
Hobbywing Scooter BLE Protocol B-01.0.01

Hobbywing Scooter BLE Protocol B-01.0.01

Version modification history

version	date	Modify the content	Reviser
01.0.01	2024-12-13	First release	Li Yong

2024-12-
13
directory

1	Bluetooth service	
	UUID.....	3
2	Meter and APP	
	communication.....	3
	2.1 Meter scheduled Reporting Script 0x00 Data	
	Format.....	3
	2.2 Meter Scheduled Reporting Instruction Code 0x01 Data	
	Format.....	5
	2.3 The app sends Bluetooth data to the meter.....	7
3	Controller OTA	
	upgrade.....	7
	3.1 App Issue Instrument Commands.....	7
	3.2 Meter (host) and controller (slave) upgrade communication.....	8
4	Meter OTA	
	upgrade.....	8

Hobbywing Scooter BLE Protocol B-01.0.01

4.1 BIN file for upgrading meters.....	8
4.2 Instrument upgrade flow chart:	9
4.3 Read VCU firmware version information.....	10
4.4 Send upgrade package information.....	11
4.5 Send the upgrade file.....	11
4.6 Example of meter upgrade (without encryption):.....	12
5 AT directive.....	
12	
5.1 Datacom Connectivity Test	12
5.2 Bluetooth Broadcast Name Query and Settings.....	13
5.3 Application layer password verification.....	13
5.4 Connecting to the Verification Password Setting Command.....	13
6 CRC16-	
MODBUS.....	14

1 Bluetooth service UUID

1.1 Transparent data transmission services

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)

Hobbywing Scooter BLE Protocol B-01.0.01

1.2 AT Instruction

AT

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

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

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

1.3 Instrument

Upgrade UUID: F000FFC0-0451-4000-B000-0000000000000000

Upgrade UUID: F000FFC1-0451-4000-B000-0000000000000000

Upgrade UUID: F000FFC2-0451-4000-B000-0000000000000000

2 Meter APP correspondence

2.1 The meter is Script 0x00 data

serial	Byte	Example	Content	Remark
0	turba	0xAB		
1	Script	0x00		
2	Total number	0x19		0x19=25
3	Motor	0x01	Take 0~1	1-positive 0-negative
4	Gear	0x01	Take 0~3	1-Indicates 2-First
5	Elect	0x64	Take 0-100	0x64=100 Indicate 100%
6	veloc1 High	0x27	Take 0-65535	0x2710=10000 Indicate 1 The 10km/h or 10mph (Refer to the Table bit6 Note)
7	veloc1 Low	0x10		
8	veloc2 High	0x27	Take 0-65535	0x2710=10000 Indicate 2 The 10km/h or 10mph (Refer to the Table bit6 Note)
9	veloc2 Low	0x10		
10	Voltage high	0x00		0x0064=100 denote 10V, service

AT reception

AT send

Hobbywing Scooter BLE Protocol B-01.0.01

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			

Bit1 5	Bit1 4	Bit1 3	Bit1 2	Bit1 1	Bit1 0	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	bit 0
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

Controller status registers:

Bit2, 1: Indicates that the headlights are on

Bit1, bit0, 0: energy saving, 1: normal, 2: movement

Hobbywing Scooter BLE Protocol B-01.0.01

2.2 Instrument scheduled reporting instruction code 0x01 data format

serial number	Byte definition	Example	Content definition	remark
0	turban	0xAB		
1	Script	0x01		
2	Total number of bytes in the frame	0x19		0x19=25
3	Entering cruise control allows the lowest speed	0x03		In the metric format, 3 is indicated 3KM/H, in imperial format 3 means 3MPH
4	Maximum speed limit in energy saving mode	0x0f		In metric format, 15 represents 15KM/H, in imperial system 15 under the format is indicated 15MPH
5	Comfort mode maximum speed limit	0x16		In metric format, 22 represents 22KM/H, in imperial system 22 is indicated

Hobbywing Scooter BLE Protocol B-01.0.01

				under the format 22MPH
6	Sport mode maximum speed limit	0x1f		In metric format, 31 represents 31KM/H, in imperial system 31 under the format is indicated 31MPH
7	Not defined	0x00		
8	Fault flag high byte	0x00		
9	Fault flag low byte	0x00		
10	panel selects High Byte	0x00		
11	panel selects low bytes	0x00		
12	Not defined	0x00		
13	Not defined	0x00		
14	Not defined	0x00		
15	Not defined	0x00		
16	Not defined	0x00		
17	Not defined	0x00		

Hobbywing Scooter BLE Protocol B-01.0.01

18	Instrument software version number	0x80		Indicates 8025_01.00.01
19	Instrument software version number	0x25		
20	Instrument software version number	0x01		
21	Instrument software version number	0x00		
22	Instrument software version number	0x01		
23	CRC16 low bytes			MODBUS—CRC16
24	CRC16 High Bytes			

Fault flags:

Bit15	Bit14	Bit13	Bit12	Bit11	Hobbywing	Scooter	BiBLE	Protocol	B-01	H-01	Bit 2	Bit 1	bit 0
Fault warning enabled bit				The throttle is not reset to F2	Brake fin ger pad not res et F1	0p am p bi as E9	Hal 1 Fau lt E7		Overcurrent E4	Communication disconne ct E3	Throt tle fault E2	Brake fail ure E1	
1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit 0
0	0	0	0	0	RG	MP	N								
1	0	0	0	0	B M S	B fa bo	ya fa bo								

0--Hide the corresponding
 1--Load the corresponding

Bit12: Speed unit
 1-100m/h

Panel Selection:

Hobbywing Scooter BLE Protocol B-01.0.01

0-m/h

2.3 The app sends Bluetooth data to the meter

serial number	Byte definition	Example	remark
0	turban	0xAB	
1	Script	0	
2	Total bytes	10	Total number of bytes in the frame
3	Data 1	0	button definition
4	Data 2	0	See description below
5	Data 3	0	See description below
6	Data 4	0	See description below
7	Data 5	0	See description below
8	CRC Low Bytes	CRC	
9	CRC High Bytes	CRC	

Hobbywing Scooter BLE Protocol B-01.0.01

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	bit0
Lock	Metric Convert	Zero start bu	Cruise butto	Ambient switc	Headlig bu	Gear butto 1	Gear de butto 0
0	0	0	0	0	0	0	0

data 2, cruise control minimum speed

Unsigned char format In the metric 1 denote 1KM/H, In imperial 1 denote 1MPH

data 3, in energy-saving mode, maximum

Unsigned char format, in the metric 1 denote 1KM/H, In imperial 1 denote 1MPH

data 4, in comfort mode, maximum speed

Unsigned char format, in the metric 1 denote 1KM/H, In imperial 1 denote 1MPH

data 5, in sport mode, maximum speed limit,

Unsigned char format, in the metric 1 denote 1KM/H, In imperial 1 denote 1MPH

3 control FATHupgra

3.1 App Issue

Start :

A5 00 FF 00 00 00 00 5A

Data

1, button definition end transmission:

A5 FF 00 00 00 00 00 5A

Start consolidating:

A5 01 FE 00 00 00 00

5A End the package:

A5 FE 01 00 00 00 00 5A

Connection signal:

A5 02 FD 5A

After the app sends the connection command, the meter starts uploading data regularly

Hobbywing Scooter BLE Protocol B-01.0.01

3.2 Instrument (host) and controller (slave) upgrade communication

51 CRCL CRCH
CRCL CRCH

2) Host sends : 01 52 81 DD Slave erase 01 52 81 DD Failed 01 D2 80 7D
CRCL CRCH
52 CRCL CRCH

3) host Read the device 01 07 00 00 00 10 20 06 6F

Cluster	command	addrH	addrL	Number of H	Number of L	Number	CRCL	CRCH
	07	00	00	00	10	20	CRCL	CRCH

4) hosts send upgdata (The effective data General 1024)

Cluster	command	panu H	panu L	bytenu H	bytenu L	Withinap 1	Withinap 2	Withinap 3
	50	00	00	Number H	Number L	XX	XX	XX

Data	4	Data	5	Data	6	Data	7	Data	8	Data	9	Data	10
		XX											

Withinap 11	Withinap 12	Withinap 13	Withinap 14	Withinap 15	Withinap 16	CRCL
	XX		XX		XX	CRCL

4 appear upgra

1) Host send connection: 01 51 C1 DC Slave answer: 01 51 C1 DC

Cluster address command code

01

Cluster address command code

01

01

Hobbywing Scooter BLE Protocol B-01.0.01

01

XX

CRCH

XX CRCH

4.1 Upgrade the BIN file for the instrument

The upgrade file is a .bin file, where the first 16 bytes of the bin file are the upgrade firmware information, and the first 16 bytes are the upgrade data.

The following image shows the first part of the bin file

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												
0020	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

Hobbywing Scooter BLE Protocol B-01.0.01

for 16 The byte parse 16 bytes are

Byte0-Byte3 Upgrade the data

Byte4-Byte5 To be vcu Version (Data.) 0x03 0x01 Corresponding V1.03)

Byte6-Byte7 Upgrade file length, un(Data.) 0xB4 0x22 Corresponding 0x22B4)

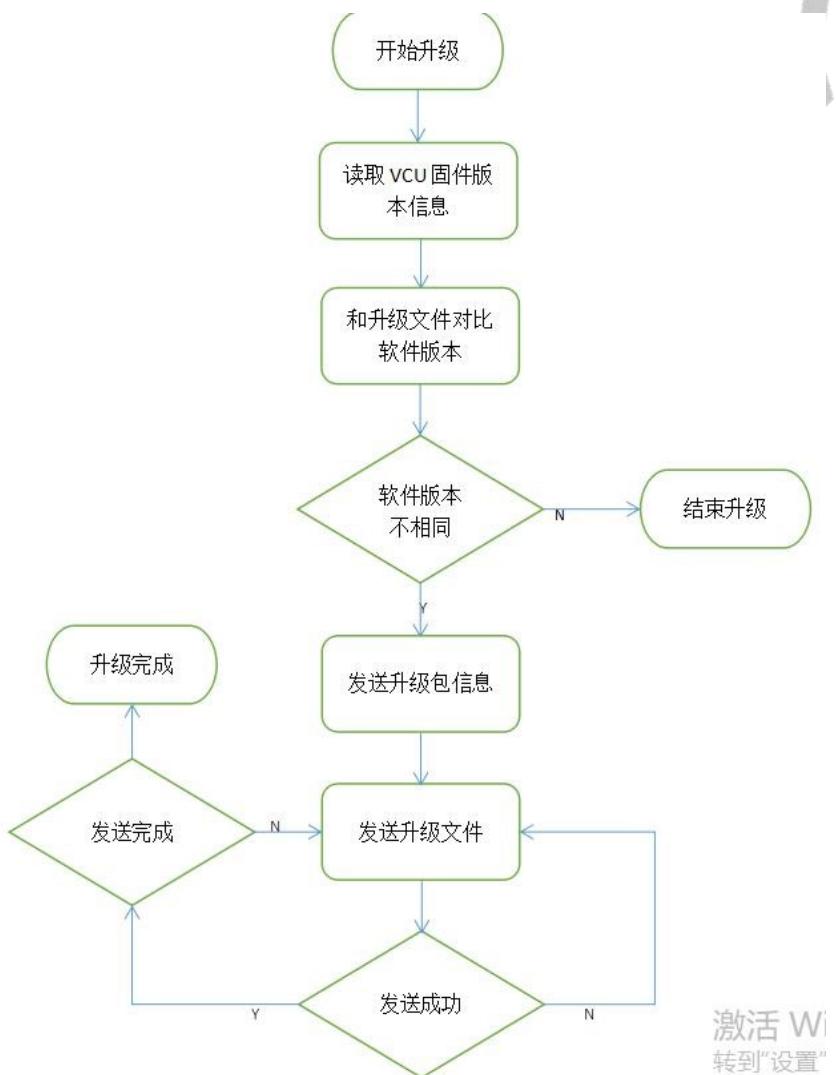
Byte8-Byte11 Upgrade (Data.) 0x42 0x42 0x42 0x42 Upgrade Instrument

Byte12-Byte13 retain

Byte14-Byte15 Bluetooth proto(Data.) 0x12 0x00 Corresponding V0.12)

App You need to check whether the version number of the instrument number is the same, the

4.2 Instrument upgrade



4.3 Read VCU firmware version information

App 发送: UUID:f000ffcc-0451-4000-b000-000000000000

serial number	Byte definition	Example
0	directives	0x00

Hobbywing Scooter BLE Protocol B-01.0.01

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

serial	Byte	Example
0	VCU The 8 bi	0x1A
1	VCU The 8 bi	0x00
2	The firmware package	0xB4
3	The length of the	0x22
4	Uid0	0x42
5	Uid1	0x42
6	Uid2	0x42
7	Uid3	0x42
8	WASThe protocol 8 bi	0x12
9	WASThe protocol 8 bi	0x00

App Can be used/cu version, other data can be ignored.



4.4 Sending Upgrade Package Information

The content sent is the first 16 bytes of the upgrade package

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

Hobbywing Scooter BLE Protocol B-01.0.01

serial	Byte	Example
0	Firmware package	0X6F
1	Firmware package	0x3C
2	Firmware package	0xD4
3	Firmware package	0xA2
4	VCU The 8 bi	0x03
5	VCU The 8 bi	0x01
6	The firmware package	0xB4
7	The length of the	0x22
8	Uid0	0x42
9	Uid1	0x42
10	Uid2	0x42
11	Uid3	0x42
12	retai	0xFF
13	retai	0xFF
14	WASThe protocol 8 bi	0x12
15	WASThe protocol 8 bi	0x00

F000FFC2-0451-4000-B000-0000000000000000

serial	Byte	Example
0	The 8 bi	0x00
1	The 8 bi	0x00

Note VCU The reply is FFC2

4.5 Send the

VCU

replied:

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

The app needs to split the entire upgrade file into 16 bytes per package and send it to VCU, and the package number needs to be automatically added to 1, and the app can be sent at intervals of 10ms per package

serial number	Byte definition	Example
0	The package number is 8 digits lower	0x00
1	The bag number is 8 digits higher	0x00

Hobbywing Scooter BLE Protocol B-01.0.01

2-17	Upgrade data 16 bytes	
------	-----------------------	--

VCU reply: f000ffc2-0451-4000-b000-0000000000000000

Only the VCU will reply if the data is received incorrectly. The reply content is the package number, and the app needs to continue sending it down from the received package number after receiving the reply.

serial number	Byte definition	Example
0	The package number is 8 digits lower	0x##
1	The bag number is 8 digits higher	0x##

4.6 Example of Meter Upgrade:

```
15:05:15.407> [0000fff2] Notification开启
15:05:15.419> [f000ffc1] Notification开启
15:05:15.518> [f000ffc2] Notification开启
15:05:15.668> [0000f2f2] Notification开启
15:05:20.435> [f000ffc1] 成功写入: "00"
15:05:20.543> [f000ffc1] Notify: "1A 00 34 26 42 42 42 42 12 00"
15:05:27.974> [f000ffc1] 成功写入: "6F 3C D4 A2 03 01 B4 22 42 42 42 42 FF
FF 12 00"
15:05:28.138> [f000ffc2] Notify: "00 00"
15:05:41.284> [f000ffc2] 成功写入: "00 00 6F 3C D4 A2 03 01 B4 22 42 42 42
42 FF FF 12 00"
15:07:37.255> [f000ffc2] 成功写入: "00 01 18 F0 9F E5 18 F0 9F E5 18 F0 9F
E5 18 F0 9F E5"
15:07:37.310> [f000ffc2] Notify: "01 00"
15:07:59.301> [f000ffc2] 成功写入: "01 00 18 F0 9F E5 18 F0 9F E5 18 F0 9F
E5 18 F0 9F E5"
15:08:14.942> [f000ffc2] 成功写入: "02 00 18 F0 9F E5 18 F0 9F E5 18 F0 9F
E5 18 F0 9F E5"
```

Inside App The sending vcu The package number app The package 0x01 0x00 data.



5.1 Datacom connectivity

Instructions

AT

Hobbywing Scooter BLE Protocol B-01.0.01

This instruction is used to judge whether the Bluetooth module is working properly, and the processor receives the response and considers that the communication between devices is normal and the Bluetooth module can be used normally.

directives	Response	parameter
AT+	OK+	not

5.2 Bluetooth Broadcast Name Query and Setting

5.3 Application-layer

direc	Respo	param
Set up AT+PWD[Para]	VerificatiOK+PWD:Y VerificatiOK+PWD:N	For: passw6 Digits

1: This instruction is used for password (0xF2F0) .

2: When the password is required, the module will verify the fixed password 10s, That is, after 10s You must enter the correct password within the time limit.

5.4 Connect the Verification

Send AT+PWDM[XXXXXX] XXXXXX It is a six-digit password.
Reply OK+PWDM:XXXXXX

directives	Response	parameter
查询: AT+NAME?	OK+NAME:Para	Para: Module name Up to 11 characters are allowed, pack
Set up: AT+NAME[For]	OK+NAME:Para	括字母、数字、下划线默 认 Para=HW_ZAxx

Illustrate:

Hobbywing Scooter BLE Protocol B-01.0.01

Otherwise disconnect.

Hobbywing Scooter BLE Protocol B-01.0.01

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,  
    0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41,0x00,0xC1,0x81,0x40,0x01,0xC0,0x80,0x41  
};  
  
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,
```

Hobbywing Scooter BLE Protocol B-01.0.01

```
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,0x9F,0x5E,0x5A,0x5A,0x9A,0x9B,0x5B,0x99,0x59,0x58,0x98,0x98,0x88,  
0x48,0x49,0x89,0x4B,0x8B,0x8A,0x4A,0x4E,0x4E,0x4F,0x8F,0x8E,0x8D,0x4D,0x8D,0x4C,0x8C, and  
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);  
}
```

