

Welling 中置驱动系统通信协议 (OBC)

[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

二〇二二年五月十二日

修改记录

修改日期	修改人	修改内容	版本号
20220512	周雄	1、引用《MOTINOVA 中置驱动系统通信协议_V3r3_20211014》进行修改, 删除原 MOTINOVA 特有的通信内容, 增加 Welling 电机特有的通信内容, 第一次发布; 2、修改电机控制器的缩写为 WMC, 用以区分 MOTINOVA 控制器; 3、WMC 运行信息中, 删除 ODO 信息, 改为预留; 4、BMS 运行信息中, 增加 SOH 信息; 5、删除所有 HMI 相关指令; 6、增加 OBC 查询 BMS 电芯电压指令。	V1.0.0

Welling 中置驱动系统通信协议 (OBC)

1 系统组成

WMC: 电机控制器 Welling 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

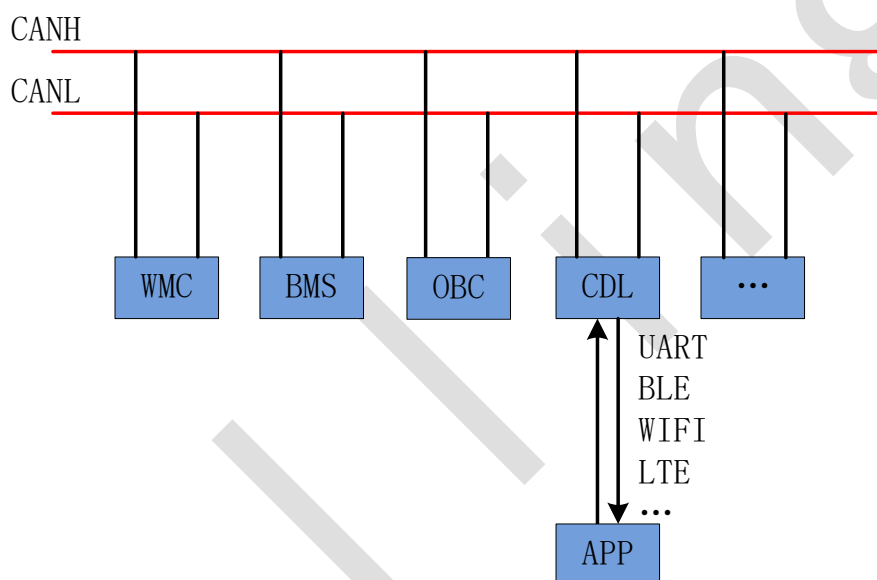


图1 系统通信接口示意图

2 通信协议规则

本协议主要描述 Welling 中置驱动系统各组件之间数据通信格式，只适用于 Welling 中置驱动系统内部组件之间通信。

2.1 硬件接口

接口类型: CAN2.0A

波特率: 250kbps

2.2 数据帧封装格式

2.2.1 数据帧格式

协议描述了每帧数据内容，包括帧头、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:

- 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。

2.2.2 ID 分配

表2 ID 分配

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

2.2.3 封装方式

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 通信内容

3.1 WMC 命令字定义

表4 WMC 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	电机运行信息 (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 开机后骑行里程:2bytes 开机后骑行时间:2byte 预留:4byte	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 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 1%, 无效为 0xFF 1km, 无效为 0xFFFF 填充 0x00 0.01Ah/km, 无效为 0xFF +40℃ +40℃ +40℃ 0.1km 1s 填充 0x00
0x710	0x0C	0x1104	电机故障码 (存在故障时)	0:无故障 低 16 位:	按位或输出, 0-正常, 1-故障

			200ms 自动发送, 故障消失后停止发送)	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:速度传感器故障 0x0040:力矩传感器故障 0x0080:霍尔传感器故障 0x0100:马达缺相 0x0200:NTC 故障 0x0400:BMS 校验失败 0x0800:预留故障 1 0x1000:OBC 校验失败 0x2000:MCU 故障 0x4000:踏频传感器故障 0x8000:指拨故障 高 16 位: 0x0001:MOS 短路 0x0002:电压异常 0x0004:电路故障 0x0008:预留故障 2 0x0010:预留故障 3 0x0020:预留故障 4 0x0040:预留故障 5 0x0080:预留故障 6	
0x710	0x0C	0x1240	电机版本信息 (返回指令)	ASCII 字符	排列顺序为: MODEL、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 Vxrxx_YYYYMMDD
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 0x08:预留 0x10:预留	按位或输出, 0-在线, 1-离线

Welling			文件名称：Welling 中置驱动系统通信协议（OBC）			密级：秘密
			文件编号：[填入文件编号]		版本：V1.0.0	共 17 页 第 6 页
				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	0x1808	关机指令 （控制器关机 前发送）	ASCII 字符	SHUTDOWN	
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 预留:3bytes	1cm 1-柔和，2-正常，3- 强劲 1km/h ±10cm，有符型 1-2 填充 0x00	

3.2 BMS 命令字定义

表5 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	1mV 1mA，有符型，放电 为负，充电为正 1mAh

				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:5bytes	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 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	按位或输出, 0-正 常, 1-故障

Welling		文件名称: Welling 中置驱动系统通信协议 (OBC)			密级: 秘密
		文件编号: [填入文件编号]		版本: V1.0.0	共 17 页 第 8 页
				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 0x1000:一级过流警告 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: MODEL、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.

3.3 OBC 命令字定义

表6 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	OBC 版本信息 (返回指令)	ASCII 字符	排列顺序为: MODEL、 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
0x730	0x0C	0x120C	OBC 物理 ID (返回指令)	ID:12bytes	不足位填充 1
0x730	0x0C	0x130C	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	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: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
发送给 WMC					
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 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
0x731	0x0C	0x3105	在线检测反馈 (返回指令)	ASCII 字符	READY
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 助力方案:1byte 预留:6bytes	有符型, 单位 1cm, ±10.0cm 1-柔和, 2-正常, 3-强劲 1-2 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	0x11	0x3900	OBC 读取电机版本信息		
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	OBC 查询 BMS 版本信息 (主动发送, 收到返回或超时停止)		
0x732	0x11	0x5200	OBC 查询 BMS 设		

			计信息(主动发送, 收到返回或超时停止)		
0x732	0x11	0x5300	OBC 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	OBC 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 预留:3Bits	0-松开 1-按下 多余位填充 0 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

3.4 CDL 命令字定义

表7 CDL 命令字定义

ID	模式	命令字	功能	数据段	备注
发送给 OBC					

0x753	0x11	0x5000	查询 OBC 物理 ID		
0x753	0x11	0x5100	查询 OBC 校验码		
0x753	0x16	0x520C	写入 OBC 校验码	校验码:12bytes	
0x753	0x16	0x5300	查询 OBC 生产信息		
0x753	0x11	0x5400	查询 OBC 版本信息		
0x753	0x16	0x5504	设置进入/退出测试模式	ASCII 字符	STAR:进入 STOP:退出
0x753	0x11	0x5600	查询 OBC 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 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 OBC 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 OBC 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

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。

4 附录 1: CRC32 计算方法

4.1 CRC32 计算多项式表

```
uint32_t Crc32Table[ 256 ] =
```

```
{
```

```
    0x00000000, 0x04C11DB7, 0x09823B6E, 0x0D4326D9, 0x130476DC, 0x17C56B6B,  
    0x1A864DB2, 0x1E475005, 0x2608EDB8, 0x22C9F00F, 0x2F8AD6D6, 0x2B4BCB61,  
    0x350C9B64, 0x31CD86D3, 0x3C8EA00A, 0x384FBDBD, 0x4C11DB70, 0x48D0C6C7,  
    0x4593E01E, 0x4152FDA9, 0x5F15ADAC, 0x5BD4B01B, 0x569796C2, 0x52568B75,  
    0x6A1936C8, 0x6ED82B7F, 0x639B0DA6, 0x675A1011, 0x791D4014, 0x7DDC5DA3,  
    0x709F7B7A, 0x745E66CD, 0x9823B6E0, 0x9CE2AB57, 0x91A18D8E, 0x95609039,  
    0x8B27C03C, 0x8FE6DD8B, 0x82A5FB52, 0x8664E6E5, 0xBE2B5B58, 0xBAEA46EF,  
    0xB7A96036, 0xB3687D81, 0xAD2F2D84, 0xA9EE3033, 0xA4AD16EA, 0xA06C0B5D,  
    0xD4326D90, 0xD0F37027, 0xDDB056FE, 0xD9714B49, 0xC7361B4C, 0xC3F706FB,  
    0xCEB42022, 0xCA753D95, 0xF23A8028, 0xF6FB9D9F, 0xFBB8BB46, 0xFF79A6F1,  
    0xE13EF6F4, 0xE5FFE643, 0xE8BCCD9A, 0xEC7DD02D, 0x34867077, 0x30476DC0,  
    0x3D044B19, 0x39C556AE, 0x278206AB, 0x23431B1C, 0x2E003DC5, 0x2AC12072,  
    0x128E9DCF, 0x164F8078, 0x1BOCA6A1, 0x1FCDBB16, 0x018AEB13, 0x054BF6A4,  
    0x0808D07D, 0x0CC9CDCA, 0x7897AB07, 0x7C56B6B0, 0x71159069, 0x75D48DDE,  
    0x6B93DDDB, 0x6F52C06C, 0x6211E6B5, 0x66D0FB02, 0x5E9F46BF, 0x5A5E5B08,  
    0x571D7DD1, 0x53DC6066, 0x4D9B3063, 0x495A2DD4, 0x44190B0D, 0x40D816BA,  
    0xACA5C697, 0xA864DB20, 0xA527FDF9, 0xA1E6E04E, 0xBFA1B04B, 0xBB60ADFC,  
    0xB6238B25, 0xB2E29692, 0x8AAD2B2F, 0x8E6C3698, 0x832F1041, 0x87EE0DF6,  
    0x99A95DF3, 0x9D684044, 0x902B669D, 0x94EA7B2A, 0xE0B41DE7, 0xE4750050,  
    0xE9362689, 0xEDF73B3E, 0xF3B06B3B, 0xF771768C, 0xFA325055, 0xFEF34DE2,  
    0xC6BCF05F, 0xC27DEDE8, 0xCF3ECB31, 0xCBFFD686, 0xD5B88683, 0xD1799B34,  
    0xDC3ABDED, 0xD8FBA05A, 0x690CE0EE, 0x6DCDFD59, 0x608EDB80, 0x644FC637,  
    0x7A089632, 0x7EC98B85, 0x738AAD5C, 0x774BB0EB, 0x4F040D56, 0x4BC510E1,  
    0x46863638, 0x42472B8F, 0x5C007B8A, 0x58C1663D, 0x558240E4, 0x51435D53,  
    0x251D3B9E, 0x21DC2629, 0x2C9F00F0, 0x285E1D47, 0x36194D42, 0x32D850F5,  
    0x3F9B762C, 0x3B5A6B9B, 0x0315D626, 0x07D4CB91, 0x0A97ED48, 0x0E56F0FF,  
    0x1011A0FA, 0x14D0BD4D, 0x19939B94, 0x1D528623, 0xF12F560E, 0xF5EE4BB9,  
    0xF8AD6D60, 0xFC6C70D7, 0xE22B20D2, 0xE6EA3D65, 0xEBA91BBC, 0xEF68060B,  
    0xD727BBB6, 0xD3E6A601, 0xDEA580D8, 0xDA649D6F, 0xC423CD6A, 0xC0E2D0DD,  
    0xCDA1F604, 0xC960EBB3, 0xBD3E8D7E, 0xB9FF90C9, 0xB4BCB610, 0xB07DABA7,  
    0xAE3AFBA2, 0xAAFBE615, 0xA7B8COCC, 0xA379DD7B, 0x9B3660C6, 0x9FF77D71,  
    0x92B45BA8, 0x9675461F, 0x8832161A, 0x8CF30BAD, 0x81B02D74, 0x857130C3,  
    0x5D8A9099, 0x594B8D2E, 0x5408ABF7, 0x50C9B640, 0x4E8EE645, 0x4A4FFBF2,  
    0x470CDD2B, 0x43CDC09C, 0x7B827D21, 0x7F436096, 0x7200464F, 0x76C15BF8,  
    0x68860BFD, 0x6C47164A, 0x61043093, 0x65C52D24, 0x119B4BE9, 0x155A565E,  
    0x18197087, 0x1CD86D30, 0x029F3D35, 0x065E2082, 0x0B1D065B, 0x0FDC1BEC,  
    0x3793A651, 0x3352BBE6, 0x3E119D3F, 0x3AD08088, 0x2A97D08D, 0x2056CD3A,  
    0x2D15EBE3, 0x29D4F654, 0xC5A92679, 0xC1683BCE, 0xCC2B1D17, 0xC8EA00A0,  
    0xD6AD50A5, 0xD26C4D12, 0xDF2F6BCB, 0xDBEE767C, 0xE3A1CBC1, 0xE760D676,
```

Welling	文件名称: Welling 中置驱动系统通信协议 (OBC)		密级: 秘密
	文件编号: [填入文件编号]	版本: V1.0.0	共 17 页 第 2 页
0xEA23F0AF, 0xEE2ED18, 0xF0A5BD1D, 0xF464A0AA, 0xF9278673, 0xFDE69BC4, 0x89B8FD09, 0x8D79E0BE, 0x803AC667, 0x84FBDBD0, 0x9ABC8BD5, 0x9E7D9662, 0x933EB0BB, 0x97FFAD0C, 0xAFB010B1, 0xAB710D06, 0xA6322BDF, 0xA2F33668, 0xBCB4666D, 0xB8757BDA, 0xB5365D03, 0xB1F740B4 }; 4.2 CRC32 计算方法 <pre>uint32_t CRC32_Calculate(uint8_t *pData, uint16_t Length) { uint32_t nReg; uint32_t nTemp = 0; uint16_t i, n; nReg = 0xFFFFFFFF; for (n = 0; n < Length; n++) { nReg ^= (uint32_t) pData[n]; for (i = 0; i < 4; i++) { nTemp = Crc32Table[(uint8_t)((nReg >> 24) & 0xFF)]; nReg <<= 8; nReg ^= nTemp; } } return nReg; }</pre>			

5 附录 3: 升级协议及流程

5.1 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:

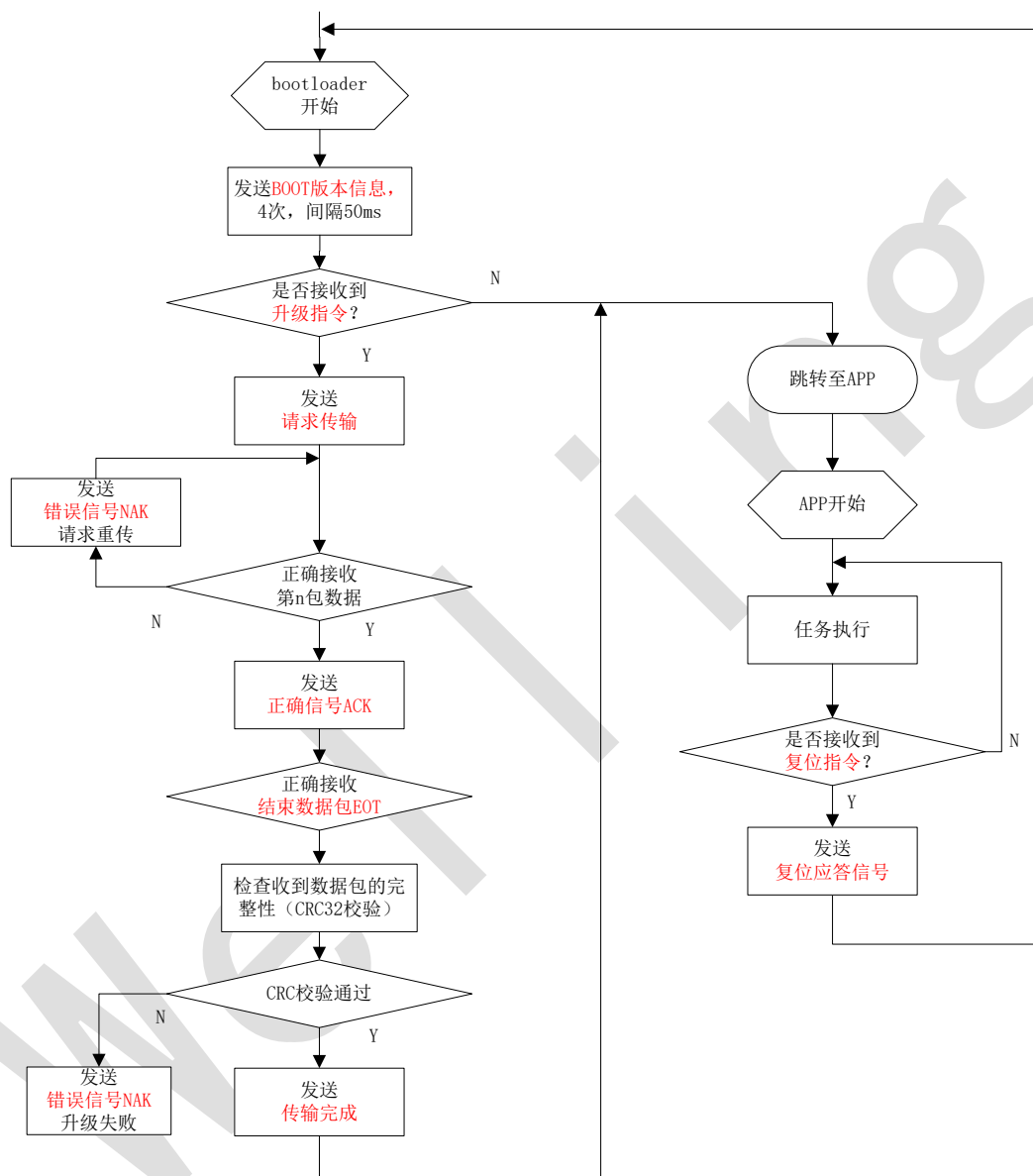


图2 升级流程图

5.2 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8 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	WMC	0x751	0x16	0x0D	0xA10B	ASCII 字符: "WMC_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	WMC	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	WMC	0x751	0x16	0x03	0xA401	EOT(04)
	BMS	0x752				
	OBC	0x753				

表9 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	WMC	0x715	0x0C	0x0B	0xC109	ASCII 字符: "WMC"+"Vx.x.x"
	BMS	0x725				ASCII 字符: "BMS"+"Vx.x.x"
	OBC	0x735				ASCII 字符: "PBU"+"Vx.x.x"
正确信号	WMC	0x715	0x0C	0x04	0xC202	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
错误信号	WMC	0x715	0x0C	0x04	0xC302	当前包号, 2byte
	BMS	0x725				
	OBC	0x735				
请求传输	WMC	0x715	0x0C	0x04	0xC402	0x00 0x00
	BMS	0x725				
	OBC	0x735				
传输完成	WMC	0x715	0x0C	0x04	0xC502	0x00 0x00
	BMS	0x725				
	OBC	0x735				