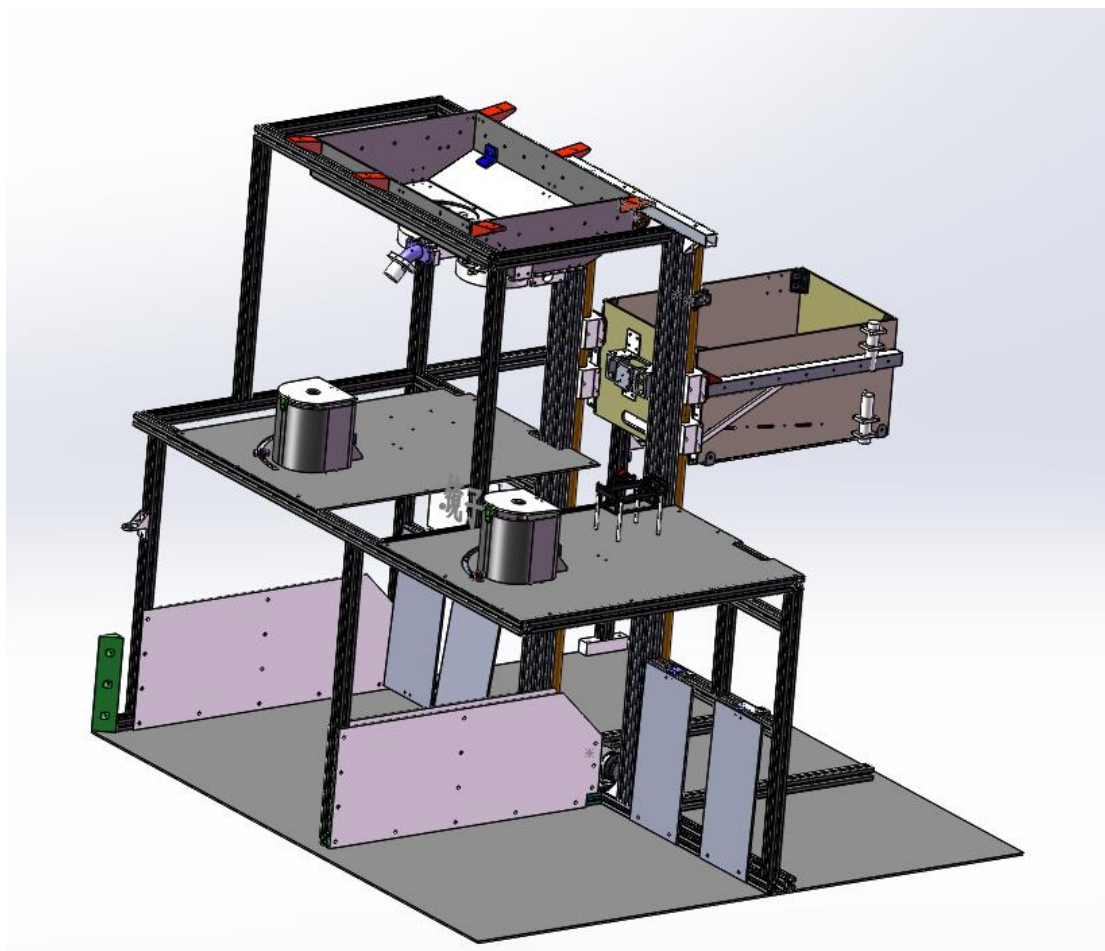


补给站技术文档



目录

一、	机械部分	3
1	概述	3
.1.	背景	3
.2.	设计思路	4
.2.1.	对步兵机器人的方案分析	4
2	结构介绍	6
2.2.	计数问题	6
2.3.	小弹仓的设计	7
3.	大弹丸的漏弹设计	8
3.	改进的方向	9
4.	维修指南	10
	Wifi 组网部分	10
1.1	数据协议	10
1.2	组网方案	12
1.3	路由器及模块设置	15
1.3.1	路由器设置	15
1.3.2	网转串模块设置	16
	一组嵌入式部分	16
1.	整体方案	16
1.1.	元件清单	16
1.2.	具体的外设功能内容为下表：	17
1.3.	控制电路图：	17
1.4.	软件架构设计和控制框图	18
2.	补给机器人开发细节	19
2.1.	方案的优化过程	19
2.2.	与工程机器人和英雄机器人交互流程	19
2.3.	17mm 子弹计数算法	20
3.	总结	21
3.1	预期效果与最终效果	21

3.2	方案创新点.....	21
3.3	存在的坑	21
3.4	可以改进的地方	21
二组嵌入式部分.....		23
1.	整体.....	23
1.1.	元件清单	23
1.2.	硬件、软件资源配置	23
1.3.	控制电路布线结构，拓扑图.....	24
1.4.	软件架构、控制框图	24
2.	细节.....	24
2.1.	方案的优化过程.....	24
3.	总结.....	25
3.1.	预期效果与最终效果	25
3.2.	方案的创新点.....	26
3.3.	存在的坑	26
3.4.	可以改进的地方	26
4.	附录.....	26

一、机械部分

1 概述

.1. 背景

robomaster 从第一期夏令营到现在的竞赛培训部三期，并伴随全国大学生机器人大赛的不断发展，机器人种类从最初的步兵机器人发

展到现在的步兵车、英雄车、空中机器人、工程机器人、自动基地、补给机器人等多元素机器人相组合。其中补给机器人作为关键以及最基本功能的兵种，起着不可替代的作用。补给机器人一方面可以接收来自官方补给的小弹丸，另一方面还可以接收工程机器人在赛场上取来的大小弹丸，并对步兵机器人和英雄机器人进行补给。由于赛场规则的变化，经过我们的慎重考虑，补给站不再像往常的一样，步兵是可以预约子弹，并且在英雄机器人在外厮杀的时间工程可以从资源岛获取的弹丸先保存在补给站上，等待英雄回来加血或者战术撤退时，进行补给大弹丸。

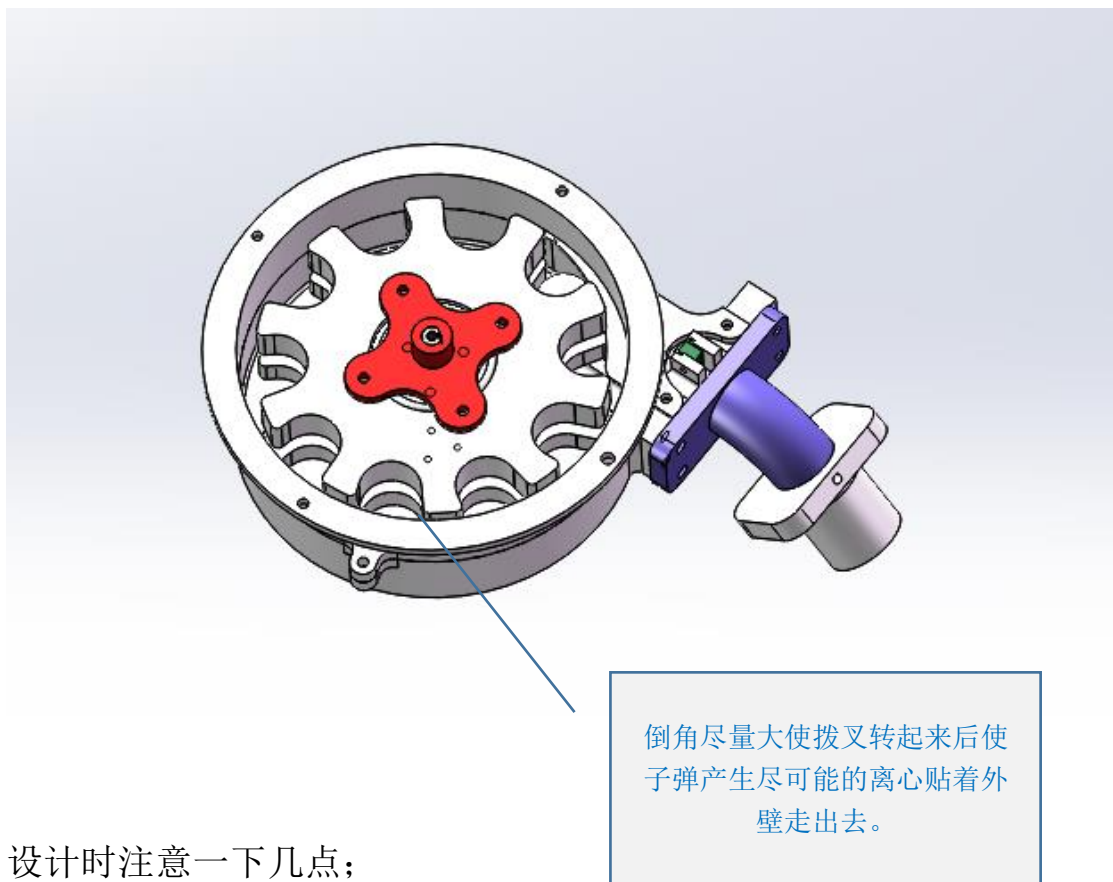
.2. 设计思路

根据上文所述，可以先从功能考虑，

.2.1. 对步兵机器人的方案分析

对步兵机器人有预约的功能，比如步兵机器人在战斗中没有了子弹或者还有少量子弹，但是需要撤退，步兵机器人就可以根据所需要的弹量进行补给，这时补给站需要精准的数弹功能，然后再从方案选择着手，根据现有方案拨弹轮是可以做到的，称重模块也是可以精准技术，但是不能满足赛场上大家不一样的子弹数，然后就是固定的容器，也不是很靠谱。所以最终选择了用拨轮进行计数的方案。

下面是拨轮的设计图：



设计时注意以下几点；

拨叉的设计：

- (1) 拨叉倒角，尽量大使子弹产生离心，会在出口处很大程度避免卡弹。
- (2) 拨叉与拨轮之间的间隙不能太大，不能刚刚好，但是也不能太大，两个子弹上下并排不能错位太多，要不然在转的时候上下子弹之间也会产生比较大的摩擦力，在出口位置很容易卡死。子弹直径 17，拨叉圆直径 18.5 比较合适。
- (3) 拨叉与拨轮上下的间距也要把握好，因为高度太低拨叉在高速转动的情况下可能会让子弹有向上的力从而子弹向上跳，卡在出口。如果太高的话，力向下，摩擦力会很大，对电机、结构都不好。

2 结构介绍

2.1. 拨轮的设计

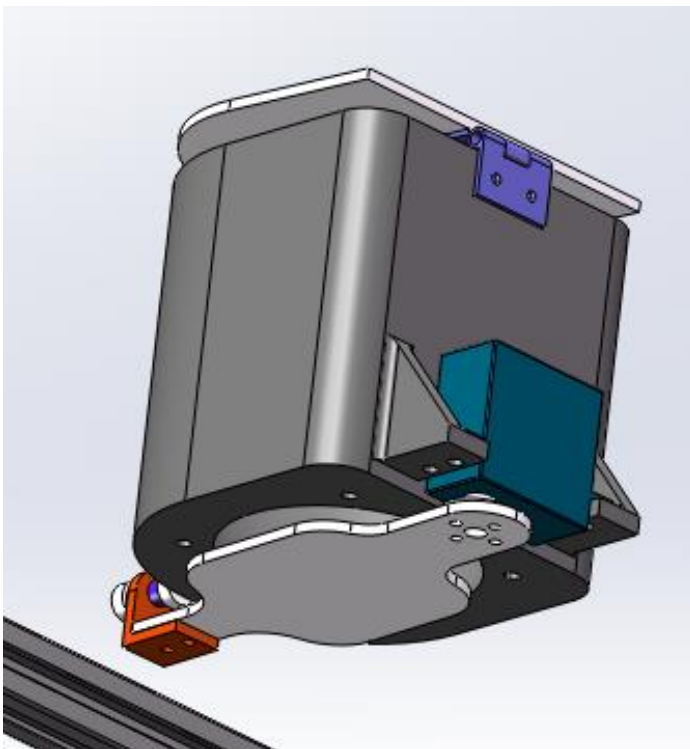
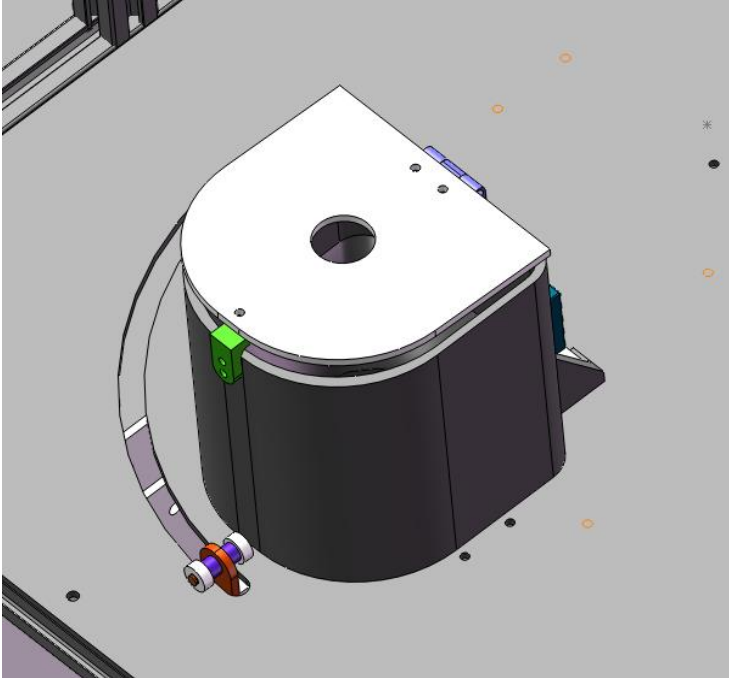
- (1) 出口的设计不要有虚位，就是设计不要有让子弹能卡在出口的空间，出口内壁对边可以倒角大一点，这样可以在卡子弹后反转子弹可以转回来，但是不能侧壁不能太低，我在设计中给的 7mm，一般在高速转动中不会跳出去，而且跳出去可能会卡在出弹口。
- (2) 出口位置尽量不要太大，刚好 18mm 的出弹口，比较好，避免子弹下上跑会卡在出弹口上沿，为了避免出现这种情况，所以在中间会加一个挡片，但是加上挡片后子弹会卡在拨轮、拨叉、挡片中间，这也是通常的卡弹位置，我的解决方法使加了皮筋和毛刷，毛刷的量需要把握，柔性的可以在子弹卡住后迅速弹开，卡弹基本解决，在挡片位置加轴承也是比较好的方法，也是官方比较认可的方法。

2.2. 计数问题

- (1) 计数问题首先采用的微型触碰开关，把位置调整好既能计数还可以把子弹限制住，防止往外流，
- (2) 为了防止触碰开关计数不准在后边还加了一个红外对管的位置，但是子弹很快的话可能发生粘连，所以加了一个弯管，弯管角度越大分离效果越好，但是角度越大越可能会卡弹，可以加轴

承解决，但是为了降低成本，把角度放小就可以。

2.3. 小弹仓的设计

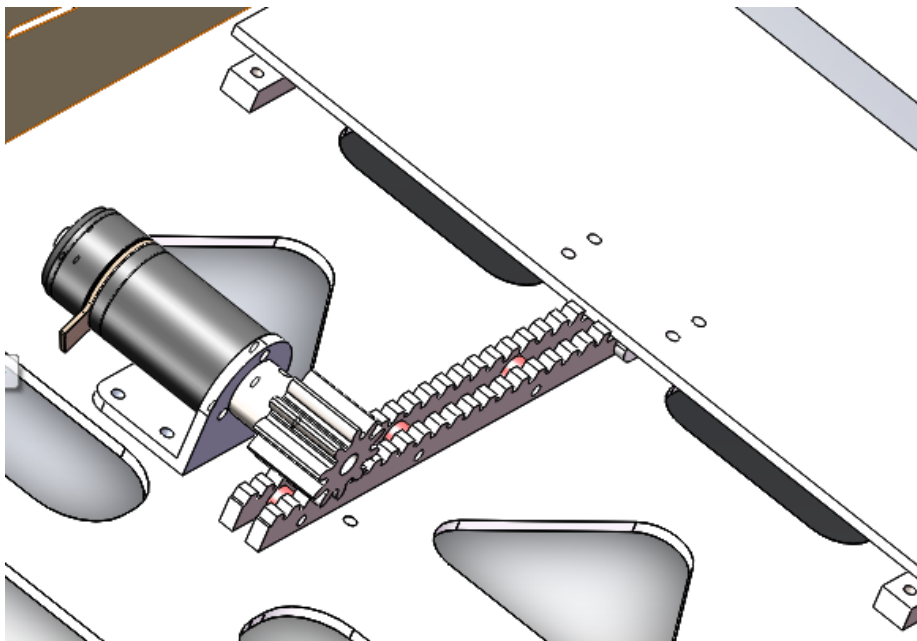


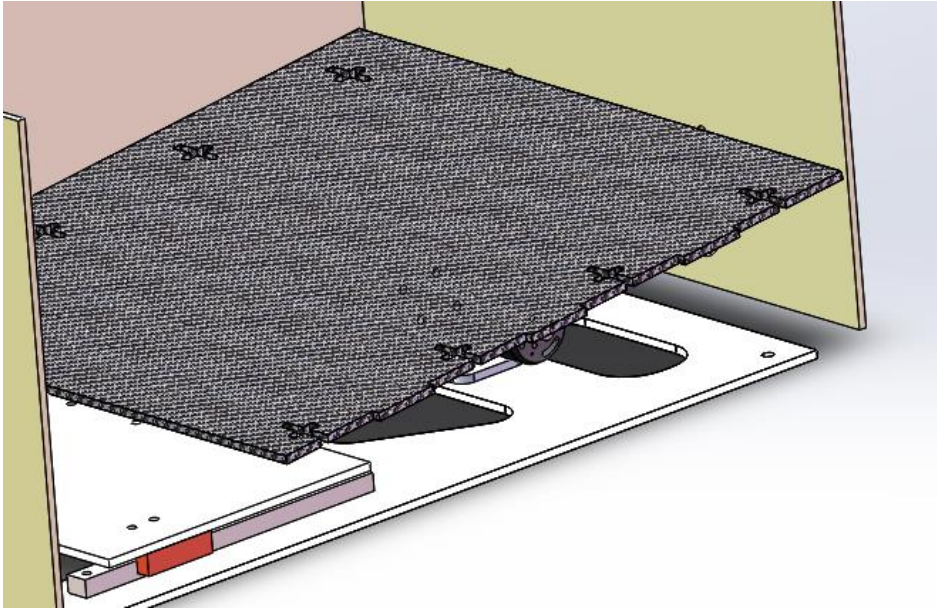
(1) 采用最普通的舵机式开合方式，但是防止子弹太

多会把挡板压弯，所以避免悬臂设计。在另一端加了轴承滑动

- (2) 弹仓的设计，在最初考虑让出弹口小一点这样流进步兵弹仓的子弹会集中一点基本上不会洒出来，所以也设计了锥形内壁，但是这是一个失败的设计，没考虑子弹卡弹，子弹多了很多子弹会在锥形斜坡上卡住，漏弹口可以同时又 4 个多子弹并排下落但是还会出现卡弹情况，所以锥形不适合做漏弹仓。

3. 大 弹 丸 的 漏 弹 设 计





- (1) 采用齿轮齿条设计，稳定并且简单容易维护。
齿条中间通过轴承减小摩擦并且定位。
- (2) 漏弹仓上加斜坡可以把所有的大子弹集中在漏弹口，漏弹口要大于 85mm，也就是 2 个大子弹可以同时下落。

3. 改进的方向

- (1) 拨弹轮一直存在一个问题就是流弹速度没有以往的快，如果预约后立刻去子弹可能并不能达到想要的子弹。
可以有个最低预约每次子弹默认最少 50 颗在小弹仓里准备，接到指令后再进行判断，可以极大的减少加弹时间。
- (2) 小弹仓容量太少，可以适当的加大小弹仓的容量，
最低可容纳 250 颗子弹，这样战术调整可能会更灵活一

点。

- (3) 小弹仓可以加传感器检测弹仓是否有子弹，要不然每次开关电都要检查弹仓是否有弹。

4. 维修指南

(1) 拨弹轮最容易出问题的是卡弹，可能有时候会有异物进去，这样会在出弹口卡住，也许可能也会在管道里卡住，震中情况需要把管拆下来，进行清理。

(2) 齿轮齿条需要每次检查一下，是否在断电前复位，不复位可能会超行程，出现打齿问题，齿条是亚克力板做的可能会折断。

Wifi 组网部分

1.1 数据协议

```
typedef enum
```

```
{
```

```
    Infantry_1    = 0x01,
```

```
    Hero_2        = 0x02,
```

```
    Engineer_3    = 0x03,
```

```
    Infantry_4    = 0x04,
```

```
    Infantry_5    = 0x05,
```

```
    Guard_6       = 0x06,
```

```
    Supply        = 0xff,
```

```
}robot_id_e;
```

每个机器人对应的 ID

```
typedef enum
```

```
{
```

```
    ROBOT_HEARTBEAT    = 0xff,
```

```
    /* Infantry */
```

```

OPEN_THE_GATE      = 0x01,
BOOK_BULLET        = 0x02,

```

```

/* Hero */

```

```

OPEN_THE_BIG_MAG    = 0x10,
LIFT_THE_BIG_MAG     = 0x11,

```

```

/* Engineer */

```

```

ENGINEER_GIVE_COMPLETE = 0x21,
GET_DOWN_THE_BIG_MAG   = 0x22,

```

```

}robot_cmd_e;

```

机器人发送给补给站的命令（数据包类型）

```

typedef enum

```

```

{

```

```

    SUPPLY_HEARTBEAT    = 0xff,
    BOOK_RESULT          = 0x02,

```

```

    BIG_MAG_POSITION     = 0x10,
    BIG_MAG_GATE_STATE   = 0x11,

```

```

}supply_return_cmd_e;

```

补给站发送给机器人的命令（数据包类型）

```

typedef __packed struct

```

```

{

```

```

    uint8_t sof;          //0xA4

```

```

    uint8_t pack_cnt;

```

```

    uint8_t robot_id;      //robot_id_e

```

```

    uint8_t target_id;

```

```

    uint8_t cmd;           //robot_cmd_e

```

机器人发出的命令

```

    __packed union

```

```

    {

```

```

        int32_t i32;

```

```

        float f32;

```

```

        uint8_t u8[4];

```

```

        int8_t i8[4];

```

```

        int16_t i16[2];

```

```

    }data;

```

```

    uint8_t eof;          //0xF0

```

```

}wifi_supply_rec_pack_t;

```

补给站接收到的数据包↑

```

typedef __packed struct

```

```

{
    uint8_t sof;           //0xA4
    uint8_t pack_cnt;
    uint8_t robot_id;      //robot_id_e    补给站发出的命令/数据包的类型
    uint8_t target_id;
    uint8_t cmd;           //supply_return_cmd_e
    __packed union
    {
        int32_t i32;
        float f32;
        uint8_t u8[4];
        int8_t i8[4];
        int16_t i16[2];
    }data;
    uint8_t eof;           //0xF0
}wifi_supply_return_pack_t;
补给站发出的数据包↑

```

1.2 组网方案

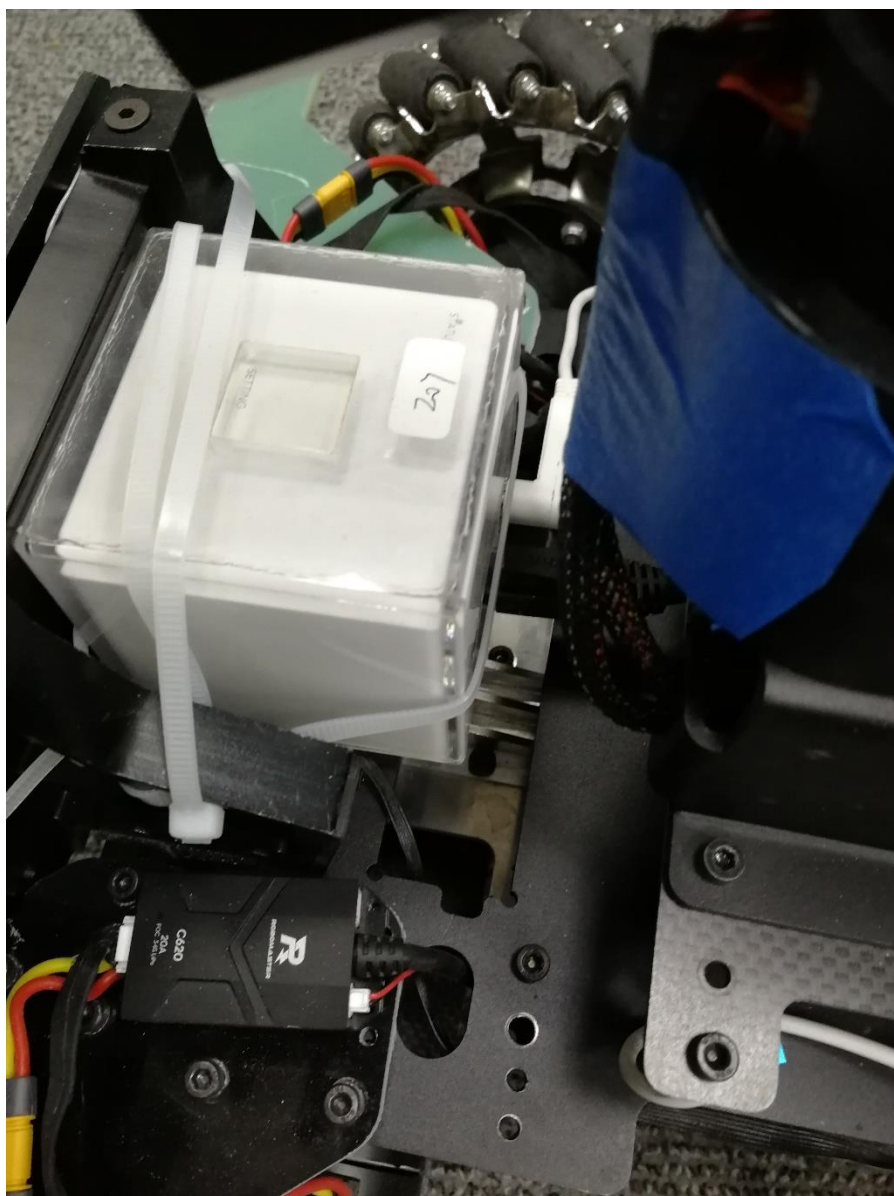
在比赛环境中，2.4G 无线频段的设备较多，遥控器、裁判系统、各种热点 wifi 都挤在 2.4GHz 频段。且规则已经开放了 5G 频段，所以我们这次采用的是 5G 频段的组网方式。

经过市面上的模块挑选，并没有合适的 5G wifi 串口模块，且为了保证信号，采用路由器来作为 wifi 的收发端比较合理。

因为路由器只有网口，所以我们还需要网转串模块来实现和机器人通信

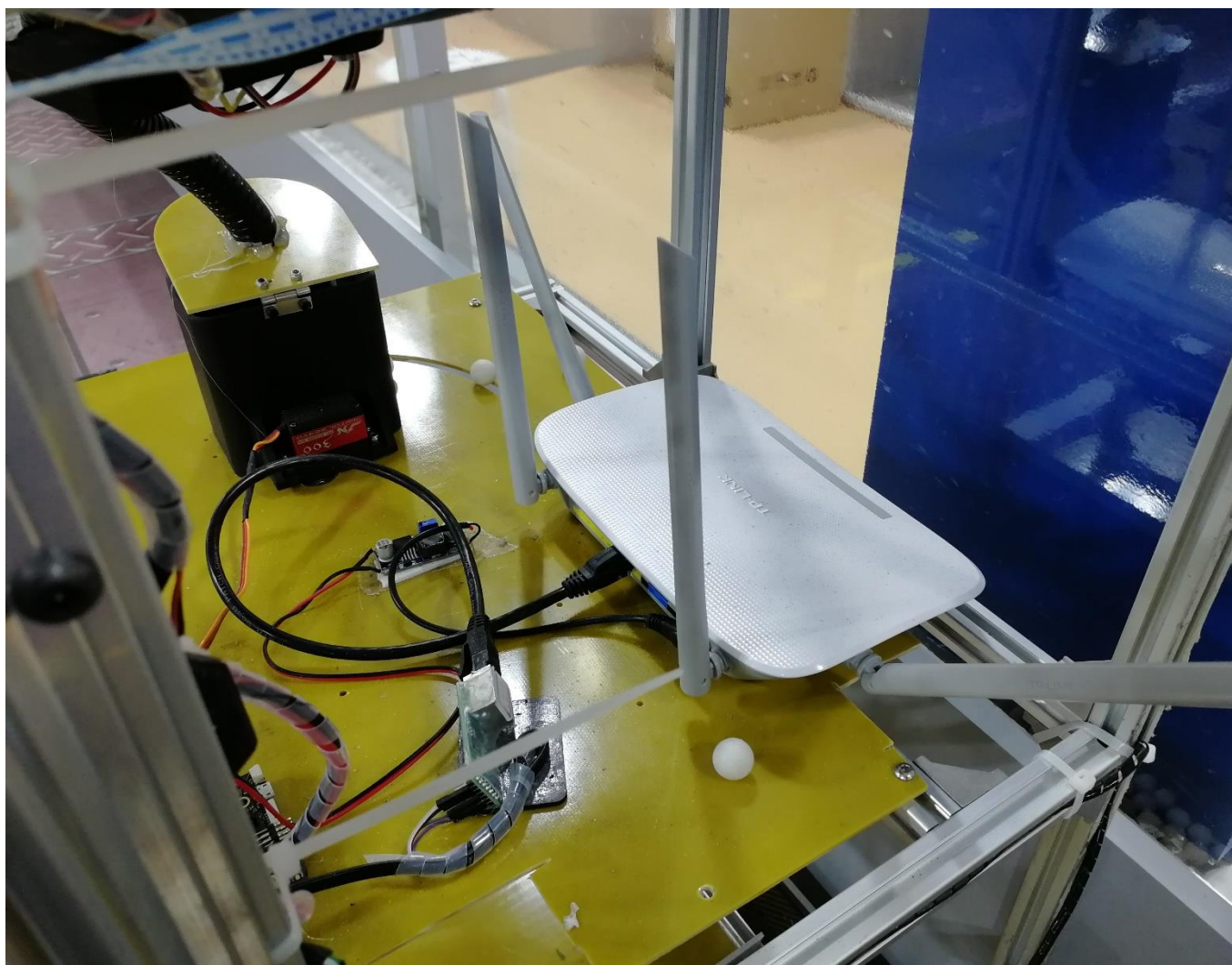
经过挑选，本次使用了日本 AU 公司的 HOME SPOT 小型路由器作为机器人搭载的 5G 路由器客户端

因为该路由器是我目前在市面上能找到的体积最小、成本最低的 5G 路由器（价格大约在 15 块一个），且安装在机器人上体积、重量都很合适



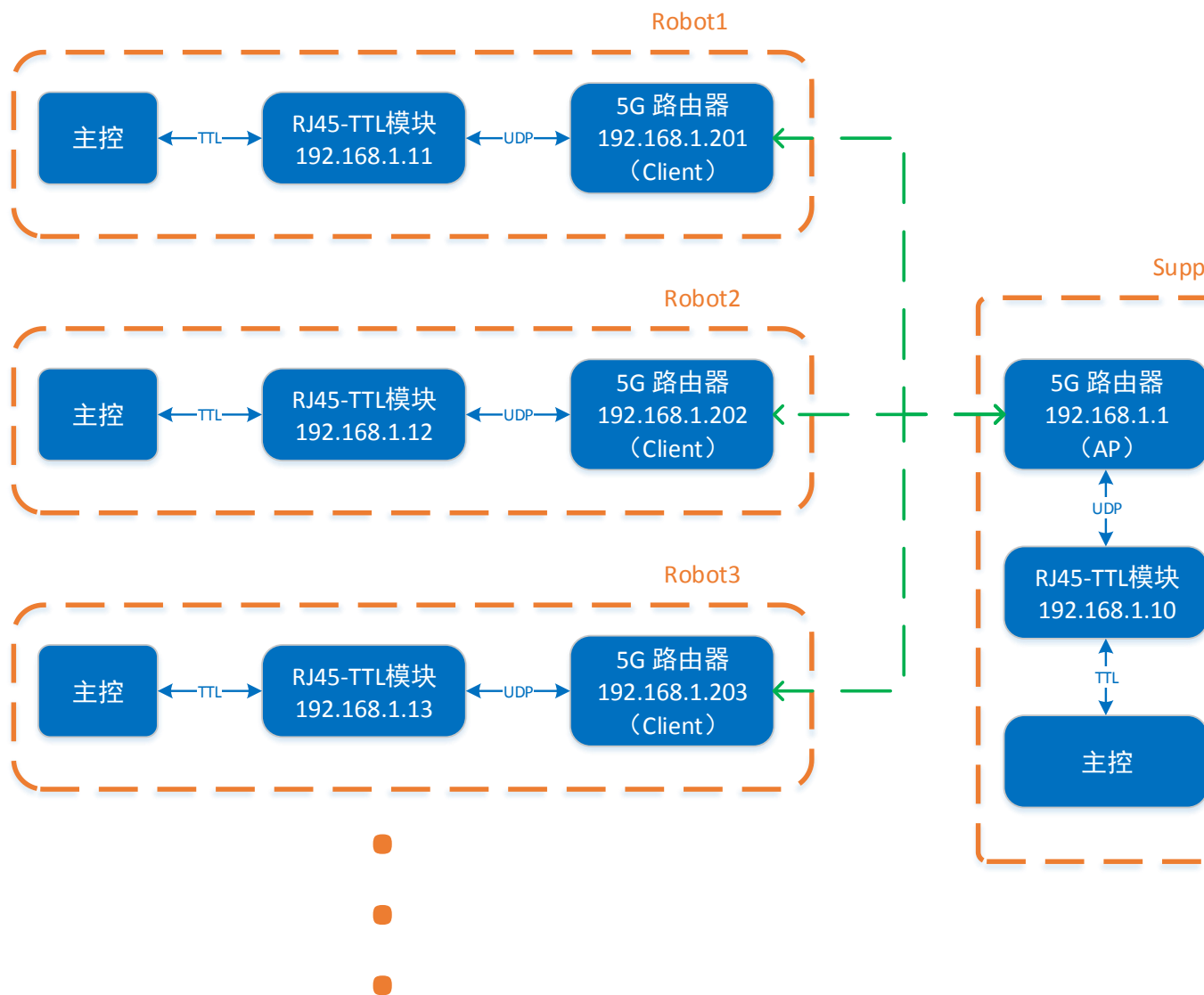
经过测试，该路由器作为热点 AP 时，信号及弱，极难连接上；
但作为客户端连接信号强的路由器 AP 时，通信连接距离可以覆盖整个场地

补给站的 wifi ap 热点使用市面上比较常见的，信号强，价格适中的 TP-LINK WDR5620 路由器（135 元）



补给站和机器人上都使用网转串模块来进行网口和串口的转换





每个机器人以及补给站按照图中示意进行连接，并设置路由器以及模块后，机器人上电变回自动连接进入局域网络，对于机器人主控来说，使用串口 **usart** 进行通信

1.3 路由器及模块设置

1.3.1 路由器设置

目前使用的 wifiap: SSID:Supply_DJI password(WPA2):robomaster

TP-LINK 的路由器按照以上的 SSID 和密码设置，DHCP 服务器最好关闭

1、HOMESPOT 原版出厂的固件是不能作为客户端连接 wifi 热点的，需要刷成某第三方提供的 SDK 固件。

刷固件步骤具体请参照附带资料的教程。可使用 **tftp** 工具或通过浏览器网页上传固件

- 2、刷完固件后，进入中文固件的设置界面后：
- A. 桥接模式，无线设置为选择模式，只开 5G
 - B. Wifi 高级设置全部打开，数字不改
 - C. Wifi 基本设置为 Client 模式，设置要连接到的 SSID，在安全里面设置密码
 - D. 进入局域网设置，DHCP 服务器关掉，设置好路由自身的 IP，局域网内唯一，且在同一个网段内，如 192.168.1.20x；
修改 Mac 地址，局域网内不能有相同的 mac 地址，随便敲一个但一定要符合 16 进制
 - E. 设置完成后，进入系统状态查看是否能连接上设置要连接的 AP

1.3.2 网转串模块设置

本次采购的模块使用串口和网口都可以进行设置。

默认 ip 192.168.1.6

设置成局域网内唯一 IP

补给站设置自身的 ip 为 192.168.1.10，发送的 ip 设置为 255.255.255.255（广播）

机器人设置自身 ip 互不相同即可，发送的 ip 设置为 192.168.1.10（只发送给补给站）

所有的模块的端口设置为同一个

以上两步都设置完成后，如小路由已经连接上补给站 ap，那么模块之间已经可以互相通信

一组嵌入式部分

1. 整体方案

步兵机器人补给用两个通道，每个通道通过触碰开关触发补给动作，配有拨弹轮精确计量子弹数量，英雄机器人补给通过弹仓的上下运动和开合来实现。

1.1. 元件清单

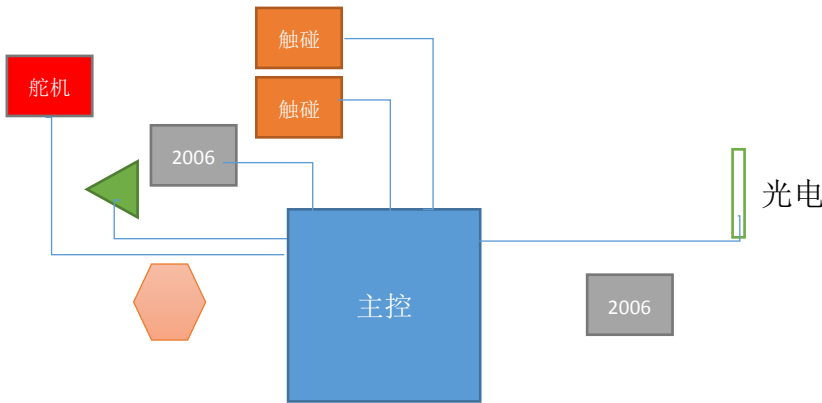
名称	型号	数量
主控板	信仰板	1
接收机	Dji 接收机	1

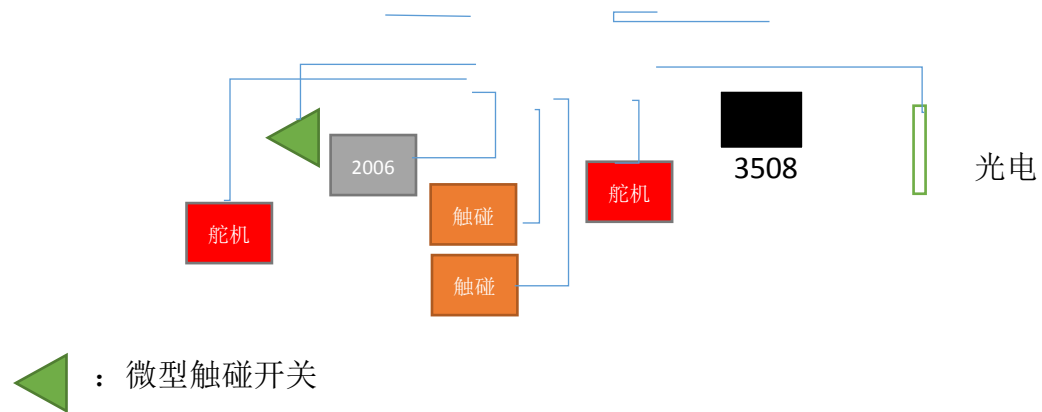
拨弹电机	RM2006 电机	3
同步带轮电机	RM3508 电机	1
拨弹电机电调	820r 电调	3
同步带轮电机电调	C620 电调	1
舵机	360°舵机	2
光电传感器		2
触碰开关		4

1.2. 具体的外设功能内容为下表：

外设名称	功能
接收机串口 usart1	遥控控制机器人的通道（必备），加入 dma 加大传输数据
定时器 TIM2	输出三路 PWM 波控制三个舵机
定时器 TIM3	机器人任务定时基准和功能计时
CAN1	与电机电调通信并控制电机

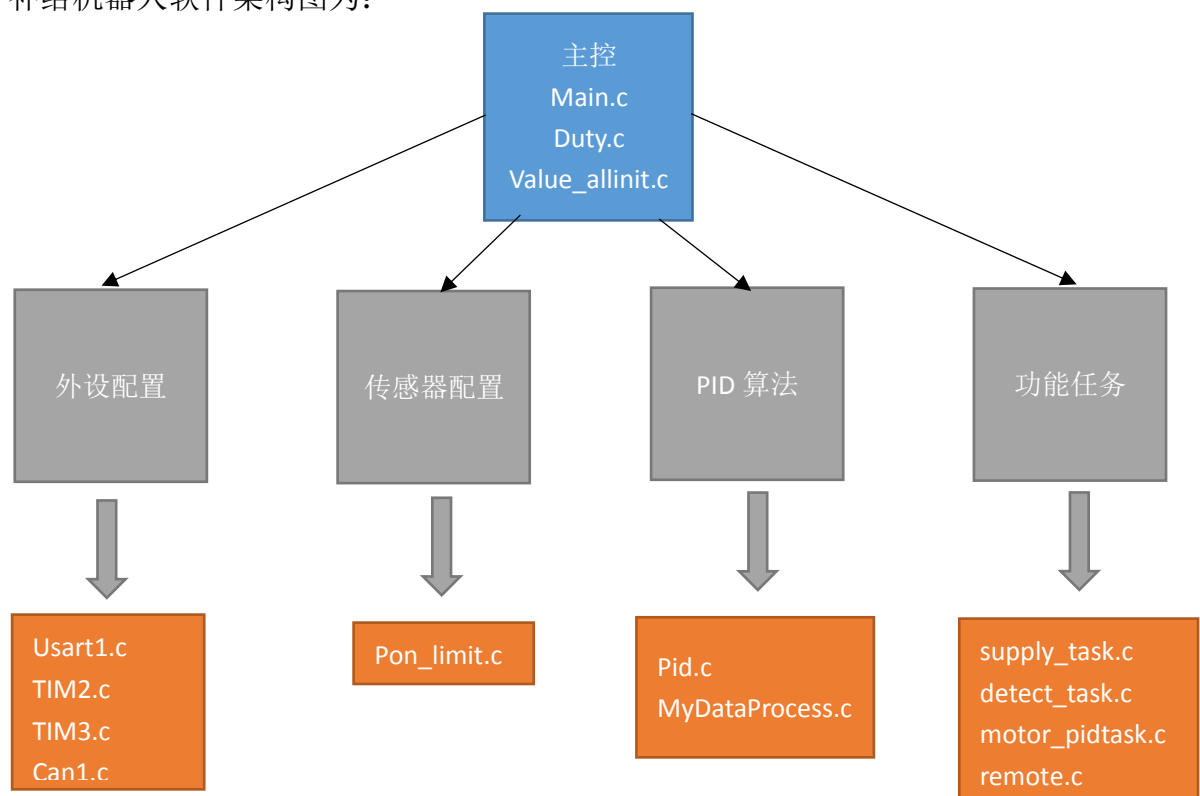
1.3. 控制电路图：





1.4. 软件架构设计和控制框图

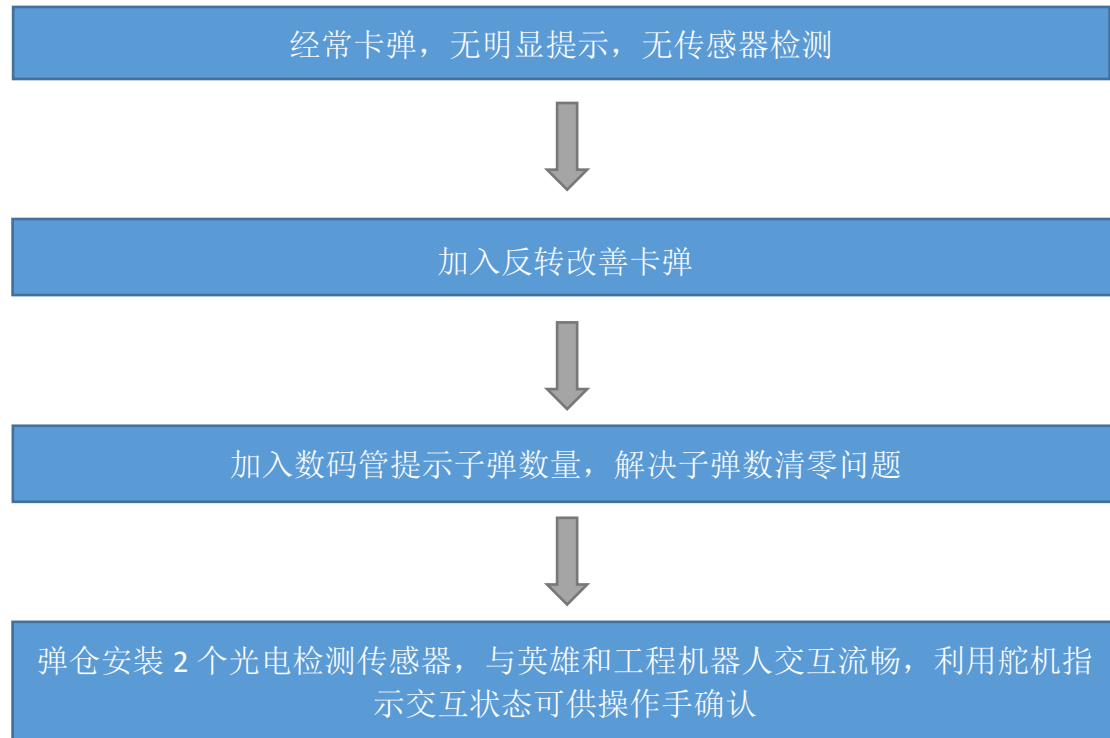
补给机器人软件架构图为：



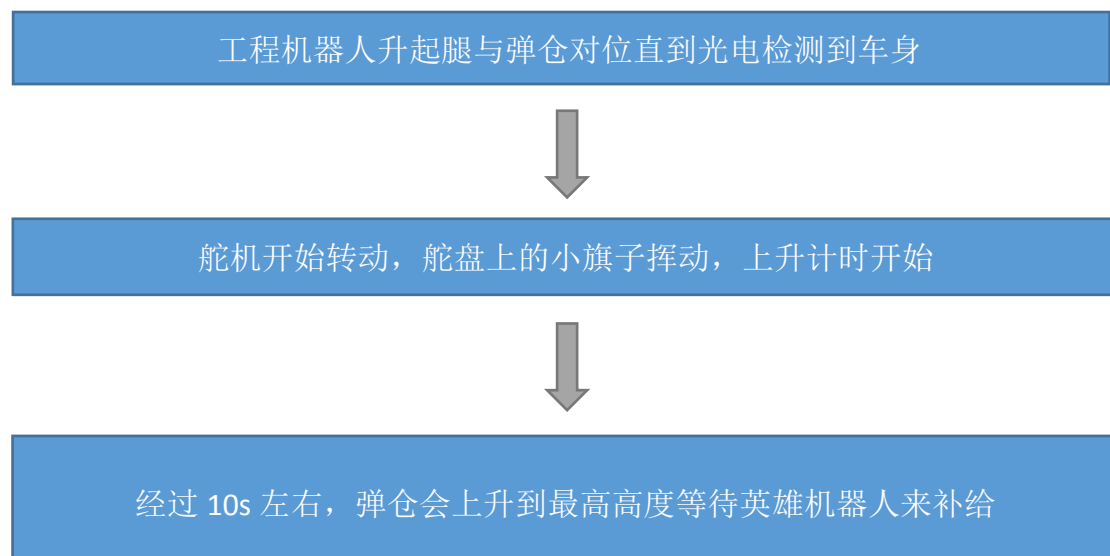
2. 补给机器人开发细节

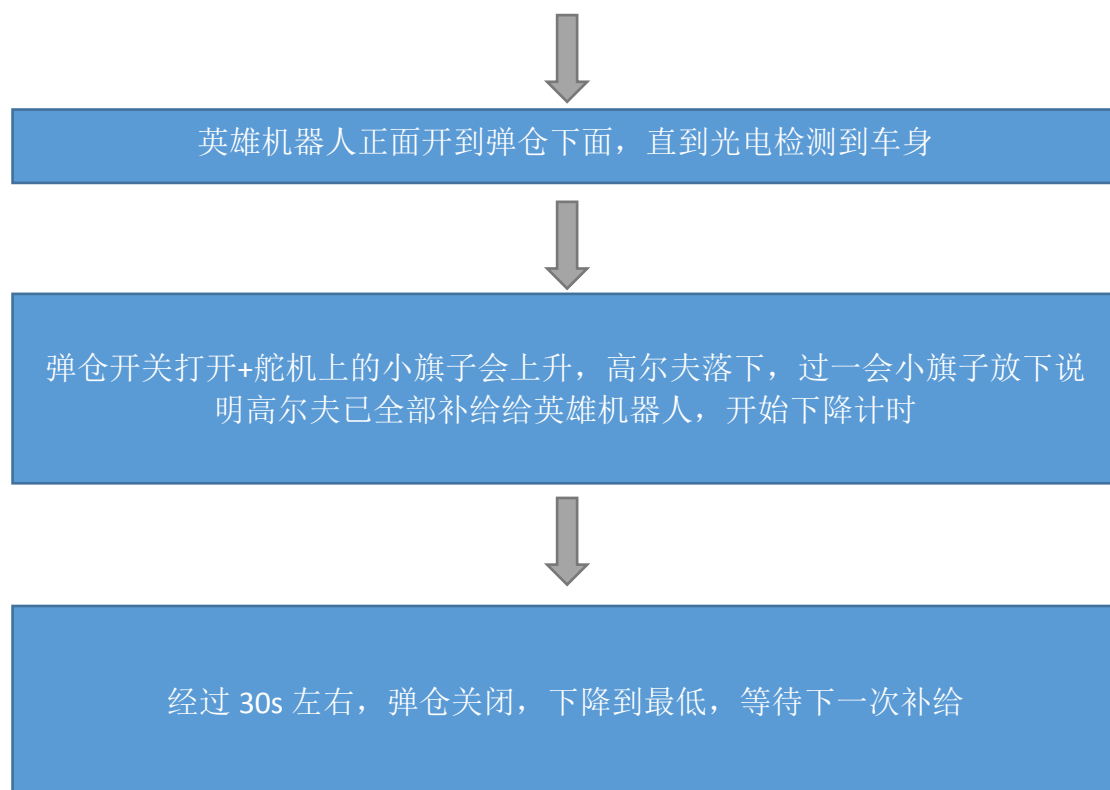
2.1. 方案的优化过程

整个补给机器人开发过程中出现过很多问题，嵌入式方面主要做大量改进和优化，主要几个过程如下：



2.2. 与工程机器人和英雄机器人交互流程





2.3. 17mm 子弹计数算法

```
if(bullet_outer1_limit50_flag == 0)
{
    if(rm2006_fanzhuan_flag1 == 0)
    {
        speed_set_1 += 7 ;
    }
    if(speed_set_1 >= rm2006_1_tar)
    {
        speed_set_1 = rm2006_1_tar;
    }
}

if(key[0] == 0 && last_key0_status == 1)
{
    bullet_cnt_1++;
}

if(bullet_cnt_1 >= 100)
```

```
{  
    bullet_outter1_limit50_flag = 1;  
    speed_set_1 = 0;  
}  
  
last_key0_status = key[0];
```

3. 总结

3.1 预期效果与最终效果

能精确计量步兵机器人补给子弹数量，合理分配子弹，与工程机器人和英雄机器人交互流畅，由明显指示，失误率低。

3.2 方案创新点

1. 利用拨弹方式准确计量子弹数，能合理分配
2. 交互时有提示功能，即检测有效

3.3 存在的坑

1. 拨弹时极其容易卡弹
2. 每个补给通道的小弹仓打开时子弹掉不下来
3. 高尔夫弹仓的开口需要开大一点

3.4 可以改进的地方

1. 在步兵补给弹仓里加入一个光电检测，用来检测是否有子弹，由此来决定是否继续补充子弹，否则每次上电前都要清空弹仓

2. 交互时加入通信方式更加智能。

4. 附录

操作手册：

上电前检查步兵机器人补给通道的小弹仓里有没有子弹，若有，则按住下面的挡板再上电，等子弹漏完立即关电，另一个小弹仓也按这种方式清空子弹，之后再上电，不然会堵塞补弹管。

英雄机器人的补给弹仓的开口要稍微开大一点，开的宽度接近一颗高尔夫的直径即可。

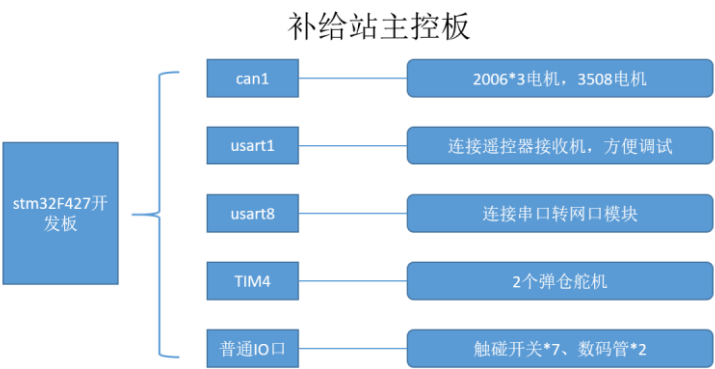
二组嵌入式部分

1. 整体

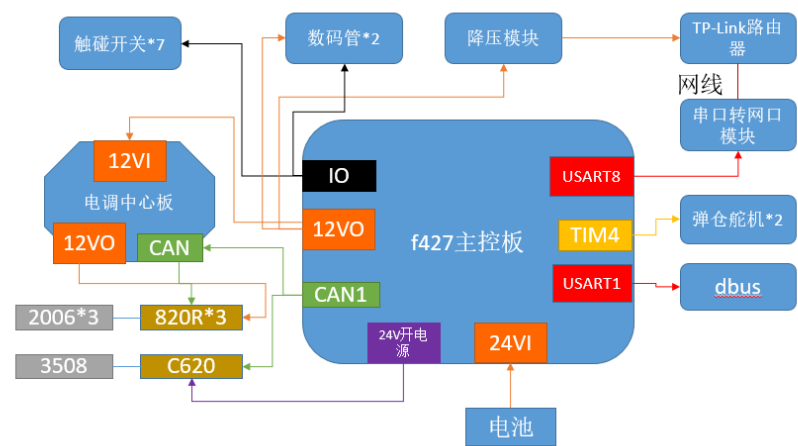
1.1. 元件清单

元件名称	数量
RMstm32f427 开发板	1
RM3508 电机	1
RM2006 电机	3
PDI-6225MG-300 舵机	2
RM 电调中心板	1
触碰开关	7
TP-Link 路由器	1
串口转以太网模块	1
三位数码管显示模块	2

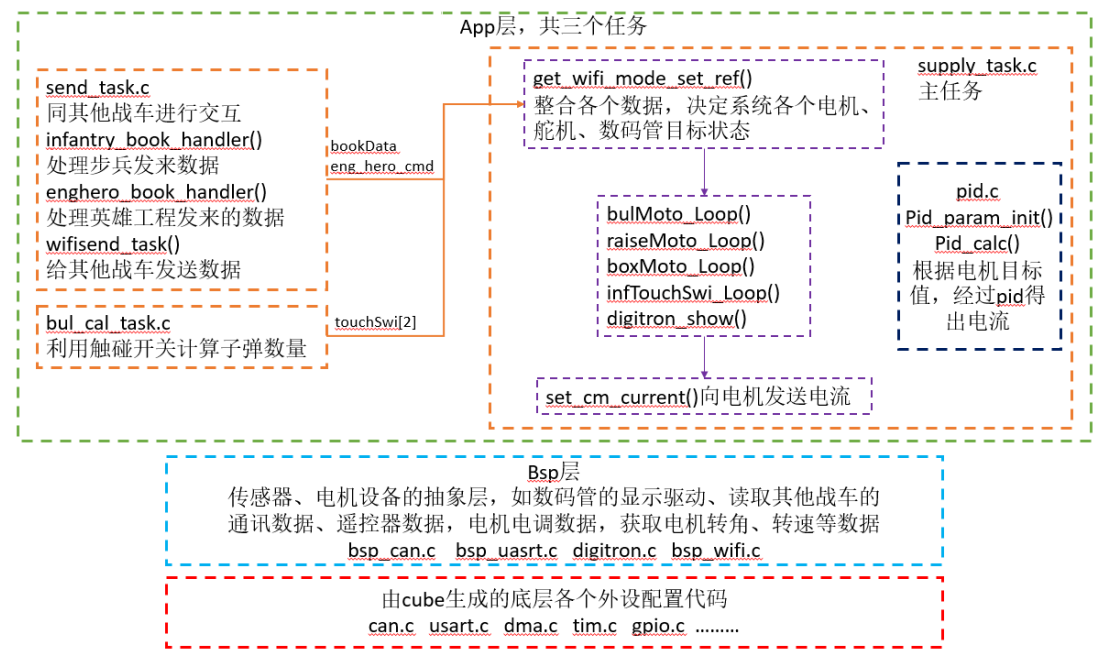
1.2. 硬件、软件资源配置



1.3. 控制电路布线结构，拓扑图



1.4. 软件架构、控制框图



2. 细节

2.1. 方案的优化过程

该期竞培营中，补给站的机械方案虽然具体机械结构一直在变，但是核心思想没有

太大变化,而电控方面则提出了截然不同的两套方案。第一套只完成一些模块的测试(如称重模块、LCD 屏幕的二维码显示是以及摄像头识别二维码),并没有做成整车进行测试,在后面开会时被否了;而第二套则是搭建了一个局域网,战车之间可以进行无线通讯接发数据,相比比第一套方案传递信息的速度、距离、信息量大小都要更好。以下是两套方案的对比:

	一版	二版 (wifi 组网通讯)
数弹方式	称重模块称重	触碰开关计算触碰次数
步兵触发补弹方式	触碰开关	两种触发方式 (1) 步兵通过 wifi 预约子弹,预约成功后,补给站返回步兵应去哪个门补弹(左或右或失败) (2) 触碰开关
步兵获取补充弹药量的方式	步兵撞开关时,步兵上的二维码识别模块扫描补给站上 LCD 屏上显示的二维码获取数据	(1) 步兵预约成功后,按键触发一个信号发送给 wifi 补给站打开对应仓门,并且补给站返回补充弹药的数量; (2) 若没有预约,而是直接撞击触碰开关,则没有返回数据(分不清时哪台车撞的)
与英雄的通讯	(1) 英雄上贴有二维码,补给站上摄像头识别算出英雄的远近,逼近时,抬高高尔夫弹药箱; (2) 英雄对好位后,操作手点亮车上的 LED 灯,供补给站识别,然后补给站打开仓门给英雄补弹	英雄通过 wifi 发送指令给补给站,操纵高尔夫弹药箱的升高以及开仓门
与工程的通讯	工程上贴有二维码,补给站上摄像头识别算出工程的远近,逼近时,降低高尔夫弹药箱,供工程车放置高尔夫	工程通过 wifi 发送指令给补给站,操纵高尔夫弹药箱的升高以及降低

3. 总结

3.1. 预期效果与最终效果

预期效果: 实现步兵的预约补弹、工程英雄的远程操作弹箱的功能

最终效果: 顺利实现步兵的预约补弹、工程英雄的远程操作弹箱的功能。

步兵操作手预约后,会返回数据 1 (左边补弹), 2 (右边), 999 (失败) 显示在用户数据上,到达补给站后,按键打开仓门,并获取实际补充了多少子弹的数据(若是撞触碰开关,则必须在±4s 内按键,前提是必须预约成功的

步兵)。但是，若已经预约的步兵挂了，只能队友通过触碰开关去碰撞来补弹（因为坑被占了，取消不了）

3.2. 方案的创新点

通过建立数个路由器建立局域网，各个战车经过串口转网口模块可以实现远程无线通讯，对提高战车之间交互能力很有意义。

3.3. 存在的坑

- (1) 目前只在楼下场地使用过这套交互系统，如果在信号更加嘈杂的环境中，不一定能够很好的完成通讯，可能会经常有战车掉线的情况发生。
- (2) 最后一周补给站才出来，调试时间不多，不清楚三台步兵同时预约、或多次预约是否会出现逻辑上的 BUG。

3.4. 可以改进的地方

- (1) 步兵一直给补给站发送心跳包，让补给站知道步兵们是否战死，若已预约的步兵不再发送，就可以取消它的预约，把坑让给上一个预约却没有成功的战车。前提是，通讯足够稳定，楼下的场地没问题，但实际比赛场上可能出问题。
- (2) 补给站的重要功能之一就是补了多少弹药发送给补弹的战车，目前只能将弹药量发给已预约了、并过来补弹时按键获取数据的车，不够好，还是希望能通过在补弹站上加摄像头扫车上二维码（或其他特征）来确认是哪台战车，否则对提升补给站性能有诸多掣肘。
- (3) 目前步兵的预约操作略复杂，在比赛中我们的操作手几乎没时间去管这个，都是直接撞开关，需要简化操作。如按一键，直接将战车上剩余弹量发给补给站，设一个封顶值（如 200），补给站直接准备这个差值数量的弹药即可。

4. 附录

1. 操作手册

- (1) 补给站的上方有一触碰开关，一直触碰它可以触发清弹功能：高尔夫弹药箱升高高度重新定位并打开仓门清除高尔夫；两个舵机打开，上方的两个拨弹电机一直旋转将小弹丸清出。
- (2) 步兵预约子弹、英雄工程远程操作补给站的操作详见第 2 组各个战车技术文档中操作手册部分。