

QH- xxx 4B 测速型车检器串行数据输出协议
V1.0C
上海智合电子科技有限公司

一、适用的通讯接口

本协议仅适用 RS232/RS485/RS422 通讯方式。

二、适用的车检器

本协议仅适用上海智合电子科技有限公司的测速型车检器系列 QH-xxx4B, 包含 QH-KCS4B、QH-KEP4B、QH-HCS4B、QH-HEP4B 等等。

三、协议的内容

3.1 数据通讯的帧结构

串行通讯协议的帧结构包含以下几个部分：1) 1 字节帧头 (0xFF); 2) 1 字节地址 (车检器的地址); 3) 2 字节数据或 34 字节数据; 4) 1 字节帧校验。

3.1.1 2 字节数据的广播

两字节数据块用于 12 比特的数据传输。

其中，两个字节数据的高 4 位，用于区分车道以及数据类型；低 12 位用于表示速度数据或车长数据。速度数据的单位是 1 公里/小时，车长数据的单位是 0.1 米。

高 4 位数据定义

0x0: 车道 1 的进入速度

0x1: 车道 2 的进入速度

0x2: 车道 1 的长度

0x3: 车道 2 的长度

0x4: 车道 1 的离开速度

0x5: 车道 2 的离开速度

0x6: 逆向车道 1 的离开速度

0x7: 逆向车道 2 的离开速度

0x8: 逆向车道 1 的进入速度

0x9: 逆向车道 2 的进入速度

0xA: 逆向车道 1 的长度

0xB: 逆向车道 2 的长度

0xC: 线圈状态广播 (12bit 数据的高 4 位固定为 0xA, 即第一字节为 0xCA)

0xD: 保留, 用于以后的数据扩展

0xE: 保留, 用于以后的数据扩展

0xF: 保留给多字节数据块传输的识别 (低 4 位为 0~7), 用于防止数据帧解码时出现混淆。

校验码是数据帧中除帧头和检验字节以外其他所有字节累加和的低 8 位, 即
校验码 = (地址字节+数据高字节+数据低字节) & 0xFF

因此, 速度的表示范围为 0~4095 公里/小时; 长度的表示范围为 0~409.5 米。

术语解释:

逆向车道,指的是车辆由车道设定的线圈 2 进入线圈 1,在一个规定了行使方向的车道上,出现逆向车道的速度和长度,表示有车在该车道上逆向行驶。

例如:

FF 01 00 21 22 表示 0x01 车检器的第一车道的速度为 33 公里/小时

FF 01 20 11 32 表示 0x01 车检器的第一车道的长度为 1.7 米

FF 01 10 1C 2D 表示 0x01 车检器的第二车道的速度为 28 公里/小时

FF 01 30 10 41 表示 0x01 车检器的第二车道的长度为 1.6 米

当数据块的第一节为 0xCA 时,为环路线圈状态发生改变的广播,其后还有 1 个字节的狀態数据,状态数据表示的是线圈状态。

线圈状态的编码规则:

D7	D6	D5	D4	D3	D2	D1	D0
L4 故障	L3 故障	L2 故障	L1 故障	L4 状态	L3 状态	L2 状态	L1 状态
0(正常)	0(正常)	0(正常)	0(正常)	0(无车)	0(无车)	0(无车)	0(无车)
1(故障)	1(故障)	1(故障)	1(故障)	1(有车)	1(有车)	1(有车)	1(有车)

上位机可以根据状态变化的广播建立重建车辆进入以及离开各线圈的时刻,并获得车辆通过每个线圈所用的时间。这样就可以在上位机上计算车辆的速度、车的长度等而不受方向设置的限制,并由此计算出时间占有率、空间占有率、流量等等交通统计参数。

(注意: 使用状态广播的车检器应当把内部的系统工作模式设置为 0x45)

3.1.2 34 字节数据的广播

三十四字节的数据块仅用于交通流量数据的传输,特征是最先收到的 2 个字节为固定为 0xF0 0xC0。真正的数据块为 32 字节,按照接收的顺序,各字段的含义如下:

偏移地址	字节数	含义
0	2	固定为 0xF0 0xC0
2	2	第一车道统计时间段内的车辆通过总数,单位为辆, int 型数据
4	2	第二车道统计时间段内的车辆通过总数,单位为辆, int 型数据
6	4	第一车道统计时间段内的所有车辆的通过时间总和,单位为毫秒, long 型数据
10	4	第二车道统计时间段内的所有车辆的通过时间总和,单位为毫秒, long 型数据
14	2	第一车道统计时间段内的所有车辆的车长总和,单位为分米, int 型数据
16	2	第二车道统计时间段内的所有车辆的车长总和,单位为分米, int 型数据
18	4	第一车道统计时间段内的所有车辆的通过速度总和,单位为公里/小时, long 型数据

22	4	第二车道统计时间段内的所有车辆的通过速度总和，单位为公里/小时，long 型数据
26	2	第一车道统计时间段内的平均速度，单位为公里/小时，int 型数据
28	2	第二车道统计时间段内的平均速度，单位为公里/小时，int 型数据
30	2	第一车道统计时间段内的时间占有率，单位为万分之一，int 型数据
32	2	第二车道统计时间段内的时间占有率，单位为万分之一，int 型数据

3.2 车检器命令及命令响应的帧结构

命令通讯协议的帧结构包含以下几个部分：1) 1 字节帧头 (0xAA)；2) 1 字节帧标志 (0x24)；3) 1 字节地址 (车检器的地址)；4) 1 字节的命令及参数长度压缩编码；5) 0 字节~7 字节命令参数；6) 1 字节的校验和。

帧头、帧标志和校验字节不参与校验的计算，校验码为其余字节累加和的低 8 位。

命令及参数长度压缩编码的编码规则为，低 3 位表示命令参数的长度，高 5 位表示命令，其中如果最高位是 1 的话，表示的是从车检器返回的命令响应帧。

3.3 串口通讯的硬件设置

波特率：115200；数据位：8；校验：无；流控制：无；停止位：1

3.4 车检器的设置与远程操作

对车检器进行远程操作，可能需要观察车检器的响应，这时需要事先暂停车检器的数据输出功能。操作完成后，可以用恢复命令恢复车检器的数据输出功能。如果忘记恢复数据输出，车检器会自动在暂停时间到达后自动恢复输出。

如果不需车检器回馈响应消息，则可以不暂停数据输出就可以对车检器进行操作。

使用数据暂停命令（命令编码 0x51,命令的参数为 0x00）可以暂时停止车检器的数据输出。

AA 24 (地址) 51 00 (51+地址+0x00)

使用数据恢复命令（命令编码 0x51,命令的参数为 0x01）可以恢复车检器的数据输出。

AA 24 (地址) 51 01 (51+地址+0x01)

例如，暂停地址为 0x01 的车检器

发送：AA 24 01 51 00 52

再例如，恢复地址为 0x01 的车检器的串口开始输出数据

发送：AA 24 01 51 01 53

3.4.1 系统远程复位

系统远程复位，使用 0x40 命令，不需要命令参数。

格式为：AA 24 (原地址) 40 (40+原地址)

例如，远程重启地址为 0x01 的车检器

发送： AA 24 **01** 40 **41**

测检器返回：AA 24 01 C0 C1

3.4.2 车检器地址的设定

系统出厂默认的地址是 **0x01**。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (原地址) 51 00 (**51+原地址**)

接下来，发送 AA 24 (原地址) 14 01 00 10 新地址 (**0x25+原地址+新地址**)

设置成功，车检器会返回所设置的地址。

例如，设置车检器的地址为 0x01

发送： AA 24 **01** 14 01 00 10 **01 27**

车检器返回：AA 24 01 8C 01 00 10 **01 9F**

再例如，设置车检器的地址为 **0x02**

发送： AA 24 **01** 14 01 00 10 **02 28**

车检器返回：AA 24 01 8C 01 00 10 **02 A0**

注：地址修改后，系统需要复位重启后，车检器才使用新的地址与上位机通讯。

3.4.3 地址码丢失条件下的车检器地址的设定

如果在管理不善时丢失了车检器的地址，可以使用一种特殊的广播方法设置网络上的车检器的地址，这种设置方法不采用地址码匹配，而是采用序列号匹配和类型匹配。用这种方法可以远程动态设置网络上任何一个车检器的地址而不担心地址冲突。

改命令使用的地址为广播地址 0xFF，命令代码为 0x4F。该命令有 7 个字节的参数，分别是：序列号的第 1 字节、序列号的第 2 字节、序列号的第 3 字节、序列号的第 4 字节、分类码的第 1 字节、分类码的第 2 字节以及地址码

发送 AA 24 FF 4F SN1 SN2 SN3 SN4 S1 S2 (地址) (**4E+ SN1+SN2+SN3+SN4+S1+S2+地址**)

设置成功，车检器会用新的车检器地址响应，返回的命令代码为 0xC8。

AA 24 (地址) C8 (C8+地址)

例如，将序列码 **B9650771** 以及型号编码为 **HEP4B** 的车检器的地址设置为 **0x02**

发送： AA 24 **FF 4F B9 65 07 71 48 45 02 73**

车检器返回：AA 24 02 C8 CA

3.4.4 线圈 1 到线圈 2 的距离设定（第 1 车道）

系统出厂默认线圈 1 到线圈 2 的距离是 1 米。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 14 01 00 14 (距离) (0x29+地址+距离)

例如，线圈 1 到线圈 2 的距离 10,即 1 米

发送： AA 24 01 14 01 00 14 0A 34

车检器返回： AA 24 01 8C 01 00 14 0A AC

再例如，线圈 1 到线圈 2 的距离 20,即 2 米

发送： AA 24 01 14 01 00 14 14 3E

车检器返回： AA 24 01 8C 01 00 14 14 B6

3.4.5 线圈 3 到线圈 4 的距离设定（第 2 车道）

系统出厂默认线圈 3 到线圈 4 的距离是 1 米。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 14 01 00 15 (距离) (0x2A+地址+距离)

例如，线圈 3 到线圈 4 的距离 10,即 1 米

发送： AA 24 01 14 01 00 15 0A 35

车检器返回： AA 24 01 8C 01 00 15 0A AD

再例如，线圈 1 到线圈 2 的距离 20,即 2 米

发送： AA 24 01 14 01 00 15 14 3F

车检器返回： AA 24 01 8C 01 00 15 14 B7

3.4.6 速度限制的设定

系统出厂默认的限制速度是 1 公里/小时。

当所测得的车速小于限制速度时，系统不输出。最大为 255，即 0xFF。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 14 01 00 16 (限制速度) (0x2B+地址+限制速度)

例如，限制速度为 1 公里/小时

发送： AA 24 01 14 01 00 16 01 2D

车检器返回： AA 24 01 8C 01 00 16 01 A5

3.4.7 运行模式的设定

系统的运行模式有以下三种：

- 1) 普通模式，代码为 0x05，有线圈状态变化的广播数据输出；
- 2) 双向速度测量模式，代码为 0x45，在普通模式基础上，叠加车辆速度和车辆长度的测量及广播输出；

- 3) 交通流量统计模式，代码为 0xC5，在双向速度测量模式基础上，叠加交通流量的统计及广播输出。

系统出厂默认的运行模式是普通模式。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 61 (模式) (61+地址+模式)

设置成功后，车检器返回代码为 0xE1，数据为所设定模式的数据。

AA 24 (地址) E1 (模式) (E1+地址+模式)

例如，设置为普通模式

发送： AA 24 01 61 05 67

车检器返回： AA 24 01 E1 05 E7

再例如，设置为双向速度测量模式

发送： AA 24 01 61 45 A7

车检器返回： AA 24 01 E1 45 27

再例如，设置为交通流量统计模式

发送： AA 24 01 61 C5 27

车检器返回： AA 24 01 E1 C5 A7

3.4.8 交通流量统计间隔时间的设定

统计间隔时间可以设置为 5 秒到 3600 秒，（建议不要低于 5 秒）。

系统出厂默认为 120 秒。

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送

AA 24 (地址) 15 02 00 18 (高字节) (低字节) (0x2F+地址+高字节+低字节)

例如，设置统计间隔时间为 1 分钟（60 秒）

发送： AA 24 01 15 02 00 18 00 3C 6C

车检器返回： AA 24 01 8D 02 00 18 00 3C E4

再例如，设置统计间隔时间为 2 分钟（120 秒）

发送： AA 24 01 15 02 00 18 00 78 A8

车检器返回： AA 24 01 8D 02 00 18 00 78 20

再例如，设置统计间隔时间为 5 分钟（300 秒）

发送： AA 24 01 15 02 00 18 01 2C 5D

车检器返回： AA 24 01 8D 02 00 18 01 2C D5

3.4.9 系统时间的设置

对于具备交通流量统计功能的车检器，主机（上位机）应当定期校准车检器内部的实时时钟。

设置车检器的时间参数，用的指令及数据长度压缩代码为 0x27，命令的参数有 7 个字节，分别是年、月、日、时、分、秒、星期，数据类型均为 unsigned char。

年的格式为压缩的格式，只能设置为 00 年~255 年，分别表示 2000 年~2255 年；

星期的格式为：0 表示星期天，1~6 分别表示星期一~星期六；

利用串口调试助手设置，接收发送都使用 16 进制。

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 27 (年) (月) (日) (时) (分) (秒) (星期) (27+地址+年+月+日+时+分+秒+星期)

所有字段的数据都是 8 位二进制数据，数据类型为 unsigned char

设置成功后，车检器会立即返回所设置的时间参数，其命令及数据长度压缩代码为 0x9F，格式为：

AA 24 (地址) 9F (年) (月) (日) (时) (分) (秒) (星期) (9F+地址+年+月+日+时+分+秒+星期)

所有字段的数据都是 8 位二进制数据，数据类型为 unsigned char

例如：设置 01 地址的车检器之系统时间为 2010 年 09 月 20 日 07 点 32 分 00 秒星期一

发送： AA 24 01 27 0A 09 14 07 20 00 01 77

车检器返回： AA 24 01 9F 0A 09 14 07 20 00 01 F7

3.4.10 读取系统时间

使用 0x19 命令和 0x00 命令参数，可以获得车检器的实时时间数据。

车检器返回的响应数据帧，命令及数据长度压缩代码为 0x9F，格式为：

AA 24 (地址) 9F (年) (月) (日) (时) (分) (秒) (星期) (9F+地址+年+月+日+时+分+秒+星期)

所有字段的数据都是 8 位二进制数据，数据类型为 unsigned char

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 19 00 (19+地址)

例如：读取 01 地址车检器的系统时间

发送： AA 24 01 19 00 1A

车检器返回： AA 24 01 9F 0A 04 02 12 2D 1B 05 0F

表示为 2010 年 4 月 2 日 18 时 45 分 27 秒星期五

3.4.11 初始化 RTC 实时时钟

当车检器电池耗尽更换后，时间可能不正常，需要使用 RTC 初始化指令来启动 RTC 的正常工作。

使用 0x19 命令和 0x04 命令参数，可以初始化 RTC 实时时钟。

车检器返回的响应数据帧，命令及数据长度压缩代码为 0x9F，格式为：

AA 24 (地址) 9F 0A 08 02 11 14 00 01 (D9+地址)

车检器默认的初始化时间为 2010 年 8 月 2 日 17 时 20 分 00 秒星期一。

所有字段的数据都是 8 位二进制数据，数据类型为 unsigned char

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 19 04 (1D+地址)

例如：读取 01 地址车检器的系统时间

发送： AA 24 01 19 04 1E

车检器返回： AA 24 01 9F 0A 08 02 11 14 00 01 DA

这是车检器默认的初始化时间。

3.4.12 获得车检器序列号

使用 0x30 命令，可以获得车检器序列号。车检器的序列号对应与外壳上的序列码，是产品管理的一个部分，在产品保修以及产品升级服务时都需要。

车检器的序列号为 4 个字节，数据类型为 unsigned long，在 0xB4 返回命令编码中带回序列号数据。

AA 24 (地址) B4 SN1 SN2 SN3 SN4 (B4+地址+SN1+ SN2 +SN3+ SN4)

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 30 (30+地址)

例如，读取 0x01 地址的车检器之序列号

发送： AA 24 01 30 31

车检器返回： AA 24 01 B4 B9 65 07 71 4B

3.4.13 获得车检器型号

使用 0x38 命令，可以获得车检器序型号。车检器的型号对应与外壳上的型号选择，是产品管理的一个部分，在产品保修以及产品升级服务时都需要。

车检器的型号为 4 个字节，前 3 个字节数据类型为 char，最后 1 个字节数据类型为 unsigned char，在 0xBC 返回命令编码中带回型号数据。

AA 24 (地址) BC S1 S2 S3 TYPE (BC+地址+S1+ S2 +S3+ TYPE)

首先，发送 AA 24 (地址) 51 00 (51+地址)

接下来，发送 AA 24 (地址) 38 (38+地址)

例如，读取 0x01 地址的车检器之序列号

发送： AA 24 01 38 39

车检器返回： AA 24 01 BC 48 45 50 4B E5

表示的型号: **HEP 4B**

3.4.14 获得 CPU 序列号

使用 0x30 命令, 可以获得 CPU 列号。车检器的 CPU 序列号是产品管理的一个部分, 在产品保修以及产品升级服务时都需要。

车检器的 CPU 序列号为 7 个字节, 数据类型为 unsigned char CPUID[7], 在 0x9F 返回命令编码中带回 CPU 序列号数据。

AA 24 (地址) 9F ID1 ID2 ID3 ID4 ID5 ID6 ID7 (9F+地址+ID1+ID2+ID3+ID4+ID5+ID6+ID7)

首先, 发送 **AA 24 (地址) 51 00 (51+地址)**
接下来, 发送 **AA 24 (地址) 19 03 (1C+地址)**

例如, 读取 0x01 地址的车检器之 CPU 序列号

发送: **AA 24 01 19 03 1D**

车检器返回: **AA 24 01 9F 00 03 00 DE 00 0A 54 DF**

3.4.15 利用 USB 存储交通流量数据的设定

客户可以选择使用 USB 存储器来存储交通流量数据 (不是默认的功能)。

取消的代码为 0x00, 打开的代码为 0x02。

利用串口调试助手设置, 接收发送都使用 16 进制。

首先, 发送 **AA 24 (地址) 51 00 (51+地址)**

接下来, 发送 **AA 24 (地址) 14 01 00 17 (代码) (2C+地址+ 代码)**

例如, 设置 01 地址的车检器使用 U 盘存储交通流量数据

发送: **AA 24 01 14 01 00 17 02 2F**

车检器返回: **AA 24 01 8C 01 00 17 02 A7**

再例如, 取消 01 地址的车检器使用 U 盘存储交通流量数据

发送: **AA 24 01 14 01 00 17 00 2D**

车检器返回: **AA 24 01 8C 01 00 17 00 A5**

四、协议起草人

彭树林

五、版权声明

本协议之版权属于上海智合电子科技有限公司。