

## 目录

准备工作.....	1
实现原料.....	1
怎样测试通讯库.....	2
ZS 可以接口那些底层通讯 IO .....	2

## 准备工作

首先，你需要学会用一个通讯库，比如，CIS,ICS,Tsocket,IPWork 等等，定位好你的通讯库的用法，并且确保不会犯下对机制理解有偏差这种错误。

接下来，你要了解你用的这套通讯库的基本工作模型是异步还是阻塞，这里是个实现接口的思路分水岭

## 实现原料

制作 ZS 底层通讯库 IO 模版都在 Source\developerRefrence 目录中

开发基于阻塞通讯库主要参考 ZS 的 Indy IO，这套 IO，都是在一个循环中，反复检查是有收到数据然后 Fill 它，是否有待发数据然后发送它，那些 API 命名都很规范。看一眼就会明白。

开发基于 ZS 异步通讯库 IO 的参考就多了，有 ICS，CrossSocket，DIOCP，异步通讯库的特征都是事件化的，你需要至少捕获一个 OnReceive 事件，在这个事件中，实现数据 Fill，如果你接口的通讯库很奇葩，类似 Synapse 没有 OnReceive 事件，就开个 Thread 去反复侦测。发送环节，处理碎片缓冲区和处理队列，你可以根据机理去参考已有的通讯库 IO 源码。

# 怎样测试通讯库

使用 `CommunicationTest.pas` 即可完全测试通讯库，具体用法，用 `Find in File` 搜索字符串去找。

比较好的代码测试模版是 6 万压测的源码，你可以把它复制出来作为测试工具来帮你编写实现代码。

做服务器的 IO，要打开内存泄漏检查，一边接口，一边检查内存泄漏。

做客户端的 IO，不需要关心内存泄漏，所有的客户端设计都是用完抛弃的处理方式。

一般来说，写通讯库 IO，不会花上 2 天，也不用反复调试，只要机制理解正确，一般都是一次性合格，只会出些手误的 bug，机制理解正确就没问题。

## ZS 可以接口那些底层通讯 IO

所有的阻塞和异步通讯库，ZS 都可以接口

ZS 可以接口蓝牙，可以接口串口，可以接口 USB，你只要了解它的通讯机理，你就能轻松接口到 ZS 中来自己使用

有空给开源项目写个通讯库 IO，贡献的同时，自己也提高，两全其美。

By qq 600585

2018-1-29