本文主要阐述原棉软件中，C++层和C#层的通信协议。

由于动态版本来自静态版本，因此主要是讲解动态版本的相关通信协议。

基础协议：

基础通信由16个字节构成，由客户端向服务端传输一个16个字节的指令，再由服务端向客户端回复一个16个字节的指令。

这16个字节常被拆为2部分，第一部分是前4个字节，为4个char型的指令，后一部分为后12个字节，为3个unsigned int型的整数或6个unsigned short型的整数。

下面用

<XXXX>表示unsigned int

[XX]表示unsigned short

（具体寄存器使用方法参见“命令表\_2017年1月12日.xls”）

【1】FPGA寄存器读取指令

客户端发送：

GETX <address> <0> <0>

服务器回复：

RETX <address> <[register](http://www.baidu.com/link?url=XY6VWp_dd8uJWjuCp44eCqfJSDjKI4xwXMhmeyE9hJnMzR8a3dNH4rCk03xccJACl7fJ0ZlfTzejOdc0Vzd4bKKp-j-eJwqwTUo3dRwicNe" \t "https://www.baidu.com/_blank)> <0>

若回复的address为0，则认为寄存器读取失败。

【2】FPGA寄存器写入指令

客户端发送：

SETX <address> <[register](http://www.baidu.com/link?url=XY6VWp_dd8uJWjuCp44eCqfJSDjKI4xwXMhmeyE9hJnMzR8a3dNH4rCk03xccJACl7fJ0ZlfTzejOdc0Vzd4bKKp-j-eJwqwTUo3dRwicNe" \t "https://www.baidu.com/_blank)> <0>

服务器回复：

RETX <address> <[register](http://www.baidu.com/link?url=XY6VWp_dd8uJWjuCp44eCqfJSDjKI4xwXMhmeyE9hJnMzR8a3dNH4rCk03xccJACl7fJ0ZlfTzejOdc0Vzd4bKKp-j-eJwqwTUo3dRwicNe" \t "https://www.baidu.com/_blank)> <0>

若回复的address为0，则认为寄存器写入失败。

【3】读取数据指令（静态版）

客户端发送：

RAWX <0> <0> <0>

服务器回复：

RETX [data1.1][data1.2] [data2.1][data2.2] [data3.1][data3.2]

【4】读取数据指令（动态版）

客户端发送：

DATX<0><0><0>

服务器回复：

RETL <LEN><0><0>

LEN\*[data]

这是一个超过基本协议的长度回复，因为要求根据len的变化来调整采样长度，因此做成了不定长的形式，先在基本协议中回复一个LEN，接下来再发送长达LEN这么多的ushort。（注意接收的时候若按字节接收，则是2\*LEN个字节）

【5】各类设置指令

下面的指令依次对称，且回复均为：

RETX <value><0><0>

设置/获取底层平均次数：

AVGX <value><0><0>

GAVX <0><0><0>

设置/获取底层判断阈值：

THRX <value><0><0>

GTHX <0><0><0>

设置/获取底层比较方式：

CMPX <value><0><0>

GCMX <0><0><0>

设置/获取底层数据长度：

LENX <value><0><0>

GLEX <0><0><0>