设计上的考虑点:

减小执行sh文件的名称长度, 方便在控制台运行

针对Rika的设备: 增加EOT的回答处理逻辑,终止当前通信.

发送任意命令: 用户可选消息末端增加CRC,BCC计算, 以方便不同协议的消息构建

为方便后续开发和维护, 把封装在类中的消息工具函数(消息检查,CRC,BCC等)转移到main部分,并适配CUnetTool的数据类型,以方便CUnetTool的后续开发.

HDIW设备: 考虑到寄存器地址从0000-000B, 因此将用户输入的条件从十进制改为十六进制.

在最初的详细设计阶段，我计划直接使用rs485common中的senddata和recvdata来完成收发功能，并按设计实现了代码。  
但是在液面计测试时，发现recvdata必须传入一个明确的接收长度，而在实际通信中，接收长度并不总是已知，因此无法满足实际需求。  
经过分析项目的特点，最终选择了更底层的scc2698 API，使用scc2698read实现不限制长度的接收功能。虽然问题解决了，但由于一开始没有意识到接口限制，在排查错误和修改关联的代码的过程中浪费了时间。

在开发过程中，发现第一次发送时，设备能正常回应；但第二次发送时却出现异常，并影响后续的接收。  
通过排查发现，液面计在接收完消息后，必须要回复一次EOT作为结束标志，如果没有回送EOT就直接再次发送，会导致接收异常。  
这是由于我在最初实现时，将通信简单地理解为“一发一收”，没有考虑到设备要求的轮询(polling)和结束信号机制。