# 上位机框架

## TCP/IP 客户端

* + Unity3D编写。
  + 发送和读取的报文（建议封装，格式统一），结构体
  + 发送结构体：

1. 车模式标志位 bool
2. 车X速度 double
3. 车Y速度 double
4. 机械臂模式标志位 bool
5. 机械臂末端X速度 double
6. 机械臂末端Y速度 double
7. 机械臂末端Z速度 double
8. 一般模式标志位 bool
9. 指令号 int
10. 轴索引 int
11. 速度 double
12. 位置 double

* 接收结构体：

1. 车轮使能状态[4] bool
2. 摆臂使能状态[4] bool
3. 机械臂使能状态[6] bool
4. 车轮位置[4] double
5. 摆臂位置[4] double
6. 机械臂位置[6] double

* 程序开启后，自动连接。运行过程中断开连接有提示，并且自动重连。自动重连提示显示5秒尝试一次，尝试10次，还未连接上，提示连接失败，可手动再次选择重连。
* U3D虚拟模型关节通过接收的状态对齐，还有独立的14个轴的状态显示框。
* 用户界面重新设计，要求简洁但功能齐全。
* 提供一些端口可以让用户自定义功能，比如一些常用位置的设定。

## TCP/IP 服务端

* 服务端放置在倍福控制器中，连带了TCP转ADS功能
* 接收到客户端的指令后，发送给ADS，并且再额外加读取轴状态函数，读取完发给客户端，这为一个循环
* 设置了3秒延时即认为断线功能，断线后先发暂停指令，延时30秒，发复位指令
* 最关键的一点：

1. 现在的模式，是机器人向外发WIFI，IP地址为机器人自身的本地固定IP地址。只有别人连接机器人，而且想控制另外一台机器人，则需要切换WIFI，无法实现总控。
2. 如果想控制多台机器人，则需要机器人都连入同一个网络。
3. 按上述方案，机器人的IP地址可能是不定的。
4. 我们可以用UDP广播的形式，向控制台广播自己的IP地址，然后去用TCP连接。

## 控制箱改版

* 只采用一个拇指按钮，控制车。
* 机械臂末端控制用按钮代替，上下、左右、前后，夹紧松开，共需8个按钮，形状如UR的操作界面。
* 8个机械臂控制按钮还不够，还需在界面上提供机械臂精细操作的虚拟按钮，也参照UR。
* 发现用户无法理解复位功能键，把复位取消，做到启动上，对应着关闭。复位是清错功能，一般不出现。
* 所以控制箱的布局要重新考虑。

## 室外导航的一些想法

* 地图信息获取不清楚是不是我做。
* 地图信息转化为机器人控制很简单，采用指令延时发送、更换，实现机器人直行、转弯等一系列动作。速度人为给定，路程信息由地图给定，延时=路程/速度。
* 还可以在执行命令的时候，申请一个栈，把命令写入这个栈，处理后可以实现机器人原路返回。
* 拐弯全是原地旋转，因为差速转弯的打滑，难以建模，原地旋转准确度较高。
* 难点：先没有考虑障碍物，地图信息没考虑，靳宇栋在负责这块吗？