|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 目标地址 | 功能码 | 数据长度 | 数据信息 | | 校验码 | |
| 1个字节 | 1个字节 | 1个字节 | 2个字节 | | 2个字节 | |
| 01 | 06 | 02 | 00 | 00 | B8 | 88 |
| 当Arduino发生错误，无法工作的情况下发送 | | | | | | |
| 01 | 06 | 02 | 00 | 01 | 79 | 48 |
| 当Arduino准备就绪，通知PLC。只发送一次。 | | | | | | |
| 01 | 06 | 02 | 00 | 02 | 39 | 49 |
| 应提示操作员“日期错误”， | | | | | | |
| 01 | 06 | 02 | 00 | 03 | F8 | 89 |
| 应提示操作员“数据错误，请重新写卡” | | | | | | |
| 01 | 06 | 02 | 00 | 04 | B9 | 4B |
| 应提示操作员“未知错误” | | | | | | |
| 01 | 06 | 02 | 00 | 05 | 78 | 8B |
| 应提示操作员“车辆卡错误” | | | | | | |
| 01 | 06 | 02 | 00 | 06 | 38 | 8A |
| 应提示操作员“人员卡错误” | | | | | | |
| 01 | 06 | 02 | 00 | 07 | F9 | 4A |
| 应提示操作员“数据上传成功”。 | | | | | | |
| 01 | 06 | 02 | 00 | 08 |  |  |
| 任务错误 | | | | | | |
| ~~地址~~ | ~~功能码~~ | ~~数据长度~~ | ~~车牌号~~ | ~~卡号~~ | ~~车载容量~~ | ~~浓度~~ |
| ~~1个字节~~ | ~~1个字节~~ | ~~1个字节~~ | ~~5个字节~~ | ~~字节长度=卡位数~~ | ~~2个字节~~ | ~~2个字节~~ |
| ~~01~~ | ~~10~~ | ~~00~~ | ~~00~~ | ~~00~~ | ~~00~~ | ~~00~~ |
| ~~车载类型~~ | ~~校验~~ |  |  |  |  |  |
| ~~1个字节~~ | ~~2个字节~~ |  |  |  |  |  |
| ~~00~~ | ~~00~~ |  |  |  |  |  |
| ~~后台的数据信息~~ | | | | | | |

Arduino向PLC发送的数据信息

PLC 向Arduino发送的数据信息

以卡号111111为例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 目标地址 | 功能码 | 数据长度 | 卡号 | 溶液剩余量 | 完成情况 | 校验 |
| 1个字节 | 1个字节 | 9个字节 | 6个字节 | 2个字节 | 1个字节 | 2个字节 |
| 01 | 03 | 9 | 01 | 00 | 00 | 00 |
| 系统完成所有操作发送完成信息 | | | | | | |

ModBus 通信协议的 CRC ( 冗余循环校验码含2个字节, 即 16 位二进制数。CRC 码由发送设备计算, 放置于所发送信息帧的尾部。接收信息设备再重新计算所接收信息 (除 CRC 之外的部分）的 CRC, 比较计算得到的 CRC 是否与接收到CRC相符, 如果两者不相符, 则认为数据出错。

1) 预置 1 个 16 位的寄存器为十六进制FFFF(即全为 1) , 称此寄存器为 CRC寄存器。  
2) 把第一个 8 位二进制数据 (通信信息帧的第一个字节) 与 16 位的 CRC寄存器的低 8 位相异或, 把结果放于 CRC寄存器。  
3) 把 CRC 寄存器的内容右移一位( 朝低位)用 0 填补最高位, 并检查右移后的移出位。  
4) 如果移出位为 0, 重复第 3 步 ( 再次右移一位); 如果移出位为 1, CRC 寄存器与多项式A001 ( 1010 0000 0000 0001) 进行异或。  
5) 重复步骤 3 和步骤 4, 直到右移 8 次,这样整个8位数据全部进行了处理。  
6) 重复步骤 2 到步骤 5, 进行通信信息帧下一个字节的处理。  
7) 将该通信信息帧所有字节按上述步骤计算完成后,得到的16位CRC寄存器的高、低字节进行交换。  
8) 最后得到的 CRC寄存器内容即为 CRC码。

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 目标地址 | 功能吗 | 数据长度 | 任务编号 | 司机代码 | 车牌号 | 体积 | 计划量 | 完成量 | 浓度 | 车载类型 | 校验码 |
| 1字节 | 1字节 | 1字节 | 4字节 | 4字节 | 5字节 | 2字节 | 2字节 | 2字节 | 2字节 | 1字节 | 2字节 |
| 0x01 | 0x08 | 0x16 | 0x00 0x00 0x00 0x00 | 0x00 0x00 0x00 0x00 | 0x00 0x00 0x00 0x00 0x00 | 0x00 0x00 | 0x00 0x00 | 0x00 0x00 | 0x00 0x00 | 0x00 | 0x00 0x00 |

01 10 16 0000001E 30303031 3832373035 0008 000A 0000 0002 02 FCC4

30 0001 82705 8 10 0 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 目标地址 | 功能码 | 数据长度 | 任务编号 | 加注量 | 校验 |
| 1个字节 | 1个字节 | 1个字节 | 4个字节 | 2个字节 | 2个字节 |
| 0x01 | 0x03 | 0x06 | 0x00 0x00 0x00 0x00 | 0x00 0x00 | 0x00 0x00 |

0103060000001E004B0144

发送给PLC的数据中，车辆的体积已经x10。

PLC发给Arduino的数据中，加注量也应该x10。