式分包, 每个数据包填入相同的 ID 号, 如下表所示:

表3 封装方式

包序号	1				N	
内容	ID	Byte1~Byte8	ID	Byte1~Byte8	ID	Byte1~ByteN

3.3 CDL 数据格式转换规则

3.3.1 接收处理

CDL 接收到 CAN 总线数据后, 按照下表格式插入 ID, 完成转发:

表4 CDL 数据转换格式

帧头	CAN ID	帧模式	命令段长度	命令字	数据段	校验位	帧尾
55 AA	ID	读/写/上报	LENGTH	COMMAND	DATA	CRC	F0

其中 ID 占用 2 字节。

3.3.2 发送处理

CDL 接收到其它设备按照上表格式发送的数据后, 取数据包中的 ID 作为当前发送数据帧的 CAN ID, 并以表 1 格式完成分包转发。

3.3.3 CDL 专用指令

用于 CDL 与 APP 之间的数据交互, 此部分指令不需要通过 CDL 发送到 CAN 总线。

3.3.3.1 CDL 在线检测

在其它 UART、WIFI、BLE 等接口方式的设备 APP 通过 CDL 访问 CAN 总线上 MC、BMS、OBC、OBC 等设备时, 需要先确认 CDL 连接是否正常。

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 收到 CDL 返回表示 CDL 在线, 超时 2s 未收到 CDL 返回表示 CDL 离线:

[55 AA 07 FF 11 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 返回:

[55 AA 07 FF 0C 02 11 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.2 设备开关机

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 并显示开机结果:

[55 AA 07 FF 16 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

2) CDL 接收到指令后使能开机信号, 同时返回:

[55 AA 07 FF 0C 03 22 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下: 0xF0:关机, 0xF1:开机

3.3.3.3 上报系统供电方式

CDL 定时 2s 发送一次系统供电方式, APP 显示供电方式, 数据如下:

[55 AA 07 FF 0C 03 33 01 DATA CRC1 CRC2 CRC3 CRC4 F0]

DATA 含义如下:

0x00: 适配器未接入、外部设备无电池

0x01: 适配器未接入、外部设备含电池

0x11: 适配器接入、外部设备含电池

0x10: 适配器接入、外部设备无电池

3.3.3.4 CDL 复位

1) APP 定时 200ms 发送, 收到 CDL 返回或超时 2s 停止发送, 并显示复位结果:

[55 AA 07 FF 11 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

2) CDL 收到指令后返回指令, 然后复位:

[55 AA 07 FF 0C 02 44 00 CRC1 CRC2 CRC3 CRC4 F0]

3.3.3.5 设备校验

通过其它 UART、WIFI、BLE 等接口方式的设备 APP 连接 CDL, 需要对设备与 CDL 进

行双向校验。

1) APP 发送随机码和校验密钥给到 CDL:

[55 AA 07 FF 16 16 55 14 随机码 密钥 CRC1 CRC2 CRC3 CRC4 F0]

其中随机码长度为 12Bytes, 密钥长度为 8Bytes。

2) CDL 根据随机码和密钥进行计算后, 将校验码发送给 APP, 由 APP 判断校验是否成功:

[55 AA 07 FF 0C 1D 55 1B 校验结果 版本号 CRC1 CRC2 CRC3 CRC4 F0]

其中校验结果长度为 12Bytes, 版本号为 15Bytes, 格式为 Vx.x.x_YYYYMMDD。

4 通信内容

4.1 MC 命令字定义

表5 MC 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x710	0x0C	0x1020	MC 运行信息 (收到 OBC 控制指令时返回)	车速:2bytes 输出转速:2bytes 电功率:2bytes 母线电压:2bytes 母线电流:2bytes 踢频:1byte 踩踏力矩:1byte 踩踏方向:1byte 助力档位:1byte 大灯状态:1byte 剩余电量:1byte 续航里程:2bytes 预留:2bytes 平均功耗:1byte PCB 温度:1bytes 绕组温度:1bytes MCU 温度:1bytes	0.1km/h 1rpm 1W 1mV 1mA 1rpm 1N.m 0-正, 1-反, 2-停止 0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开 1%, 无效为 0xFF 1km, 无效为 0xFFFF 0x00 0.01Ah/km, 无效为 0xFF +40℃ +40℃ +40℃

				开机后骑行里程:2bytes 开机后骑行时间:2byte 预留:4byte	0.1km 1s 填充 0x00
0x710	0x0C	0x1104	MC 故障码 (存在故障时 200ms 自动发送, 故障消失后停止发送)	0:无故障 低 16 位: 0x0001:过流保护 0x0002:低压保护 0x0004:过压保护 0x0008:堵转保护 0x0010:过热保护 0x0020:SPS 故障 0x0040:TQS 故障 0x0080:霍尔故障 0x0100:马达缺相 0x0200:NTC 故障 0x0400:BMS 校验失败 0x1000:OBC 校验失败 0x2000:MCU 故障 0x4000:踏频故障 0x8000:指拨故障 高 16 位: 0x0001:MOS 短路 0x0002:电压异常 0x0004:电路故障 0x0040:马达失速故障 0x0080:预留故障	按位或输出, 0-正常, 1-故障
0x710	0x0C	0x1240	电机版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20 FW 命名格式为 Vxrxxr_x_YYYYMMDD
0x710	0x0C	0x1305	关机就绪 (返回指令)	ASCII 字符	READY
0x710	0x0C	0x1401	在线检测结果 (返回指令)	0x00:系统正常 0x01:BMS 离线 0x02:OBC 离线 0x04:预留	按位或输出, 0-在线, 1-离线

				0x08: 预留 0x10: 预留 0x20: 预留 0x40: 预留 0x80: 预留	
0x710	0x0C	0x1510	骑行历史信息 (返回指令)	ODO 里程: 4bytes ODO 时间: 4bytes TRIP 里程: 4bytes TRIP 时间: 4bytes	0.1km 1min 0.1km 1min
0x710	0x0C	0x1720	特殊性说明	ASCII 字符	结束符为',', 无效填充 0x20
0x710	0x0C	0x1808	关机指令 (控制器关机前发送) [适用 V2 电机]	ASCII 字符	SHUTDOWN
发送给 BMS					
0x712	0x11	0x3009	BMS 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x712	0x11	0x3100	查询 BMS 物理 ID (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3200	查询 BMS 校验码 (主动发送, 收到返回或超时停止)		
0x712	0x11	0x3300	查询 BMS 设计信息 (主动发送, 收到返回或超时停止)		
发送给 OBC					
0x713	0x11	0x5009	OBC 在线检测 (主动发送, 收到返回或超时停止)	ASCII 字符	HANDSHAKE
0x713	0x11	0x5100	查询 OBC 物理 ID (主动发送, 收到返回或超时停止)		

0x713	0x11	0x5200	查询 OBC 校验码 (主动发送, 收到返回或超时停止)		
0x713	0x0C	0x5303	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x713	0x0C	0x5408	返回 OBC 用户参数	轮胎默认周长: 1byte 启动模式: 1byte 限速: 1byte 周长微调: 1byte 助力方案: 1byte 自动关机时间: 1byte 预留: 2bytes	1cm 1-柔和, 2-正常, 3-强劲 1km/h ±10cm, 有符型 1-2 1min 填充 0x00
发送给 CDL					
0x715	0x0C	0xA408	校验码计算密钥 (返回指令)	密钥: 8bytes	ASCII
0x715	0x0C	0xA520	生产信息 (返回指令)	生产商: 8bytes 生产地: 8bytes 生产日期: 8bytes 产品标识: 8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD GF_250_1 GF_350_1 GR_250_1 GR_350_1 GR_500_1 GR_750_1 GR_10X_1
0x715	0x0C	0xA610	自定义可存储字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA710	自定义可存储字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA810	自定义可存储字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x715	0x0C	0xA903	通用反馈指令 (返回指令)	ASCII 字符	ACK

0x715	0x0C	0xAB88	返回存储器指定起始和结束地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 填充 0xFF
0x715	0x0C	0xB226	马达信息 (返回指令)	极对数:2bytes 电阻:2bytes d 轴电感:2bytes q 轴电感:2bytes 永磁体磁链:2bytes Id 最大值:2bytes Id 最小值:2bytes 额定转速:2bytes 额定功率:2bytes 额定电流:2bytes 额定电压:2bytes 惯量:2bytes 最大转矩:2bytes 预留:12byte	1-65535 0.1 mOhm 0.01 uH 0.01 uH 0.001 mWb 0.01 A 0.01 A 1 rpm 1 W 0.01 A 0.1 V 10^{-7} kg*m ² 0.1 Nm 填充 0x00
0x715	0x0C	0xB31A	整车信息	轮胎周长:2bytes 电控传动比:2bytes 助力最大限速:2bytes 推行模式限速:2bytes 前牙盘 T 数:2bytes 后牙盘 T 数:2bytes 助力方案 1 编号:2bytes 助力方案 2 编号:2bytes 前后灯电压:2bytes 周长微调值:2bytes 启动模式: 2bytes 自动关机时间:2bytes 预留:2bytes	1 cm 0.1 1 km/h 1 km/h 1 1 每 2 bit 代表 1 个档位的助力曲线, 由低到高依次表示 1-5 档 2 bit 代表 1 个档位的助力曲线, 由低到高依次表示 1-5 档 高位: 前灯, 6:6V, 12:12V 低位: 后灯, 6:6V, 12:12V ± 10 cm, 有符型 1-柔和, 2-正常, 3-强劲 1min, 设置 0 为不自动关机 0x00

0x715	0x0C	0xB420	控制器参数	位置传感器零	Q14
				点:2bytes	
				位置传感器当前零	Q14
				点:2bytes	
				峰值电流:2bytes	0.01 A
				电流保护阈	0.01 A
				值:2bytes	
				最高档电压保护阈	0.1 V
				值:2bytes	
				最高档欠压保护阈	0.1 V
				值:2bytes	
				超速保护阈	1 rpm
				值:2bytes	
				温度保护阈	1 Ce
				值:2bytes	
				温度保护恢复阈	1 Ce
				值:2bytes	
				温度降额启动阈	1 Ce
				值:2bytes	
				预留:12bytes	填充 0x00
0x715	0x0C	0xB528	传感器参数	力矩传感器出厂零	Q12
				点值:2bytes	
				力矩传感器历史零	Q12
				点值 1:2bytes	
				力矩传感器历史零	Q12
				点值 2:2bytes	
				力矩传感器历史零	Q12
				点值 3:2bytes	
				力矩传感器最新零	Q12
				点值:2bytes	
				力矩传感器最大转	0.1 Nm
				矩:2bytes	
				第一阶段转矩点加	0.1 Nm
				载值:2bytes	
				第一阶段转矩点采	Q12
				集值:2bytes	
				第二阶段转矩点加	0.1 Nm
				载值:2bytes	
				第二阶段转矩点采	Q12
				集值:2bytes	
				第三阶段转矩点加	0.1 Nm
				载值:2bytes	
				第三阶段转矩点采	Q12
				集值:2bytes	

保密文件， 禁止外传

MOTINOVA		文件名称：MOTINOVA E-Bike 电气系统通信协议				密级：秘密	
		文件编号：[填入文件编号]			版本：V4.2	共 133 页 第 105 页	
					待速最大电 流:2bytes	0.01 A	
					车速限幅启动阈 值:2bytes	0.1km	
					车速限幅停止阈 值:2bytes	0.1 km	
					踏频占比:2bytes		
					预留:16bytes	填充 0x00	
0x715	0x0C	0xB74C	历史信息	开机次数:2bytes	1		
				使用时间 H:2bytes	1 min		
				使用时间 L:2bytes	1 min		
				NTC 温度传感器最 高温:2bytes	1 Ce		
				NTC 温度传感器最 低温:2bytes	1 Ce		
				硬件过流保护次 数:2bytes	1		
				软件过流保护次 数:2bytes	1		
				过温保护次 数:2bytes	1		
				堵转次数:2bytes	1		
				缺相次数:2bytes	1		
				过压次数:2bytes	1		
				欠压次数:2bytes	1		
				通讯超时次 数:2bytes	1		
				平均功耗 1:2bytes	0.01 Ah/km		
				平均功耗 2:2bytes	0.01 Ah/km		
				平均功耗 3:2bytes	0.01 Ah/km		
				平均功耗 4:2bytes	0.01 Ah/km		
				平均功耗 5:2bytes	0.01 Ah/km		
				ODO 里程 H:2bytes	0.1 km		
				ODO 里程 L:2bytes	0.1 km		
				ODO 时间 H:2bytes	1 min		
				ODO 时间 L:2bytes	1 min		
				TRIP 里程 H:2bytes	0.1 km		
				TRIP 里程 L:2bytes	0.1 km		
				TRIP 时间 H:2bytes	1 min		
				TRIP 时间 L:2bytes	1 min		
				力矩传感器故障次 数:2bytes	1		
				踏频传感器故障次 数:2bytes	1		

保密文件， 禁止外传

				预留:8bytes	填充 0x00
0x715	0x0C	0xB904	存储标志	EEPROM 存储 FLAG:2bytes SIP 偏置校准 FLAG:2bytes	0-未存储,1-已存储 0-未存储,1-已存储

4.2 BMS 命令字定义

表6 BMS 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x720	0x0C	0x1010	电池运行信息 (返回指令)	电压:2bytes 平均电流:2bytes 剩余容量:2bytes 满充容量:2bytes 电芯温度:1byte 剩余电量:1byte 运行状态:1byte (按位或输出) SOH:1byte 预留:4bytes	1mV 1mA, 有符型, 放电 为负, 充电为正 1mAh 1mAh +40℃ 0~100% 0x00:休眠 0x01:充电器接入 0x02:预留 0x04:预留 0x08:预留 0x10:预留 0x20:预留 0x40:预留 0x80:预留 0~100% 填充 0x00
0x720	0x0C	0x1120	电芯电压 (返回指令)	Cell_1:2bytes Cell_16:2bytes 不足部分填充 0x00	1mV 1mV
0x720	0x0C	0x1204	BMS 故障码 (存在故障时 200ms 自动发 送, 故障消失后 停止发送)	高 16 位: 0x0001:充电过压 警告 0x0002:放电低压 警告 0x0004:充电过流 警告 0x0008:放电过流 警告 0x0010:充电高温 警告	按位或输出, 0-正 常, 1-故障

				0x0020: 充电低温警告 0x0040: 放电高温警告 0x0080: 放电低温警告 0x0100: MOS 高温警告 低 16 位: 0x0001: 二级放电过流保护 0x0002: 充电过流保护 0x0004: 短路保护 0x0008: 过放保护 0x0010: 过充保护 0x0020: 放电低温保护 0x0040: 放电高温保护 0x0080: 充电低温保护 0x0100: 充电高温保护 0x0200: 放电 MOS 故障 0x0400: 充电 MOS 故障 0x0800: 温度传感器故障 0x1000: 预留 0x2000: 一级过流保护 0x4000: AFE 故障 0x8000: MCU 故障	
0x720	0x0C	0x1308	关机指令 (主动发送, 收到返回或超时停止)	ASCII 字符	常按开关键 3s 或监测到母线电流小于 50mA 且 CAN 总线空闲持续 30min 后, 执行发送 SHUTDOWN, 延时 1s 后, 关闭放电开关
0x720	0x0C	0x1410	电池设计信息 (返回指令)	设计容量: 2bytes 设计电压: 1byte 电芯型号: 8bytes	1mAh 1V ASCII, 0x2E 结束,

				预留:5bytes	无效填充 0x20 填充 0x00
0x720	0x0C	0x1540	电池版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、 SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20; FW 命名格式为 Vxrxrx_YYYYMMDD.
0x720	0x0C	0x160C	电池物理 ID (返回指令)	ID:12bytes	不足位填充 1
0x720	0x0C	0x170C	电池校验码 (返回指令)	校验码:12bytes	
发送给 MC					
0x721	0x0C	0x3005	在线检测反馈 (返回指令)	ASCII 字符	READY
发送给 CDL					
0x725	0x0C	0x5028	电池 BMS 历史信息 (返回指令)	电芯最高温:1byte 电芯最低温:1byte 最大放电电 流:2bytes 最大充电电 流:2bytes 循环次数:2bytes 最近充电间隔时 间:2bytes 最大充电间隔时 间:2bytes 充电过流保护次 数:2bytes 放电过流保护次 数:2bytes 过充保护次 数:2bytes 过放保护次 数:2bytes 短路保护次 数:2bytes 充电低温保护次 数:2bytes 充电高温保护次 数:2bytes 放电低温保护次	+40℃ +40℃ 无符型, 单位 1mA 无符型, 单位 1mA 次 小时 小时 次 次 次 次 次 次 次 次 次 次 次 次 次 次

				数:2bytes 放电高温保护次数:2bytes 运行时间:4bytes SOH:1byte 预留:5bytes	次 1min 0~100% 填充 0x00
0x725	0x0C	0x5120	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x725	0x0C	0x5210	自定义可存储字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5310	自定义可存储字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5410	自定义可存储字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x725	0x0C	0x5503	通用反馈指令 (返回指令)	ASCII 字符	ACK
0x725	0x0C	0x5688	存储器指定起始和结束地址的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF

4.3 OBC 命令字定义

表7 OBC 命令字定义

ID	模式	命令字	功能	数据段	备注
广播指令					
0x730	0x0C	0x1140	版本信息 (返回指令)	ASCII 字符	排列顺序为: MODE、SN、HW、FW; 每条信息长度为 16 bytes, 结束符为 '.', 无效填充 0x20 FW 命名格式为 Vxrxxr_YYYYMMDD
0x730	0x0C	0x120C	物理 ID (返回指令)	ID:12bytes	不足位填充 1
0x730	0x0C	0x130C	校验码 (返回指令)	校验码:12bytes	
0x730	0x0C	0x1405	关机就绪	ASCII 字符	READY

			(返回指令)		
0x730	0x0C	0x1504	故障代码 (存在故障时 200ms 自动发 送, 故障消失后 停止发送)	高 16 位: 0x0000 低 16 位 0x0000: 无故障 0x0001: +键失效 0x0002: -键失效 0x0004: i 键失效 0x0008: 灯键失效 0x0010: Walk 键失 效 0x0020: 电源键失 效 0x0400: MC 通讯异 常	按位或输出, 0-正 常, 1-故障
发送给 MC					
0x731	0x0C	0x3002	控制电机指令 (定时上传)	助力档位: 1byte 大灯状态: 1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x05: 力矩预留 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x731	0x0C	0x3105	在线检测反馈 (返回指令)	ASCII 字符	READY
0x731	0x11	0x3300	查询用户参数 (主动发送, 收 到返回或超时 停止)		
0x731	0x16	0x3408	设置用户参数 (主动发送, 收 到返回或超时 停止)	周长微调值: 1byte 启动模式: 1byte 助力方案: 1byte 自动关机时 间: 1byte 预留: 6bytes	有符型, 单位 1cm, ±10.0cm 1-柔和, 2-正常, 3- 强劲 1-2 1min, 设置为 0 时控 制器不自动关机 填充 0x00
0x731	0x11	0x3500	查询骑行历史 信息 (主动发送, 收 到返回或超时 停止)		

0x731	0x16	0x3605	清除电机 TRIP 信息（主动发送，收到返回或超时停止）	ASCII 字符	CLEAR
0x731	0x11	0x3900	OBC 读取电机版本信息		
0x731	0x0C	0x3A02	OBC 发送按键状态（主动发送，按键按下触发发送一次）	Byte1: 预留 Byte2: Bit7:0 Bit6:0 Bit5: 电源键 Bit4: +键 Bit3: -键 Bit2: Walk 键 Bit1: Light 键 Bit0: M/Set 键	0x00 按键状态按位或填充，1-按下，0-弹起
发送给 BMS					
0x732	0x11	0x5000	查询 BMS 运行信息（主动发送，收到返回或超时停止）		
0x732	0x11	0x5100	查询 BMS 版本信息（主动发送，收到返回或超时停止）		
0x732	0x11	0x5200	查询 BMS 设计信息（主动发送，收到返回或超时停止）		
0x732	0x11	0x5300	查询 BMS 电芯电压（主动发送，收到返回或超时停止）		
发送给 CDL					
0x735	0x0C	0x9003	通用反馈指令（返回指令）	ASCII 字符	ACK
0x735	0x0C	0x9101	发送按键状态（定时上传）	+键: Bit0 -键: Bit1 i 键: Bit2 灯键: Bit3 Walk 键: Bit4	0-松开 1-按下 多余位填充 0

				电源键:Bit5	
0x735	0x0C	0x9310	运行历史信息 (返回指令)	运行时间:4bytes 开机次数:4bytes 预留:8bytes	1min 次 填充 0x00
0x735	0x0C	0x9420	生产信息 (返回指令)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x735	0x0C	0x9510	自定义可存储 字符串 1 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9610	自定义可存储 字符串 2 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9710	自定义可存储 字符串 3 (返回指令)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x735	0x0C	0x9888	存储器指定起 始和结束地址 的数据	起始地址:4bytes 结束地址:4bytes 数据:128bytes	结束地址-起始地址 <128 时, 无效部分 填充 0xFF

4.4 CDL 命令字定义

表8 CDL 命令字定义

ID	模式	命令字	功能	数据段	备注
发送给 MC					
0x751	0x11	0x1000	查询电机校验 码密钥		
0x751	0x16	0x1108	写入电机校验 码密钥		
0x751	0x11	0x1200	查询电机版本 信息		
0x751	0x11	0x1300	查询自定义可 存储字符串 1		
0x751	0x16	0x1410	写入自定义可 存储字符串 1	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1500	查询自定义可 存储字符串 2		
0x751	0x16	0x1610	写入自定义可 存储字符串 2	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x11	0x1700	查询自定义可 存储字符串 3		

0x751	0x16	0x1810	写入自定义可 存储字符串 3	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x16	0x1901	写入电机工作 模式	工作模式:1byte	0-运行, 1-配置
0x751	0x11	0x1E00	查询电机历史 信息		
0x751	0x11	0x1F00	查询电机生产 信息		
0x751	0x11	0x2100	查询设备在线 检测结果		
0x751	0x16	0x2210	写入电机 Mode (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x16	0x2310	写入电机 SN (可选, 仅供 生产商写入)	ASCII 字符	结束符为 0x2E, 无 效填充 0x20
0x751	0x16	0x2420	写入生产信息 (可选, 仅供 生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 产品标识:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD GF_250_1 GF_350_1 GR_250_1 GR_350_1 GR_500_1 GR_750_1 GR_10X_1.
0x751	0x16	0x2505	复位指令	ASCII 字符串	RESET
0x751	0x16	0x2605	数据清除	ASCII 字符串	CLEAR
0x751	0x16	0x2708	参数还原	ASCII 字符串	RECOVERY
0x751	0x16	0x2802	控制指令	助力档位:1byte 大灯状态:1byte	0x00: OFF 0x01: 力矩 ECO 0x02: 力矩 NORM 0x03: 力矩 SPORT 0x04: 力矩 TURBO 0x22: 推行模式 0x33: 智能模式 0xF0-关, 0xF1-开
0x751	0x16	0x2C01	设置电机转速 百分比	百分比:1byte	5~100
0x751	0x11	0x2D08	读取存储器指 定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤ 128Bytes

0x751	0x11	0x2E00	查询骑行历史信息		
0x751	0x16	0x3909	历史信息清除	ASCII	LOG CLEAR
0x751	0x11	0x3A00	读取马达信息		
0x751	0x16	0x3B28	写入马达信息	存储标志:2bytes 极对数:2bytes 电阻:2bytes d 轴电感:2bytes q 轴电感:2bytes 永磁体磁链:2bytes Id 最大值:2bytes Id 最小值:2bytes 额定转速:2bytes 额定功率:2bytes 额定电流:2bytes 额定电压:2bytes 惯量:2bytes 最大转矩:2bytes 预留 0:12byte	0-不存储, 1-存储 1-65535 0.1 mOhm 0.01 uH 0.01 uH 0.001 mWb 0.01 A 0.01 A 1 rpm 1 W 0.01 A 0.1 V 10^{-7} kg*m ² 0.1 Nm 填充 0x00
0x751	0x11	0x3C00	读取整车信息		
0x751	0x16	0x3D1C	写入整车信息	存储标志:2bytes 轮径:2bytes 电控传动比:2bytes 助力最大限 速:2bytes 推行模式限 速:2bytes 前牙盘 T 数:2bytes 后牙盘 T 数:2bytes 助力方案 1 编 号:2bytes 助力方案 2 编 号:2bytes 前后灯电压:2bytes 周长微调值:2bytes 启动模式: 2bytes	0-不存储, 1-存储 1 cm 0.1 1 km/h 1 km/h 1 1 每 2 bit 代表 1 个档 位的助力曲线, 由低 到高依次表示 1-5 档 2 bit 代表 1 个档位 的助力曲线, 由低到 高依次表示 1-5 档 高位: 前灯, 6:6V, 12:12V 低位: 后灯, 6:6V, 12:12V ±10cm, 有符型 1-柔和, 2-正常, 3-

				自动关机时间:2bytes 预留:2bytes	强劲 1min, 设置 0 为不自 动关机 0x00
0x751	0x11	0x3E00	查询控制器参数		
0x751	0x16	0x3F22	写入控制器参数	存储标志:2bytes 位置传感器零点:2bytes 位置传感器当前零点:2bytes 峰值电流:2bytes 电流保护阈值:2bytes 最高档电压保护阈值:2bytes 最高档欠压保护阈值:2bytes 超速保护阈值:2bytes 温度保护阈值:2bytes 温度保护恢复阈值:2bytes 温度降额启动阈值:2bytes 预留:12bytes	0-不存储, 1-存储 Q14 Q14 0.01 A 0.01 A 0.1 V 0.1 V 1 rpm 1 Ce 1 Ce 1 Ce 填充 0x00
0x751	0x11	0x4000	查询传感器参数		
0x751	0x16	0x420E	写入其它传感器参数	存储标志:2bytes 踏频传感器一圈脉冲数:2bytes 车速传感器一圈脉冲数:2bytes 预留:8bytes	0-不存储, 1-存储 1 1 填充 0x00
0x751	0x11	0x4304	查询助力参数	助力转矩曲线编号:2bytes 助力踏频曲线编号:2bytes	1 1
0x751	0x16	0x4450	写入助力参数	存储标志:2bytes 填充:2bytes 零速启动增益:2bytes	0-不存储, 1-存储 0x00 0.1

				巡航启动增益:2bytes	0.1
				助力转矩曲线编号:2bytes	助力转矩曲线编号1-15
				助力踏频曲线编号:2bytes	助力踏频曲线编号1-5
				转矩曲线.a:4bytes	
				转矩曲线.b:4bytes	
				转矩曲线.c:4bytes	
				转矩曲线.d:4bytes	
				踏频曲线.a:4bytes	
				踏频曲线.b:4bytes	
				踏频曲线.c:4bytes	
				踏频曲线.d:4bytes	
				助力启动阈值:2bytes	
				助力停止阈值:2bytes	
				启动时电流增长阶梯:2bytes	
				启动对应踏频脉冲数:2bytes	
				转矩滤波对应踏频脉冲数:2bytes	0.1 Nm
				待速转速:2bytes	0.1 Nm
				待速最大电流:2bytes	Q14
				车速限幅启动阈值:2bytes	1
				车速限幅停止阈值:2bytes	1
				踏频占比:2bytes	
				预留:16bytes	1 rpm 0.01 A 0.1km 0.1 km 填充 0x00
0x751	0x11	0x4500	查询调试参数		
0x751	0x16	0x463C	写入调试参数	存储标志:2bytes 整体运行模	1- 不存储, 1-存储

MOTINOVA			文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
			文件编号: [填入文件编号]	版本: V4. 2	共 133 页 第 118 页
				式:2bytes 位置获取模 式:2bytes 采样模式:2bytes 旋转方向:2bytes 定位电流:2bytes 拖拽电压:2bytes 拖拽电流:2bytes 拖拽频率:2bytes 加速斜率:2bytes 减速斜率:2bytes 转速环控制器带 宽:2bytes 转速环控制器 m:2bytes 电流环控制器带 宽:2bytes 电流环控制器 m:2bytes 磁链观测器带 宽:2bytes 磁链观测器 m:2bytes 锁相环带宽:2bytes 锁相环 m:2bytes 惯量:2bytes PWM 最大占空 比:2bytes PWM 七段式切五段式 阈值:2bytes 功率限幅值:2bytes 功率 Error:2bytes 功率限幅 Kp:2bytes 功率限幅 Ki:2bytes 预留:8bytes	
					填充 0x00
0x751	0x11	0x4800	读取存储标志		
0x751	0x16	0x4906	写入存储标志	存储标志:2bytes EEPROM 存储 FLAG:2bytes SIP 偏置校准存储	0-不存储, 1-存储 0-未存储, 1-已存储 0-未存储, 1-已存储

				FLAG:2bytes	
发送给 BMS					
0x752	0x11	0x3000	查询电池物理 ID		
0x752	0x11	0x3100	查询电池校验码		
0x752	0x16	0x320C	写入电池校验码	校验码:12bytes	
0x752	0x11	0x3300	查询电池版本信息		
0x752	0x11	0x3400	查询电池运行信息		
0x752	0x11	0x3500	查询电芯电压		
0x752	0x11	0x3600	查询电池设计信息		
0x752	0x11	0x3700	查询电池生产信息		
0x752	0x11	0x3800	查询电池历史信息		
0x752	0x11	0x3900	查询自定义可存储字符串 1		
0x752	0x16	0x3A10	写入自定义可存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x11	0x3B00	查询自定义可存储字符串 2		
0x752	0x16	0x3C10	写入自定义可存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x11	0x3D00	查询自定义可存储字符串 3		
0x752	0x16	0x3E10	写入自定义可存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x752	0x16	0x3F20	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x752	0x16	0x4010	写入 BMS Mode (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x752	0x16	0x4110	写入 BMS SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20

0x752	0x16	0x4205	复位指令	ASCII 字符串	RESET
0x752	0x11	0x4308	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤128Bytes
发送给 OBC					
0x753	0x11	0x5000	查询物理 ID		
0x753	0x11	0x5100	查询校验码		
0x753	0x16	0x520C	写入校验码	校验码:12bytes	
0x753	0x16	0x5300	查询生产信息		
0x753	0x11	0x5400	查询版本信息		
0x753	0x16	0x5504	设置进入/退出按键测试模式	ASCII 字符	STAR:进入 STOP:退出
0x753	0x11	0x5600	查询历史信息		
0x753	0x11	0x5900	查询自定义可存储字符串 1		
0x753	0x16	0x5A10	写入自定义可存储字符串 1	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x753	0x11	0x5B00	查询自定义可存储字符串 2		
0x753	0x16	0x5C10	写入自定义可存储字符串 2	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x753	0x11	0x5D00	查询自定义可存储字符串 3		
0x753	0x16	0x5E10	写入自定义可存储字符串 3	ASCII 字符串	以 0x2E 结束, 无效填充 0x20
0x753	0x16	0x5F20	写入生产信息 (可选, 仅供生产商写入)	生产商:8bytes 生产地:8bytes 生产日期:8bytes 预留:8bytes	ASCII, 0x2E 结束, 无效填充 0x20 ASCII, 0x2E 结束, 无效填充 0x20 ASCII, YYYYMMDD 填充 0x00
0x753	0x16	0x6010	写入 Model (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x753	0x16	0x6110	写入 SN (可选, 仅供生产商写入)	ASCII 字符	结束符为 0x2E, 无效填充 0x20
0x753	0x16	0x6205	复位指令	ASCII 字符串	RESET
0x753	0x11	0x6308	读取存储器指定地址数据	起始地址:4bytes 结束地址:4bytes	读取数据大小≤128Bytes
0x753	0x16	0x6609	清除开机密码	ASCII 字符	PSW_CLEAR

注: CDL 发送的所有指令均按照定时 200ms 发送, 其中查询指令收到返回的信息或

超时 1s 停止发送，写入指令收到通用反馈指令或超时 1s 停止发送。

仅供内部传阅

5 附录 1: CRC32 计算方法

5.1 CRC32 计算多项式表

```
93. uint32_t Crc32Table[ 256 ] =
94. {
95.     0x00000000, 0x04C11DB7, 0x09823B6E, 0x0D4326D9, 0x130476DC, 0x17C56B6B,
96.     0x1A864DB2, 0x1E475005, 0x2608EDB8, 0x22C9F00F, 0x2F8AD6D6, 0x2B4BCB61,
97.     0x350C9B64, 0x31CD86D3, 0x3C8EA00A, 0x384FBDDB, 0x4C11DB70, 0x48D0C6C7,
98.     0x4593E01E, 0x4152FDA9, 0x5F15ADAC, 0x5BD4B01B, 0x569796C2, 0x52568B75,
99.     0x6A1936C8, 0x6ED82B7F, 0x639B0DA6, 0x675A1011, 0x791D4014, 0x7DDC5DA3,
100.    0x709F7B7A, 0x745E66CD, 0x9823B6E0, 0x9CE2AB57, 0x91A18D8E, 0x95609039,
101.    0x8B27C03C, 0x8FE6DD8B, 0x82A5FB52, 0x8664E6E5, 0xBE2B5B58, 0xBAEA46EF,
102.    0xB7A96036, 0xB3687D81, 0xAD2F2D84, 0xA9EE3033, 0xA4AD16EA, 0xA06C0B5D,
103.    0xD4326D90, 0xD0F37027, 0xDDB056FE, 0xD9714B49, 0xC7361B4C, 0xC3F706FB,
104.    0xCEB42022, 0xCA753D95, 0xF23A8028, 0xF6FB9D9F, 0xFBB8BB46, 0xFF79A6F1,
105.    0xE13EF6F4, 0xE5FFEB43, 0xE8BCCD9A, 0xEC7DD02D, 0x34867077, 0x30476DC0,
106.    0x3D044B19, 0x39C556AE, 0x278206AB, 0x23431B1C, 0x2E003DC5, 0x2AC12072,
107.    0x128E9DCF, 0x164F8078, 0x1B0CA6A1, 0x1FCD8B16, 0x018AEB13, 0x054BF6A4,
108.    0x0808D07D, 0x0CC9CDA, 0x7897AB07, 0x7C56B6B0, 0x71159069, 0x75D48DDE,
109.    0x6B93DDDB, 0x6F52C06C, 0x6211E6B5, 0x66D0FB02, 0x5E9F46BF, 0x5A5E5B08,
110.    0x571D7DD1, 0x53DC6066, 0x4D9B3063, 0x495A2DD4, 0x44190B0D, 0x40D816BA,
111.    0xACA5C697, 0xA864DB20, 0xA527FDF9, 0xA1E6E04E, 0xBFA1B04B, 0xBB60ADFC,
112.    0xB6238B25, 0xB2E29692, 0x8AAD2B2F, 0x8E6C3698, 0x832F1041, 0x87EE0DF6,
113.    0x99A95DF3, 0x9D684044, 0x902B669D, 0x94EA7B2A, 0xE0B41DE7, 0xE4750050,
114.    0xE9362689, 0xEDF73B3E, 0xF3B06B3B, 0xF771768C, 0xFA325055, 0xFE34DE2,
115.    0xC6BCF05F, 0xC27DEDE8, 0xCF3ECB31, 0xCBFFD686, 0xD5B88683, 0xD1799B34,
116.    0xDC3ABDED, 0xD8FBA05A, 0x690CE0EE, 0x6DCDFD59, 0x608EDB80, 0x644FC637,
117.    0x7A089632, 0x7EC98B85, 0x738AAD5C, 0x774BB0EB, 0x4F040D56, 0x4BC510E1,
118.    0x46863638, 0x42472B8F, 0x5C007B8A, 0x58C1663D, 0x558240E4, 0x51435D53,
119.    0x251D3B9E, 0x21DC2629, 0x2C9F00F0, 0x285E1D47, 0x36194D42, 0x32D850F5,
120.    0x3F9B762C, 0x3B5A6B9B, 0x0315D626, 0x07D4CB91, 0x0A97ED48, 0x0E56F0FF,
121.    0x1011A0FA, 0x14D0BD4D, 0x19939B94, 0x1D528623, 0xF12F560E, 0xF5EE4BB9,
122.    0xF8AD6D60, 0xFC6C70D7, 0xE22B20D2, 0xE6EA3D65, 0xEBA91BBC, 0xEF68060B,
123.    0xD727BBB6, 0xD3E6A601, 0xDEA580D8, 0xDA649D6F, 0xC423CD6A, 0xC0E2D0DD,
124.    0xCDA1F604, 0xC960EBB3, 0xBD3E8D7E, 0xB9FF90C9, 0xB4BCB610, 0xB07DABA7,
125.    0xAE3AFBA2, 0xAABFE615, 0xA7B8C0CC, 0xA379DD7B, 0x9B3660C6, 0x9FF77D71,
126.    0x92B45BA8, 0x9675461F, 0x8832161A, 0x8CF30BAD, 0x81B02D74, 0x857130C3,
127.    0x5D8A9099, 0x594B8D2E, 0x5408ABF7, 0x50C9B640, 0x4E8EE645, 0x4A4FFBF2,
128.    0x470CDD2B, 0x43CDC09C, 0x7B827D21, 0x7F436096, 0x7200464F, 0x76C15BF8,
129.    0x68860BFD, 0x6C47164A, 0x61043093, 0x65C52D24, 0x119B4BE9, 0x155A565E,
130.    0x18197087, 0x1CD86D30, 0x029F3D35, 0x065E2082, 0x0B1D065B, 0x0FDC1BEC,
131.    0x3793A651, 0x3352BBE6, 0x3E119D3F, 0x3AD08088, 0x2497D08D, 0x2056CD3A,
132.    0x2D15EBE3, 0x29D4F654, 0xC5A92679, 0xC1683BCE, 0xCC2B1D17, 0xC8EA00A0,
```

MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 123 页
133. 0xD6AD50A5, 0xD26C4D12, 0xDF2F6BCB, 0xDBEE767C, 0xE3A1CBC1, 0xE760D676, 134. 0xEA23F0AF, 0xEEE2ED18, 0xF0A5BD1D, 0xF464A0AA, 0xF9278673, 0xFDE69BC4, 135. 0x89B8FD09, 0x8D79E0BE, 0x803AC667, 0x84FBDDBD, 0x9ABC8BD5, 0x9E7D9662, 136. 0x933EB0BB, 0x97FFAD0C, 0xAFB010B1, 0xAB710D06, 0xA6322BDF, 0xA2F33668, 137. 0xBCB4666D, 0xB8757BDA, 0xB5365D03, 0xB1F740B4 138. };			
5.2 CRC32 计算方法			
39. uint32_t CRC32_Calculate(uint8_t *pData, uint16_t Length) 40. { 41. uint32_t nReg; 42. uint32_t nTemp = 0; 43. uint16_t i, n; 44. 45. nReg = 0xFFFFFFFF; 46. for (n = 0; n < Length; n++) 47. { 48. nReg ^= (uint32_t) pData[n]; 49. for (i = 0; i < 4; i++) 50. { 51. nTemp = Crc32Table[(uint8_t)((nReg >> 24) & 0xFF)]; 52. nReg <<= 8; 53. nReg ^= nTemp; 54. } 55. } 56. return nReg; 57. }			
保密文件， 禁止外传			

6 附录 2: 警告/故障码列表

当总线上任何一个设备上报警告/故障代码时,显示设备按照下列规则进行转换后,在屏幕上依次循环显示代码,间隔周期为 1s,显示顺序以接收到代码的先后顺序为准,当超时 3s 未接受到任何警告/故障代码时恢复原显示界面。

表9 警告/故障码列表

信号来源	显示代码	类型	描述	显示内容	解决办法
MC	10	警告	过流保护	Over Current Protect	5s 后自动恢复
MC	11	警告	低压保护	Under Voltage Protect	电池充电
MC	12	警告	过压保护	Over Voltage Protect	更换正确电池
MC	13	警告	堵转保护	Rotor Locked	关机重启
MC	14	警告	过热保护	Over Heat Protect	关机静置 30min 后使用
MC	15	故障	温度传感器故障	NTC Fault	返修
MC	16	故障	速度传感器故障	Speed Sensor Fault	检查或更换速度传感器
MC	17	故障	力矩传感器故障	Torque Sensor Fault	返修
MC	18	故障	马达霍尔传感器故障	Motor Fault	返修
MC	19	警告	BMS 校验失败	BMS Check Fault	更换电池
MC	20	警告	OBC 校验失败	OBC Check Fault	更换按键
MC	22	故障	马达缺相	PhaseLine Fault	返修
MC	23	故障	踏频传感器故障	Cadence Sensor Fault	返修
MC	24	故障	指拨故障	Gas Sensor Fault	更换指拨
MC	25	故障	MOS 短路	MOS Short Circuit	返修
MC	26	故障	电压异常波动	Bus Voltage Abnormal	更换电池
MC	27	故障	处理器故障	MCU Fault	返修
MC	28	故障	电路故障	Circuit Fault	返修
MC	31	警告	MC 校验失败	MC Check Fault	返修
MC	32	故障	马达失速	Motor Stall	返修
MC	33	故障	预留故障位	-	-
BMS	40	警告	预留故障位	-	-
BMS	41	警告	充电过压警告	Charge Over Voltage Alarm	停止充电或更换充电器
BMS	42	警告	放电低压警告	DisCharge Under Voltage Alarm	及时充电
BMS	43	警告	充电过流警告	Charge Over Current Alarm	更换充电器
BMS	44	警告	放电过流警告	Over Current Alarm	停止骑行后故障消失

BMS	45	警告	充电高温警告	Charge Over Heat Alarm	停止充电
BMS	46	警告	充电低温警告	Charge Low Temperature Alarm	停止充电
BMS	47	警告	放电高温警告	DisCharge Over Heat Alarm	关机静置 30min 后使用
BMS	48	警告	放电低温警告	DisCharge Low Temperature Alarm	建议停止使用
BMS	49	警告	MOS 高温警告	MOS Over Heat Alarm	5s 后自动恢复
OBC	60	故障	+按键失效	+ Key Fault	检查或更换按键
OBC	61	故障	-键失效	- Key Fault	检查或更换按键
OBC	62	故障	i 键失效	- Key Fault	检查或更换按键
OBC	63	故障	Light 键失效	Light Key Fault	检查或更换按键
OBC	64	故障	Walk 键失效	Walk Key Fault	检查或更换按键
OBC	65	故障	电源键失效	Power Key Fault	检查或更换按键
OBC	70	故障	MC 通讯异常	MC Communication Fault	检查线缆或更换部件
OBC	71	故障	预留故障位 1		
OBC	72	故障	预留故障位 2		

7 附录 3: 升级协议及流程

7.1 升级流程图

系统 CAN 总线上除 CDL 外所有设备均需通过 CDL 完成升级, 升级流程图如下图所示:

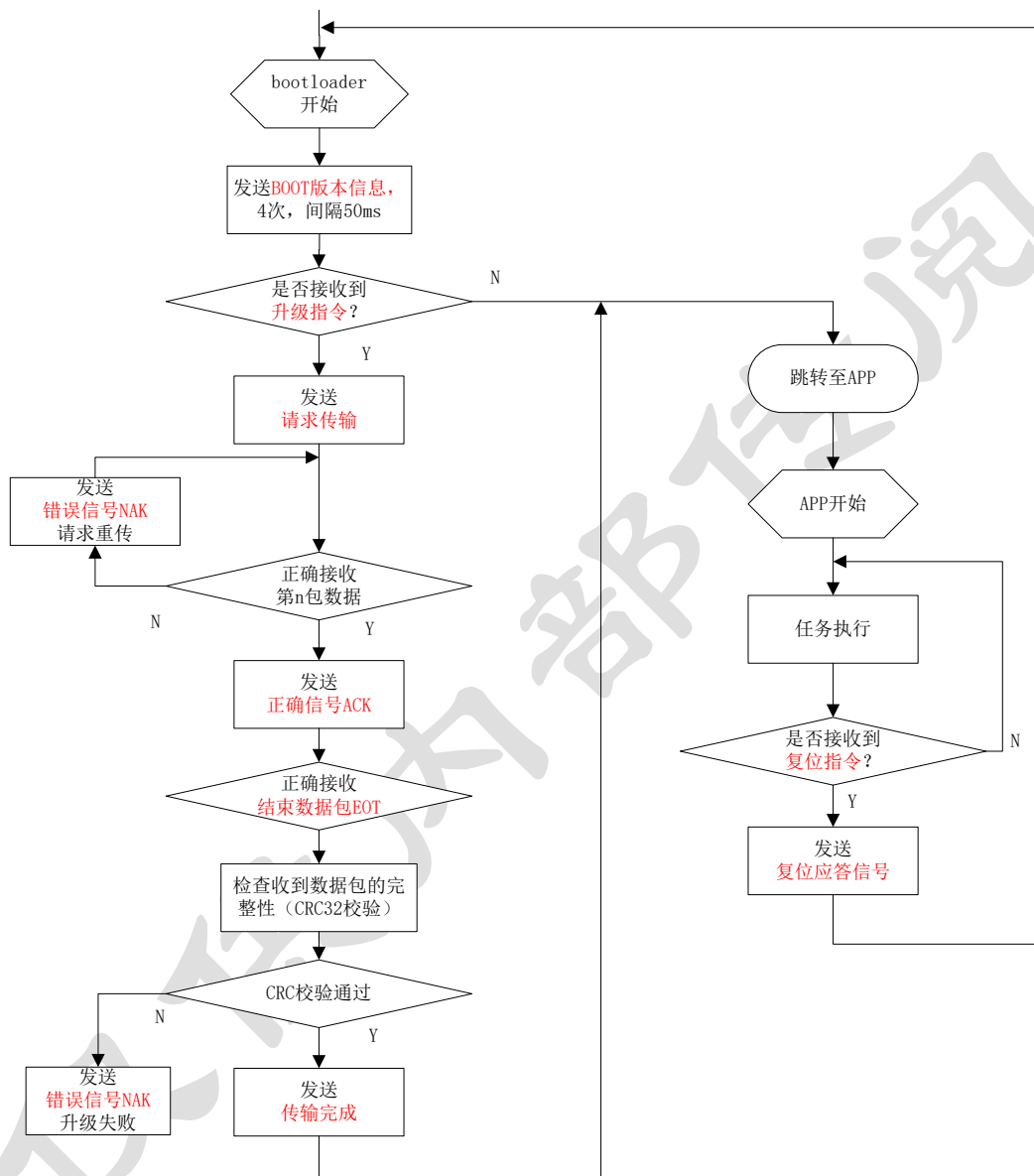


图3 升级流程图

7.2 升级协议

APP 程序中复位指令和复位应答指令参考第 3 章有关 CDL 发送给各设备的复位指令, 以及各设备发送给 CDL 的通用应答指令。

Bootloader 中的协议描述如下:

表10 CDL 发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
升级指令	MC	0x751	0x16	0x0D	0xA10B	ASCII 字符: "MC_UPD" + 数据包大小 4 字节 (如 78351B, 内容为: 0F 32 01 00)
	BMS	0x752				ASCII 字符: "BMS_UPD" + 数据包大小 4 字节
	OBC	0x753				ASCII 字符: "OBC_UPD" + 数据包大小 4 字节
数据包格式	MC	0x751	0x16	0x87	0xA385	SOH(01)+序号(1~65535)+总包数(1~65535)+数据(长度 128B, 无效填充 0xFF)
	BMS	0x752				
	OBC	0x753				
结束数据包	MC	0x751	0x16	0x03	0xA401	EOT(04)
	BMS	0x752				
	OBC	0x753				

表11 设备发出指令

指令功能	设备名称	ID	模式	命令段长度	命令字	数据段
BOOT 版本信息	MC	0x715	0x0C	0x0B	0xC109	ASCII 字符: "MC "+"Vx.x.x"
	BMS	0x725				ASCII 字符: "BMS"+"Vx.x.x"
	OBC	0x735				ASCII 字符: "OBC"+"Vx.x.x"
正确信号	MC	0x715	0x0C	0x04	0xC202	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
错误信号	MC	0x715	0x0C	0x04	0xC302	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
请求传输	MC	0x715	0x0C	0x04	0xC402	0x00 0x00
	BMS	0x725				
	OBC	0x735				
传输完成	MC	0x715	0x0C	0x04	0xC502	0x00 0x00
	BMS	0x725				
	OBC	0x735				


```
8.     if(AddressBegin <= AddressEnd)
9.     {
10.        DataLength = AddressEnd - AddressBegin+ 1;
11.        memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.        SendData(ID_MC_TO_CDL, MODE_REPORT, (0xAB08 + DataLength), (uint8_t*)Data);
13.    }
14.    }while(0);
15.    break;
16. }
```

8.4 数据结构定义

```
1. //故障日志记录信息 64 bytes, 1K 空间可以保存最近 16 条故障记录
2. typedef struct
3. {
4.     UWORD      ErrorIndex; // 故障序列号, 2bytes, 地址偏移 0
5.     UWORD      NotesInfo1; // 备注信息 1, 2bytes, 地址偏移 2
6.     UWORD      NotesInfo2; // 备注信息 2, 2bytes, 地址偏移 4
7.     UWORD      NotesInfo3; // 备注信息 3, 2bytes, 地址偏移 6
8.     ULONG      ErrorCode;  // 故障码, 4 bytes, 地址偏移 8
9.     ULONG      RunTime;    // 运行时间, 4bytes, 地址偏移 12
10.    MC_RunInfo_Struct_t RunInfo; // 运行信息 32bytes, 地址偏移 16
11.    SWORD      IqCurrentPu; // Iq 电流, 2bytes, 地址偏移 48
12.    SWORD      IqVoltagePu; // Iq 电压, 2bytes, 地址偏移 50
13.    SWORD      IdCurrentPu; // Id 电流, 2bytes, 地址偏移 52
14.    SWORD      IdVoltagePu; // Id 电压, 2bytes, 地址偏移 54
15.    UWORD      RS1;         // 预留信息 1, 2bytes, 地址偏移 56
16.    UWORD      RS2;         // 预留信息 1, 2bytes, 地址偏移 58
17.    UWORD      RS3;         // 预留信息 1, 2bytes, 地址偏移 60
18.    UWORD      RS4;         // 预留信息 1, 2bytes, 地址偏移 62
19. } ERROR_LOG; //共 64 字节
20.
21. //电机运行信息, 占用空间 32bytes
22. typedef struct
23. {
24.     uint16_t BikeSpeed; //车速 0.1km/h, 地址偏移 0
25.     int16_t MotorSpeed; //输出转速 1rpm, 地址偏移 2
26.     uint16_t Power; //电功率 1W, 地址偏移 4
27.     uint16_t BusVoltage; //母线电压 1mV, 地址偏移 6
28.     uint16_t BusCurrent; //母线电流 1mA, 地址偏移 8
29.     uint8_t Cadence; //踏频 1rpm, 地址偏移 10
30.     uint8_t Torque; //踩踏力矩 1Nm, 地址偏移 11
31.    MC_CadenceDir_Struct_t CadenceDir; //踩踏方向 0-正,1-反,2-停止, 地址偏移 12
```

MOTINOVA	文件名称: MOTINOVA E-Bike 电气系统通信协议		密级: 秘密
	文件编号: [填入文件编号]	版本: V4.2	共 133 页 第 130 页
32. uint8_t GearSt; //助力档位, 地址偏移 13 33. MC_LightSwitch_Struct_t LightSwitch; //灯开关 0xF0-关, 0xF1-开, 地址偏移 14 34. uint8_t SOC; //剩余电量 1%, 地址偏移 15 35. uint16_t RemainDistance; //续航里程 1km, 地址偏移 16 36. uint16_t ODO_Km; //总里程 1km, 地址偏移 18 37. uint8_t PowerPerKm; //平均功耗 0.01Ah/km , 地址偏移 20 38. uint8_t T_PCB; //PCB 温度 +40℃, 地址偏移 21 39. uint8_t T_Coil; //绕组温度 +40℃, 地址偏移 22 40. uint8_t T_MCU; //MCU 温度 +40℃, 地址偏移 23 41. uint16_t Ride_Km; //开机后骑行里程 0.1km, 地址偏移 24 42. uint16_t Ride_Time; //开机后骑行时间 1s, 地址偏移 26 43. uint8_t TorqueSensorData1; //力矩传感器 1 原始值 44. uint8_t TorqueSensorData2; //力矩传感器 2 原始值 45. uint8_t TorqueSensorData3; //力矩传感器 3 原始值 46. uint8_t RS ; //预留 1byte 47. }MC_RunInfo_Struct_t;			
仅供内部			
保密文件, 禁止外传			


```
10.         DataLength = AddressEnd - AddressBegin+ 1;
11.         memcpy((uint8_t*)(Data + 8), (uint8_t*)(AddressBegin), DataLength);
12.         SendData(ID_MC_TO_CDL, MODE_REPORT, (0x5608 + DataLength), (uint8_t*)Data);
13.     }
14. }while(0);
15. break;
16. }
```

9.4 数据结构定义

```
1. //故障日志记录信息 128 bytes, 2K 空间可以保存最近 16 条故障记录
2. typedef struct
3. {
4.     uint16_t Error_Index;           //故障列表索引, 2 bytes, 地址偏移 0
5.     uint16_t NotesInfo1;           //备注信息 1, 2 Bytes, 地址偏移 2
6.     uint16_t NotesInfo2;           //备注信息 2, 2 Bytes, 地址偏移 4
7.     uint16_t NotesInfo3;           //备注信息 3, 2Bytes, 地址偏移 6
8.     uint32_t ErrorCode;             //故障码, 4 bytes, 地址偏移 8
9.     uint16_t CellVoltage_1;         //电芯 1 电压, 单位 mV, 2 Bytes, 地址偏移 12
10.    uint16_t CellVoltage_2;         //电芯 2 电压, 单位 mV, 2 Bytes, 地址偏移 14
11.    uint16_t CellVoltage_3;         //电芯 3 电压, 单位 mV, 2 Bytes, 地址偏移 16
12.    uint16_t CellVoltage_4;         //电芯 4 电压, 单位 mV, 2 Bytes, 地址偏移 18
13.    uint16_t CellVoltage_5;         //电芯 5 电压, 单位 mV, 2 Bytes, 地址偏移 20
14.    uint16_t CellVoltage_6;         //电芯 6 电压, 单位 mV, 2 Bytes, 地址偏移 22
15.    uint16_t CellVoltage_7;         //电芯 7 电压, 单位 mV, 2 Bytes, 地址偏移 24
16.    uint16_t CellVoltage_8;         //电芯 8 电压, 单位 mV, 2 Bytes, 地址偏移 26
17.    uint16_t CellVoltage_9;         //电芯 9 电压, 单位 mV, 2 Bytes, 地址偏移 28
18.    uint16_t CellVoltage_10;        //电芯 10 电压, 单位 mV, 2 Bytes, 地址偏移 30
19.    uint16_t CellVoltage_11;        //电芯 11 电压, 单位 mV, 2 Bytes, 地址偏移 32
20.    uint16_t CellVoltage_12;        //电芯 12 电压, 单位 mV, 2 Bytes, 地址偏移 34
21.    uint16_t CellVoltage_13;        //电芯 13 电压, 单位 mV, 2 Bytes, 地址偏移 36
22.    uint16_t CellVoltage_14;        //电芯 14 电压, 单位 mV, 2 Bytes, 地址偏移 38
23.    uint16_t CellVoltage_15;        //电芯 15 电压, 单位 mV, 2 Bytes, 地址偏移 40
24.    uint16_t CellVoltage_16;        //电芯 16 电压, 单位 mV, 2 Bytes, 地址偏移 42
25.    int32_t ChargeCurrent;           //充电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 44
26.    int32_t DisChargeCurrent;        //放电电流, 单位 mA, 放电为负值, 4Bytes, 地址偏移 48
27.    uint32_t FullChargeCapacity;     //满充容量, 单位 mAh, 4 Bytes, 地址偏移 52
28.    uint32_t RemainCapacity;         //剩余容量, 单位 mAh, 4 Bytes, 地址偏移 56
29.    uint16_t CycleCount;             //循环次数, 2 Bytes, 地址偏移 60
30.    uint8_t Temperature_1;          //温度 1, 单位°C, 偏移 40, 1 Byte, 地址偏移 62
31.    uint8_t Temperature_2;          //温度 2, 单位°C, 偏移 40, 1 Byte, 地址偏移 63
32.    uint8_t Temperature_3;          //温度 3, 单位°C, 偏移 40, 1 Byte, 地址偏移 64
```

```
33.    uint8_t Temperature_4;           //温度 4, 单位°C, 偏移 40, 1 Byte, 地址偏移 65
34.    uint8_t SOC;                     //剩余电量, 1 Byte, 地址偏移 66
35.    uint8_t SOH;                     //电荷寿命, 1 Byte, 地址偏移 67
36.    uint8_t AFE_Status;              //AFE 状态, 1 Byte, 地址偏移 68
37.    uint8_t Working_State;           //工作状态, 1 Byte, 地址偏移 69
38.    uint16_t MaxChargeTime;           //最大充电持续时间, 单位 min, 地址偏移 70
39.    uint16_t MaxBetweenChargeTime;    //最大充电间隔时间, 单位 min, 地址偏移 72
40.    uint16_t MaxBetweenDisChargeTime; //最大放电持续时间, 单位 min, 地址偏移 74
41.    uint16_t LastBetweenDisChargeTime; //最大放电间隔时间, 单位 min, 地址偏移 76
42.    uint16_t MaxUVPTIME;             //最大欠压保护间隔时间, 单位 min, 地址偏移 78
43.    RTC_Struct_t RTC;                //实时时钟, 8 Bytes, 地址偏移 80
44.    RTC_Struct_t LastChargeTime;      //最近充电时钟, 8 Bytes, 地址偏移 88
45.    RTC_Struct_t LastDisChargeTime;    //最近放电时钟, 8 Bytes, 地址偏移 96
46.    RTC_Struct_t LastUVP_RTC;         //最近欠压保护时钟, 8 Bytes, 地址偏移 104
47.    RTC_Struct_t LastUVP_Active_RTC;  //最近欠压保护激活时钟, 8 Bytes, 地址偏移 112
48.    RTC_Struct_t LastFCC_Update_RTC  //最近满充容量更新时钟, 8 Bytes, 地址偏移 120
49. }BMS_ErrorLogSaveInfo_Struct_t;
50.
51. //RTC 数据结构
52. typedef struct
53. {
54.    uint8_t RS1;                       //预留, 0x00
55.    uint8_t Year;                      //年
56.    uint8_t Mouth;                    //月
57.    uint8_t Date;                     //日
58.    uint8_t RS2;                       //预留, 0x00
59.    uint8_t Hour;                     //时
60.    uint8_t Minute;                   //分
61.    uint8_t Second;                   //秒
62. }
```