版本：V1

总体要求：

1. 实现设备连接
2. 实现数据发送，与接收显示。
3. 开发语言要求，使用VisualStudio，使用C或C++开发封发和接收发送。并使用C#开发模拟收发和显示。或全部使用C#，但C#代码保护性不好。。。C#做界面显示方便

具体功能要求

1. 从节点模拟
   1. 模拟一个网络节点，IP和Mac地址可配置
   2. 模拟数据通过该节点发送。UDP，数据可以配置。
2. 主节点
   1. 实现从节点检索，显示连接节点数目，ip定制或相关其他信息。多个从节点。
   2. 实现接收节点数据信息，并显示
   3. 实现发送数据到节点
3. 数据解析
   1. UDP包数据结构

|  |  |  |
| --- | --- | --- |
| PackLen | 包有效数据长度 | Byte0 |
| Byte1 |
| ~PackLen | 长度取反校验 | Byte2 |
| Byte3 |
| DataType | 此包的数据类型 | Byte4 |
| LiveCounter | 此类数据的循环+1计数 | Byte5 |
|
| DataStructSum | 数据类型计数 | 对应的每包统计数 |
| Databuf | 数据内容,字节 | Byte6..... |

* 1. 数据类型和数据结构体定义

数据的结构体定义如下

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据类型1=0xCA | 说明 | 时间戳 | 编号 | 方向 | 信息位结构体 | ID | DLC | Data |
| bytes数量 | 4 | 1 | 1 | 1 | 4 | 1 | 0~64 |
| 数据类型 | Ulong | 1,2 | 0=TX,1=RX | IdType:2bit, DataType:2, Type:2 | u32 | （0~64） | 0~64个bytes |

Data的有效长度有DLC决定，一般情况结构体都使用固定64个 bytes。

一个数据占据76个字节。那么一个UDP包，除了包头长度类型等，可以包含至少13个数据。

（不同的数据类型，占据的字节总数不一样。一个UDP包含的数据的个数不一样。本次仅以0xCA开发）

* 1. 数据的封发

数据的发送与接收以数据结构体为单位，所以分两个函数部分进行数据分发与解析。

要求函数封装接口参考如下：

数据封装：

数据n

UDP

数据2

数据3

数据1

提供封装泛型函数，如PackData( byte dataType, T-Structure data,byte? Index)。

调用此函数，则将数据函数封装到内存【或UDP包中】。

数据拆分：

数据n

数据1

UDP

提供泛型函数，如GetData( byte dataType, T-Structure[] data)。

调用则读取对应类型结构体数据，返回得到的长度。如果没有新的数据则返回0；

泛型函数，目的实现不同类型数据的封发。

* 1. UDP数据的发送与接收
     1. UDP的发送需要讨论以下两种方式更优
        1. 函数调用发送
        2. 后台定时自动封装发送
     2. UDP接收与存储
        1. 收到UDP数据，自动？解析到结构体内存给封发函数调用