Hobbywing Scooter BLE Protocol B-01.0.01

版本修改记录

版本	日期	修改内容	修订人
01.0.01	2024-12-13	首次发布	李 勇

2024-12-13

目录

1 蓝牙服务 UUID

1.1 透传数据服务

透传服务 UUID：0000f1f0-0000-1000-8000-00805f9b34fb

透传接收 UUID：0000f1f1-0000-1000-8000-00805f9b34fb（write）

透传发送 U UID：0000f1f2-0000-1000-8000-00805f9b34fb（notify）

1.2 AT 指令服务

AT 服务　UUID：0000f2f0-0000-1000-8000-00805f9b34fb

AT 接收　UUID：0000f2f1-0000-1000-8000-00805f9b34fb（write）

AT 发送　UUID：0000f2f2-0000-1000-8000-00805f9b34fb（notify）

1.3 仪表升级服务

升级服务 UUID：f000ffc0-0451-4000-b000-000000000000

升级接收 UUID：f000ffc1-0451-4000-b000-000000000000

升级发送 UUID：f000ffc2-0451-4000-b000-000000000000

2 仪表和 APP 通信

2.1 仪表定时上报 指令码 0x00 数据格式

序号	字节定义	样例	内容定义	备注
0	包头	0xAB		
1	指令码	0x00		
2	帧总字节数	0x19		0x19=25
3	电机方向	0x01	取值 0~1	1-正方向，0-反方向
4	档位	0x01	取值 0~3	1-表示当前 2 档
5	电量	0x64	取值 0-100	0x64=100 表示电量 100%
6	速度 1 高字节	0x27	取值 0-65535	0x2710=10000 表示电机 1 速度为 10km/h 或 10mph（具体单位参考下表 bit6 说明）
7	速度 1 低字节	0x10		
8	速度 2 高字节	0x27	取值 0-65535	0x2710=10000 表示电机 2 速度为 10km/h 或 10mph（具体单位参考下表 bit6 说明）
9	速度 2 低字节	0x10		
10	电压高字节	0x00		0x0064=100 表示 10V，

11	电压低字节	0x64		单位是 0.1V
12	电流高字节	0x00		Q6 格式，0x0040=64 表示 1A
13	电流低字节	0x40		
14	电调温度	0x0A		0x0A=10 表示 10 摄氏度
15	电机温度	0x0A		0x0A=10 表示 10 摄氏度
16	小计里程高字节	0x00		0x0064=100 表示 10.0km 或 10.0mile（具体单位参考下表 bit6 说明）
17	小计里程低字节	0x64		
18	总里程高 16 位字节	0x00		0x0003E8=1000 表示 100.0km 或 100.0mile（具体单位参考下表 bit6 说明）
19	总里程高 8 位	0x03		
20	总里程低 8 位	0xE8		
21	控制器状态寄存器高字节			见下文定义
22	控制器状态寄存器低字节			
23	CRC16 低字节			MODBUS——CRC16
24	CRC16 高字节			

控制器状态寄存器：

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
氛围灯开关状态	左转向灯	右转向灯	定速到	锁定电机	开关	定速开关	蜂鸣器 1	蜂鸣器 0	公制英制	零启动	尾灯 1	尾灯 0	前灯开关	档位设定 1	档位设定 0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Bit7，bit8，蜂鸣器，0 不响，1，一短鸣，2，两短鸣，3，一长鸣
Bit11，当该状态位 0 时，表示电机锁定，默认 1，电机解锁
Bit9，1：表示定速巡航打开，
Bit6，0：表示公制 ，1：表示英制
Bit5，0：表示滑行启动，1：表示零起步
Bit2，1：表示大灯开，
Bit1，bit0，0：表示节能，1：表示正常，2：表示运动

2.2 仪表定时上报 指令码 0x01 数据格式

序号	字节定义	样例	内容定义	备注
0	包头	0xAB		
1	指令码	0x01		
2	帧总字节数	0x19		0x19=25
3	进入定速巡航允许最低速度	0x03		在公制格式下，3 表示 3KM/H，在英制格式下 3 表示 3MPH
4	节能模式最高速限制	0x0f		在公制格式下，15 表示 15KM/H，在英制格式下 15 表示 15MPH
5	舒适模式最高速限制	0x16		在公制格式下，22 表示 22KM/H，在英制格式下 22 表示 22MPH
6	运动模式最高速限制	0x1f		在公制格式下，31 表示 31KM/H，在英制格式下 31 表示 31MPH
7	未定义	0x00		
8	故障标志高字节	0x00		
9	故障标志低字节	0x00		
10	面板选择高字节	0x00		
11	面板选择低字节	0x00		
12	未定义	0x00		
13	未定义	0x00		
14	未定义	0x00		
15	未定义	0x00		
16	未定义	0x00		
17	未定义	0x00		
18	仪表软件版本号	0x80		表示 8025_01.00.01
19	仪表软件版本号	0x25		
20	仪表软件版本号	0x01		
21	仪表软件版本号	0x00		
22	仪表软件版本号	0x01		
23	CRC16 低字节			MODBUS——CRC16
24	CRC16 高字节			

故障标志：

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
故障预警使能位				油门指拨未复位F2	刹车指拨未复位F1	运放偏置E9		霍尔故障E7			过流E4	通讯断连E3	油门故障E2	刹车故障E1	
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

面板选择：

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
			速度单位选择0-m/h 1-100m/h	BMS	RGB面板	MP3面板	SN码面板								
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0--隐藏对应的面板功能
1--加载对应的面板功能

Bit12：速度单位选择
0-m/h　1-100m/h

2.3 App 下发蓝牙数据到仪表

序号	字节定义	样例	备注
0	包头	0xAB	
1	指令码	0	
2	总字节数	10	帧总字节数
3	数据 1	0	按钮定义
4	数据 2	0	见下面描述
5	数据 3	0	见下面描述
6	数据 4	0	见下面描述
7	数据 5	0	见下面描述
8	CRC 低字节	CRC	
9	CRC 高字节	CRC	

数据 1，按钮定义

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
锁车按钮	公制英制转换按钮	零启动按钮	定速巡航按钮	氛围灯开关	大灯按钮	档位设定按钮 1	档位设定按钮 0
0	0	0	0	0	0	0	0

数据 2，定速巡航最低速度限制，
Unsigned char 格式， 在公制格式下，1 表示 1KM/H, 在英制格式下 1 表示 1MPH
数据 3，节能模式下，最高速度限制，
Unsigned char 格式，在公制格式下，1 表示 1KM/H, 在英制格式下 1 表示 1MPH
数据 4，舒适模式下，最高速度限制，
Unsigned char 格式，在公制格式下，1 表示 1KM/H, 在英制格式下 1 表示 1MPH
数据 5，运动模式下，最高速度限制，
Unsigned char 格式，在公制格式下，1 表示 1KM/H, 在英制格式下 1 表示 1MPH

3 控制器 OTA 升级

3.1 App 下发仪表指令

开始透传:
A5 00 FF 00 00 00 00 5A

结束透传：
A5 FF 00 00 00 00 00 5A

开始拼包：
A5 01 FE 00 00 00 00 5A

结束拼包：
A5 FE 01 00 00 00 00 5A

连接信号：
A5 02 FD 5A
App 发送连接指令之后，仪表开始定时上传数据

3.2 仪表(主机)和控制器(从机)升级通信

1）主机发送连接：01 51 C1 DC 从机应答：01 51 C1 DC

丛机地址	命令码	CRCL	CRCH
01	51	CRCL	CRCH

2）主机发送擦除：01 52 81 DD 从机擦除成功应答：01 52 81 DD 失败应答：01 D2 80 7D

丛机地址	命令码	CRCL	CRCH
01	52	CRCL	CRCH

3）主机发送读设备版本信息：01 07 00 00 00 10 20 06 6F

丛机地址	命令码	地址 H	地址 L	寄存器数 H	寄存器数 L	字节数	CRCL	CRCH
01	07	00	00	00	10	20	CRCL	CRCH

4）主机发送升级包数据 （有效数据长度可变，一般建议 1024）

丛机地址	命令码	包号 H	包号 L	字节数 H	字节数 L	数据内容 1	数据内容 2	数据内容 3
01	50	00	00	字节数 H	字节数 L	XX	XX	XX

数据内容 4	数据内容 5	数据内容 6	数据内容 7	数据内容 8	数据内容 9	数据内容 10
XX	XX	XX	XX	XX	XX	XX

数据内容 11	数据内容 12	数据内容 13	数据内容 14	数据内容 15	数据内容 16	CRCL	CRCH
XX	XX	XX	XX	XX	XX	CRCL	CRCH

4 仪表 OTA 升级

4.1 升级仪表用的 BIN 文件

升级文件为.bin 文件，其中 bin 文件前 16 字节为升级固件信息，16 字节之后为升级数据。

下图为 bin 文件前面部分内容：

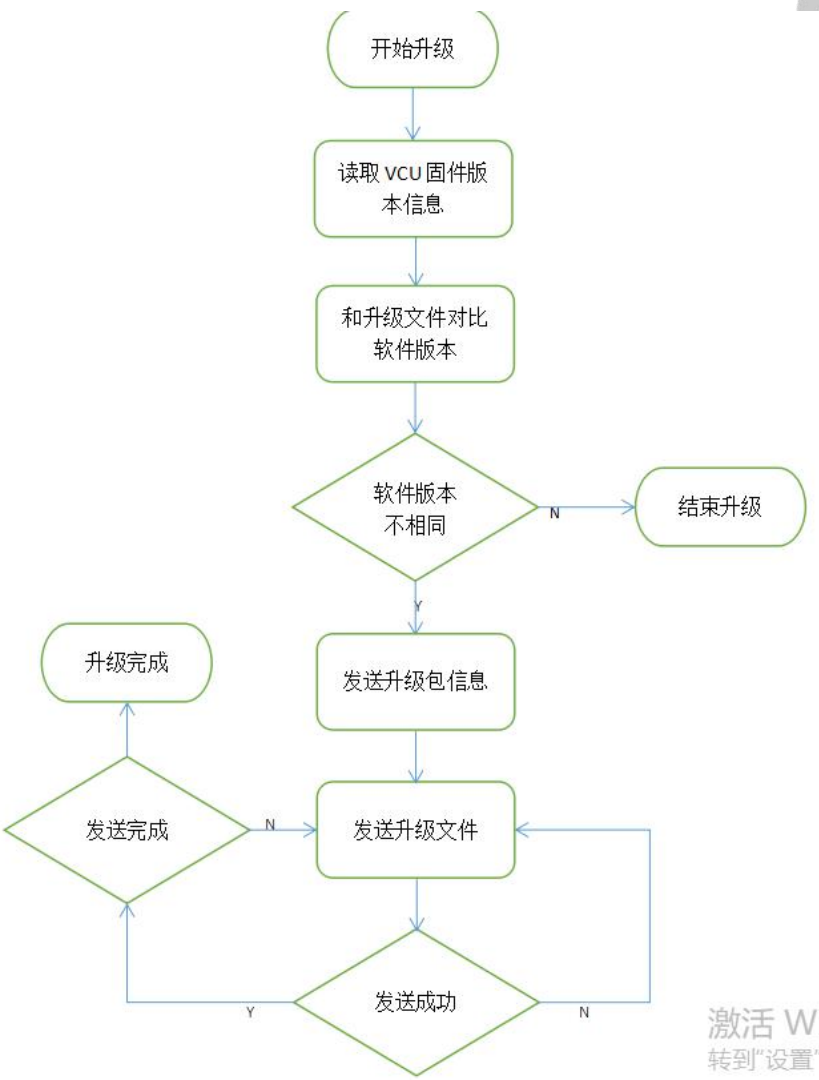
Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	6F	3C	D4	A2	03	01	B4	22	42	42	42	42	FF	FF	12	00
0010	18	F0	9F	E5	18	F0	9F	E5	18	F0	9F	E5	18	F0	9F	E5
0020	18	F0	9F	E5	18	F0	9F	E5	18	F0	9F	E5	18	F0	9F	E5
0030	9D	64	84	E2	01	00	40	E2	01	00	44	E2	01	00	48	E2
0040	01	00	4C	E2	01	00	54	E2	01	00	50	E2	01	00	58	E2

前 16 字节解析如下所示(低字节在前)：

Byte0-Byte3　升级数据校验字节
Byte4-Byte5　待升级 vcu 版本号　　（数据 0x03 0x01 对应版本号 V1.03）
Byte6-Byte7　升级文件长度，单位：字　（数据 0xB4 0x22　对应长度 0x22B4）
Byte8-Byte11　升级类型　　（数据　0x42 0x42 0x42 0x42　升级仪表程序）
Byte12-Byte13　保留
Byte14-Byte15　蓝牙协议栈版本　（数据 0x12 0x00 对应版本号 V0.12）

App 需要检测待升级仪表版本号是否和仪表当前版本号不相同，如果版本号一样仪表不会进行升级。

4.2 仪表升级流程图：



4.3 读取 VCU 固件版本信息

App 发送： UUID：f000ffc1-0451-4000-b000-000000000000

序号	字节定义	样例
0	指令	0x00

VCU 回复： UUID：f000ffc1-0451-4000-b000-000000000000

序号	字节定义	样例
0	VCU 版本号低 8 位	0x1A
1	VCU 版本号高 8 位	0x00
2	固件包长度低字节	0xB4
3	固件包长度高字节	0x22
4	Uid0	0x42
5	Uid1	0x42
6	Uid2	0x42
7	Uid3	0x42
8	BLE 协议栈版本低 8 位	0x12
9	BLE 协议栈版本高 8 位	0x00

App 可以只使用 vcu 版本，其他数据可以不用关注。

4.4 发送升级包信息

发送的内容为升级包的前 16 个字节

App 发送： f000ffc1-0451-4000-b000-000000000000

序号	字节定义	样例
0	固件包校验字节	0X6F
1	固件包校验字节	0x3C
2	固件包校验字节	0xD4
3	固件包校验字节	0xA2
4	VCU 版本号低 8 位	0x03
5	VCU 版本号高 8 位	0x01
6	固件包长度低字节	0xB4
7	固件包长度高字节	0x22
8	Uid0	0x42
9	Uid1	0x42
10	Uid2	0x42
11	Uid3	0x42
12	保留	0xFF
13	保留	0xFF
14	BLE 协议栈版本低 8 位	0x12
15	BLE 协议栈版本高 8 位	0x00

VCU 回复： f000ffc2-0451-4000-b000-000000000000

序号	字节定义	样例
0	包号低 8 位	0x00
1	包号高 8 位	0x00

需注意 VCU 回复是通过 FFC2

4.5 发送升级文件

App 发送： f000ffc2-0451-4000-b000-000000000000

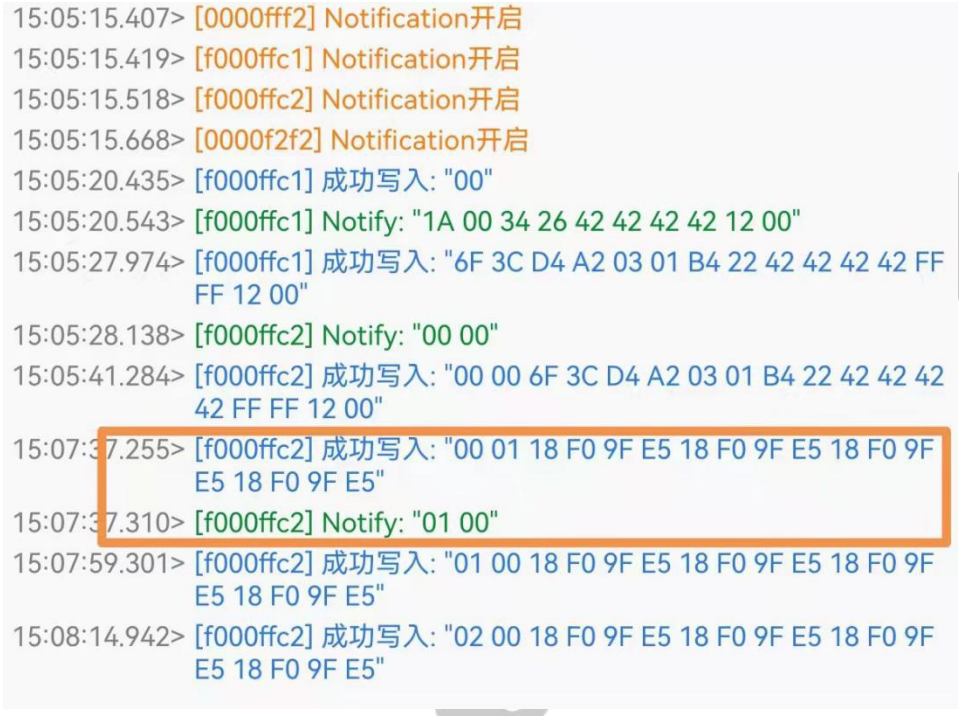
App 需要将整个升级文件拆成每包 16 字节全部发给 VCU，发送完成包号需自动加 1 ， app 可以每包间隔 10ms 发送

序号	字节定义	样例
0	包号低 8 位	0x00
1	包号高 8 位	0x00
2-17	升级数据 16 字节	

Vcu 回复： f000ffc2-0451-4000-b000-000000000000
只有 vcu 接收数据错误才会有回复 回复内容为包号，app 收到回复之后需要从收到的包号位置继续往下发送

序号	字节定义	样例
0	包号低 8 位	0x##
1	包号高 8 位	0x##

4.6 仪表升级示例：



方框内 App 发送格式有误 vcu 会回复包号，提醒 app 需要包号为 0x01 0x00 的数据。

5 AT 指令

5.1 数据通讯连接测试

该指令用于判断蓝牙模块是否正常工作，处理器收到应答则认为设备间通讯正常，蓝牙模块可以正常使用。

指令	应答	参数
AT+	OK+	无

5.2 蓝牙广播名称查询与设置

指令	应答	参数
查询：AT+NAME?	OK+NAME:Para	Para：模块名称 最长允许 11 个字符，包括字母、数字、下划线 默认 Para=HW_ZAxx
设置： AT+NAME[Para]	OK+NAME:Para	

5.3 应用层密码验证

指令	应答	参数
设置：AT+PWD[Para]	验证成功：OK+PWD:Y 验证失败：OK+PWD:N	Para: 密码-6 位数字

说明：

1：该指令用于蓝牙连接时密码验证（服务 uuid 0xF2F0）。

2：当 APP 端需要验证密码时，才需要使用此指令验证身份; 模块与模块连接时，主机会自动进行身份验证。固定密码验证时间为 10s， 即在连接上后 10s 时间内必须输入正确的密码才可以保持连接，否则断开连接。

5.4 连接校验密码设置指令

发送 AT+PWDM[XXXXXX] XXXXXX 为六位数字
回复 0K+PWDM:XXXXXX

6 CRC16-MODBUS

```
/*****************************************************************************
@func       :Calculate CRC16-MODBUS
@poly       :8005(x16+x15+x2+1)
@init       :0xFFFF
@xorout     :0x0000
@refin      :yes
@refout     :yes
*****************************************************************************/
uint8_t CRCH[] =
{
    0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,
    0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,
    0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,
    0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,
    0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,
    0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,
    0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,
    0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,
    0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,
    0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,
    0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,
    0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,
    0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,
    0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,
    0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,
    0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,
    0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,
    0x40
};

uint8_t CRCL[] =
{
    0x00,0xC0,0xC1,0x01,0xC3,0x03,0x02,0xC2,0xC6,0x06,0x07,0xC7,0x05,0xC5,0xC4,
    0x04,0xCC,0x0C,0x0D,0xCD,0x0F,0xCF,0xCE,0x0E,0x0A,0xCA,0xCB,0x0B,0xC9,0x09,
    0x08,0xC8,0xD8,0x18,0x19,0xD9,0x1B,0xDB,0xDA,0x1A,0x1E,0xDE,0xDF,0x1F,0xDD,
    0x1D,0x1C,0xDC,0x14,0xD4,0xD5,0x15,0xD7,0x17,0x16,0xD6,0xD2,0x12,0x13,0xD3,
    0x11,0xD1,0xD0,0x10,0xF0,0x30,0x31,0xF1,0x33,0xF3,0xF2,0x32,0x36,0xF6,0xF7,
    0x37,0xF5,0x35,0x34,0xF4,0x3C,0xFC,0xFD,0x3D,0xFF,0x3F,0x3E,0xFE,0xFA,0x3A,
    0x3B,0xFB,0x39,0xF9,0xF8,0x38,0x28,0xE8,0xE9,0x29,0xEB,0x2B,0x2A,0xEA,0xEE,
    0x2E,0x2F,0xEF,0x2D,0xED,0xEC,0x2C,0xE4,0x24,0x25,0xE5,0x27,0xE7,0xE6,0x26,
    0x22,0xE2,0xE3,0x23,0xE1,0x21,0x20,0xE0,0xA0,0x60,0x61,0xA1,0x63,0xA3,0xA2,
    0x62,0x66,0xA6,0xA7,0x67,0xA5,0x65,0x64,0xA4,0x6C,0xAC,0xAD,0x6D,0xAF,0x6F,
    0x6E,0xAE,0xAA,0x6A,0x6B,0xAB,0x69,0xA9,0xA8,0x68,0x78,0xB8,0xB9,0x79,0xBB,
```

```
    0x7B,0x7A,0xBA,0xBE,0x7E,0x7F,0xBF,0x7D,0xBD,0xBC,0x7C,0xB4,0x74,0x75,0xB5,
    0x77,0xB7,0xB6,0x76,0x72,0xB2,0xB3,0x73,0xB1,0x71,0x70,0xB0,0x50,0x90,0x91,
    0x51,0x93,0x53,0x52,0x92,0x96,0x56,0x57,0x97,0x55,0x95,0x94,0x54,0x9C,0x5C,
    0x5D,0x9D,0x5F,0x9F,0x9E,0x5E,0x5A,0x9A,0x9B,0x5B,0x99,0x59,0x58,0x98,0x88,
    0x48,0x49,0x89,0x4B,0x8B,0x8A,0x4A,0x4E,0x8E,0x8F,0x4F,0x8D,0x4D,0x4C,0x8C,
    0x44,0x84,0x85,0x45,0x87,0x47,0x46,0x86,0x82,0x42,0x43,0x83,0x41,0x81,0x80,
    0x40
};

uint16_t    CalculateCRC16(uint8_t const *msgPtr, u32 msgLen)
{
   uint8_t crcHigh = 0xFF;
   uint8_t crcLow = 0xFF;
   uint8_t index;

   while (msgLen--)
   {
      index = crcLow ^ (*(msgPtr++));
      crcLow = crcHigh ^ CRCH[index];
      crcHigh = CRCL[index];
   }
   return (uint16_t)((uint16_t)(crcHigh<<8) | crcLow);
}
```