

2018RM 夏令营二组技术报告

组员及分工：

龚楚越	协调人员，跟踪进度，购置物资，嵌入测试、剪辑视频，制作 UI、视觉开发
刘臣轩	编写算法，制定方案，设计结构，装配整车，编写嵌入，调式整车、视觉开发
段权耕	机械建模、装配整车
张捷	机械建模、装配整车
蔡子逸	机械加工、装配整车、视频剪辑
罗匡	算法测试、物资管理、嵌入测试、视频剪辑
邢庆宇	嵌入测试、视觉开发、进行技术储备

一、需求分析 (由刘臣轩同学设计)

(一) 第一阶段：

任务要求从资源区夹取方块放置到路基上，需要设计夹取装置和抬升装置，资源区积木摆放方式为窄边朝内，且出于资源区两积木间隙的考虑，车辆不易设计得过宽，所以设计夹取窄边（2 格积木）的装置效率会较高，通常情况下夹取时间会远大于路程时间，而且考虑到资源区方块间距问题，设计储存机构将导致车辆尺寸大幅增加，容易在资源区中撞到方块，所以无需设计一次夹取多个方块或存储方块的结构。

算法部分，第一阶段生成连通所有城堡的铺满地图，由操作手从中心向边缘按颜色区域铺设方块，UI 的设计要求简洁、大方、美观，以给操作手提供良好的视觉体验。

(二) 第二阶段：

1. 进攻方：需要设计能夹取城堡边缘（4 格积木）的夹取装置，优先将防守方的

基础城堡(第一阶段默认的城堡)整体运回相应颜色的资源区,如果防守方有能力搭建多色城堡,则拆除该城堡周围的路基方块,或根据方块颜色和数量判断是否移入某一颜色资源区。

2. 防守方:需要设计垒放城堡的装置,由于需要从资源区获取方块,夹取窄边(2格积木)的格点(积木突起的点)效率较高。需要编写算法便利现存路基,计算按照何种顺序升级路基得分最高。
3. 第二阶段会有交替十五秒的黑屏,在评估了视觉算法稳定性以及现场情况复杂性的基础上,我们认为方块识别的单目视觉无法对目标任务进行任何有效辅助,而SLAM建图的开发复杂度远超过收益,所以我们没有编写任何自动化程序。

二、所需技术点(关键词)

(一) 机械:皮带轮抬升、丝杆夹取

(二) 嵌入式:单轴云台、串级pid控制

(三) 算法:深度搜索、遍历、快速排序

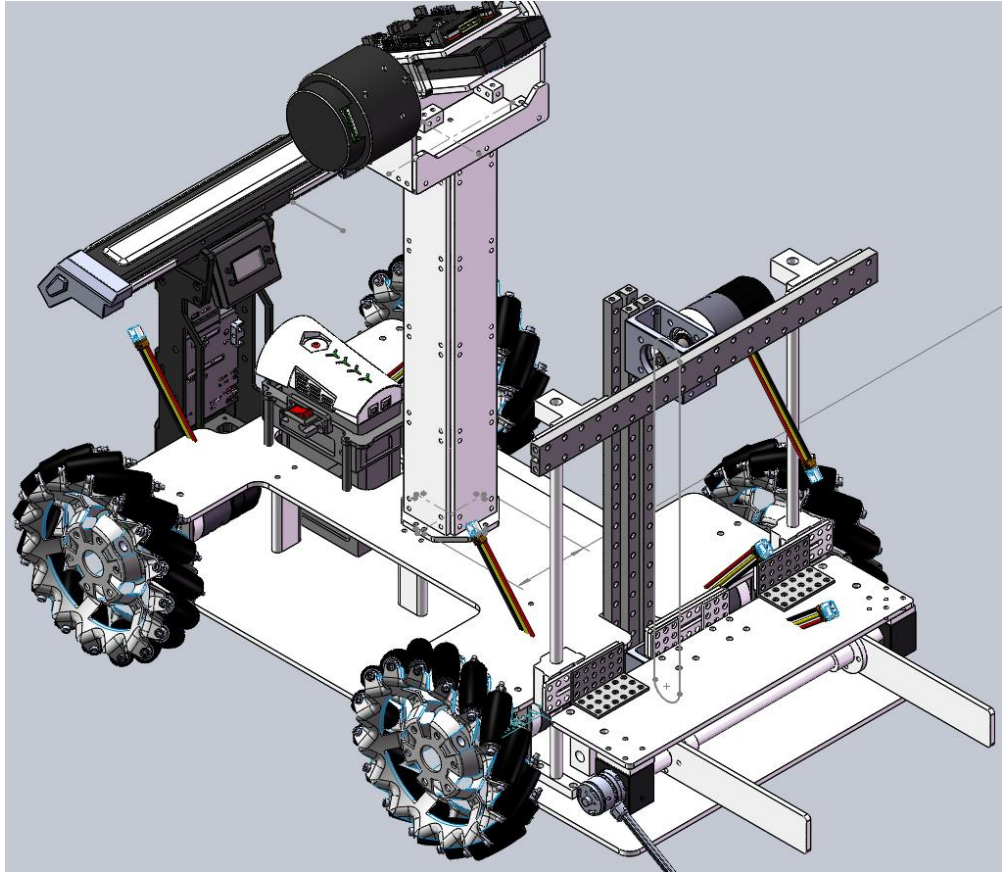
三、总体方案(由刘臣轩同学设计)

RM 智能电池连接主控灯柱。主控灯柱 24V 无功率限制输出(白线)连接下底板中心板,连接四个 C620 电调并连接到 M3508 电机,下底板中心板通过 8pin 线与云台 6623 电机相连,再通过 XT30 转接线与 RM 主控板(架设在云台上)相连。主控灯柱 24V 有功率限制输出(红黑线)连接上底板中心板,连接 C620 电调控制抬升结构的 M3508 电机,连接 C610 电调控制夹取结构的 M2006 电机,通过 XT30 转接线(自行焊接延长 can 线)与 RM 主控板的 can1 相连。主控灯柱图传线与图传(架设在云台上)相连,测速模块连接线与 UWB(架设在主控灯柱上)相连。遥控器接收机通过 DBUS 线连接到 RM 主控板 UART1,遥控器连接到 PC 上,通过键盘键位控制抬升和夹取。

四、各模块方案

(一) 机械 (除特殊注明外, 由刘臣轩同学设计, 段权耕、张捷同学制图, 全部建模

见附件 EastFlowMechanic)



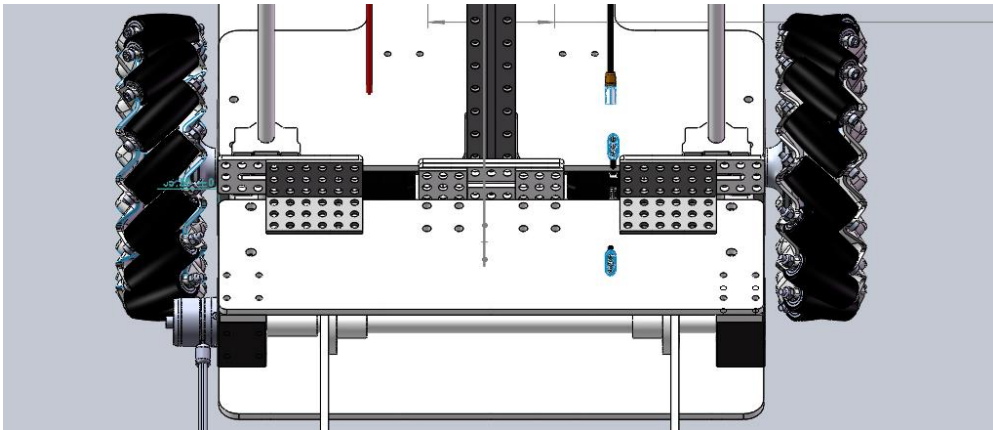
下底板上安装主控灯柱, 整车动力系统, 抬升结构竖直固定光杆、铝型材, 以及支撑上底板的立柱, 在中心位置安装中心板, 连接动力系统的 C620 电调。

上底板上安装电池支架、云台支架、在前部安装中心板, 连接控制抬升装置 M3508 电机的 C620 电调和控制夹取装置 M2006 电机的 C610 电调。

抬升装置 (由蔡子逸同学设计), 采用电机驱动皮带轮带动皮带进行升降传动, 灵感来自于 CRC 比赛的升降结构, 又因为 CRC 机器人体型较大, 所以对抬升结构做了适当的精简与优化: 将抬升动力安置在车的中轴线上, 将夹取装置用型材固定在皮带上, 两侧辅以光杆作为限位及分解受力, 达到平稳高效率抬升的目的。

夹取装置采用正反牙丝杆, 夹片上安装螺纹法兰, 水平光杆穿过夹片上的孔确保夹

片不会在竖直方向上旋转，采用 M2006 电机进行位置控制，能在短时间内夹紧方块。



(二) 嵌入式 (由刘臣轩同学编写，详见附件 EastFlowFirmware\Gimble)

1. 底盘控制：由于没有 yaw 轴云台，将 `chassis_custom.c` 中云台跟随模式改为开环控制模式，在右侧拨杆拨到最上时云台电机上电而底盘能够进行旋转控制
2. 云台控制：屏蔽 yaw 轴电机离线的检测
3. 抬升控制：使用速度环与位置环嵌套控制，获取键盘输入控制角度值的目标值，Z、X、C、V 分别对应不同积木高度，按下时电机将转动使得夹子升降到目标高度
4. 夹取控制：使用速度环与位置环嵌套控制，获取键盘输入控制角度值的目标值，Q、E 分别对应开、合状态
5. 代码架构：在 `execute_custom.c` 中实现抬升部分和夹取部分的初始化代码以及按键控制代码，在 `gimble_task.c` 中调用

(三) 算法

1. 第一阶段 (由刘臣轩同学编写，详见附件 EastFlowAlgo)
 - 1) 输入数据，用 0, 1, 2, 3, 4 分别表示路基、红色、绿色、蓝色、黄色，存储在 `int map[8][8]` 中，并在 `bool castle[8][8]` 中标注初始城堡，在下面的描述中，我们将 `map` 中的元素称作方块，`castle` 为 `true` 的方块称为城

堡。

- 2) 从地图左上角开始遍历，找到第一个城堡，在 `bool visit[8][8]` 中将该城堡标注为 `true`，然后进入 `bool dfs(int x, int y, int block)` 函数，`x`，`y` 分别表示当前坐标的 `x` 值和 `y` 值，`block` 表示当前坐标方块颜色的数字表示。
- 3) 在 `dfs()` 函数中，对当前方块的四个方向进行遍历，仅当存在周围方块（在 `map` 边界范围内）时，进行如下判断。（4 和 5 中的该方块即遍历到的周围方块）
- 4) 若该方块为路基（即在 `map` 中数值为 0），则在 `map` 中将该方块赋值为 `block`（由 `dfs` 函数传入），并继续对该方块进行 `dfs` 搜索，若返回值为 `true`（表示在该方块上铺设 `block` 颜色对应的积木后持续搜索最终可以覆盖全图），则向上返回 `true`（该返回值最终由主函数处理）；若返回值为 `false`（表示在该方块上铺设 `block` 颜色对应的积木后持续搜索最终无法覆盖全图，即在正确的解中该方块不应铺设该种颜色的积木或进入该方块前的某一方块铺设错误），则在 `map` 中将该方块的数值赋值为 0，并继续对当前方块（见 3），的其他方向进行遍历。
- 5) 若该方块为城堡并且该方块颜色与 `block` 一致，并且该城堡没有被 `visit`（已被 `visit` 城堡已被联通，不得再作为路径的结束点），则 `visit` 该城堡，判断地图是否铺满（判断所有方块是否不为路基，且所有城堡是否都被访问过），若铺满则返回 `true`（最后一条路径已被联通，获得解）；若没有铺满，则从左上角开始遍历找一个为城堡且没有被 `visit` 的方块（路径的起点必须为未被 `visit` 的城堡）进行 `dfs`，若返回 `true` 则向上返回 `true`，若返回 `false` 则将该城堡的 `visit` 设置为 `false`，若全图遍历完成没有找到未被

访问的城堡，则说明虽然全部城堡联通但是没有铺满地图，则取消该方块城堡的 visit

6) 若对 3) 中的当前方块的四个方向都进行搜索后仍未找到解，则返回 false

7) 当主函数获得 dfs 返回的 true，则打印该地图。

2. 第二阶段

1) 防守方:对第一阶段所有非城堡方块 (共 48 个进行遍历)，分别计算将该方块升级成单色城堡后得分，对得分进行排序，确定应首先升级的城堡。重复便利，直到所有方块都被升级成城堡，但考虑到机械结构搭建城堡的困难程度，算法对分数的影响将极其微小；如果考虑升级成多色城堡，问题的复杂度将大幅上升，编写算法的实际意义却没有增加，故第二阶段防守方不考虑算法的辅助。

2) 进攻方:使用官方提供的分数数据，选择对周围方块影响最大的城堡进行拔出 (由于情况较少且各城堡差距不会太大，不单独编写算法)

3. 用户界面 (由龚楚越同学编写，详见附件 EastFlowUI)

项目的 UI 框架是 Microsoft .NET Framework，语言是 VB.NET，虽然是一门看上去古老的语言，但是这款语言依靠.NET 框架获得了新生的活力。UI 部分完成了完整封装的按钮、路基/城堡方块控件，并且这些控件出于美学考虑增加了逐帧生成的动画 (参数可调)，达到了美观的目的。在后期，UI 部分完成了 TXT 文本的自动录入、场地的 TXT 生成、Shell 寄生注入运行、线程池管理等功能，达到了快速响应、方便使用的目的。

4. 视觉录入地图(C++版本,由刘臣轩同学编写,详见附件 EastFlowVision\Cpp)

基于 OpenCV 进行开发，对 BGR 通道进行分离后做二值化处理，红色方块仅

在 R 通道中可见 绿色方块仅在 G 通道中可见 ,蓝色方块仅在 B 通道中可见 ,黄色方块在 R 和 G 通道中可见 ,灰色路基方块在 BGR 通道中均可见 ,使用 ROI 将整图划分为 64 个方块 ,通过判断值为 255 的像素点的数量判断该 ROI 是否可见 ,通过上述规律确定该 ROI 颜色值 ,最终输出整图。

5. 视觉录入地图(Py 版本 ,由龚楚越同学编写 ,详见附件 EastFlowVision\Python)

项目采用了 OpenCV 库来读取摄像头 ,并且通过 Pillow 库进行图像处理 ,完成了基于视觉的自动场地录入。基础的思路是通过 OpenCV 获取摄像头静帧 ,之后送入 PIL 进行图像分割 ,分割为 $8*8=64$ 个小方块后进行单个方块的解算 ,统计整个像素中 R、G、B 三种颜色各占的分量 ,通过实验的方法测算出阈值之后即可高效分类单块颜色 ,将所有小方块解算完毕之后即可进行最终数据的生成 ,将数据输入到 UI 即可自动录入信息并且计算最终的方块填充路线。

注 :出于录入数据时长长达两分钟以及视觉对图像的质量要求 ,在实际比赛时不考虑使用视觉辅助数据录入

五、理论分析

(一) 机械 :

1. 轮距轴距 (由刘臣轩同学完成)

资源区积木摆放方式为窄边朝内 ,每个方块宽约 124mm ,两方块间距为 188mm ,故车宽不易超过 500mm ,转动丝杆的 M2006 电机支架宽 23mm ,刚性联轴器长 20mm ,夹取方块窄边 (2 格积木) 的格点 ,给操作手每边预留 25mm 空间 (夹片与被夹方块之间) ,车板材宽度 270mm ,轮距 330mm ,连麦轮宽 360 ,麦轮与邻近方块每边有 70mm 操作空间。

2. 夹取机构选型 (由刘臣轩同学完成)

在双向夹紧的问题上，我们选用了正反牙丝杆的结构，反思分析认为，这种结构虽然体积小，制作简单，但是由于丝杆摩擦力大，M2006 电机扭力有限，旋转丝杆的速度有限，夹取耗时较长。

经过比赛，我们发现由于 M2006 电机反馈值异常的问题，对位置反馈值要求较高的丝杆机构不能很好的被控制，容易造成无法夹紧或夹过紧导致方块滑脱的问题。

经过进一步查询资料，我发现与 C620 电调相比，C610 电调的终端电阻默认状态下是没有开启的，这极有可能导致电调报文无法正常收发，从而导致位置控制异常。

3. 抬升机构选型(由蔡子逸同学完成):在抬升方面,我刚开始选用了型号为 5M 宽度为 15mm 的皮带作为传动皮带,这种皮带强度高,配合专用的 5M 皮带传动轮不易脱齿,有良好的传动性能,皮带及其配套的皮带轮到货以后,我发现了一个巨大的问题:传动轮过宽,和 3508 电机一起固定时会阻挡 3508 与电机架的螺丝孔位,在通过与组员的商量之后,决定不再重新购置新的皮带及皮带轮,这时仓库刚好进了一批皮带,我们去仓库查看后,发现这种皮带能够满足我们的需求:夹子比较轻,可以有效带动夹取模块,但是问题在于仓库没有配套的皮带轮,于是我们与其他组以物换物换到了配套的皮带轮,装上之后发现运行状况良好,满足我们机器人抬升夹取模块的要求。

4. 车辆无法水平移动的问题(由张捷和段权耕同学分析):

- 1) 张捷:在切板后,我们装配好了整个机器,但是,在调试的过程中我们发现了四个轮子的压力不同,经过对不同部件的测试与东流的指导,我们发现出问题出在上下底板上。经过重新的装配,我们解决了轮子压力不同的问

题。

- 2) 段权耕 :在场地中我们发现在执行平移命令的时候 ,机器人会绕着一个后轮旋转 ,在对程序和硬件与机械结构的多方面检查与向导师咨询之后 ,最后发现是上底板将下底板拉成了弯的导致了机器的轮子悬空的问题 ,于是我们去掉了几个下底板与上底板的连接 ,问题就减轻了很多。

(二) 算法 :

1. 第一阶段 (由刘臣轩同学完成): 对于不同的所给地图 , 深度优先搜索的复杂度不一 , 在比赛题目范围内的数据的计算时间为毫秒级 , 异常极端数据 (大片空白区域) 花费时间较高但人眼可以秒解 , 比赛中不可能出现 , 不做考虑。

(三) 嵌入式 (由刘臣轩同学完成)

1. 抬升控制 : 角度变化较缓 , 防止剧烈升降导致积木滑脱 , 双环 pid 控制参数如下 :

```
pid_init(&pid_lift_speed, 10000, 0, 5, 0, 0);  
  
pid_init(&pid_lift_angle, 7000, 0, 15, 0, 0);
```
2. 夹取控制 : 尽可能在不抖动的情况下加快响应速度 , 减少夹取花费时间 , 双环 pid 控制参数如下 :

```
pid_init(&pid_pinch_speed, 20000, 0, 2, 0, 0);
```

```
pid_init(&pid_pinch_angle, 30000, 0, 20, 0, 0);
```

3. Pitch 轴云台 : 在负载较低 (没有弹仓等装置) 的情况下 , 需要减小 kp 防抖 , 双环 pid 控制参数如下 :

```
pid_init(&pid_pit, 2000, 0, 25, 0, 0);
```

```
pid_init(&pid_pit_speed, 5000, 2000, 2.5, 0.1, 0);
```

六、制作与测试流程

- (一) 方案讨论：看组内会议进行头脑风暴，集思广益考虑方案。
- (二) 方案决策：对多种方案进行评估，确定最终方案。
- (三) 方案实施：算法部分对一阶段进行分析，实现算法；嵌入部分阅读例程和实践指导书，搭建控制框架；机械部分使用 Solidworks 进行建模；界面部分搭建框架。
- (四) 方案修改：根据场地实际情况对方案进行小幅修改。
- (五) 算法测试：算法与界面进行联调，使用 FreeFlow 游戏中的 180+ 样例对算法进行测试，尝试生成极端数据对代码进行测试。
- (六) 机械修改：由于作图规范问题，需要按规范重新作图。
- (七) 整车安装：对车辆进行整体安装，嵌入式接线理线。
- (八) 整车调试：嵌入和机械对车辆进行调试与优化。

七、结果与评价

- (一) 算法方面：算法方面的熟练度和效率无疑是全夏令营最高的，仅一天时间就完成了
一阶段算法的编写以及 UI 和算法的联调，高效而又稳定，但是很遗憾没能实现什
么创新技术，第二阶段算法也由于其他方面工作过多而没有时间编写。
- (二) 机械方面：
 - 1. 画图：由于组内机械同学 Solidworks 软件使用经验不足，画图习惯不良的原
因，作图效率极低，简单的设计需要花费大量的时间才能在图纸上呈现，而且
错漏百出，稍做一点工作就需要其他人进行审核，思路极不清晰，每一步需要
做什么操作，每个零件需要装配在怎样的位置都没有相应的概念，缺乏自主思
考的能力。
 - 2. 安装：整车的安装经常出现错漏，在没有审核的情况下，经常出现螺丝没上
紧或空位安装错误导致干涉的情况，安装时也没有清晰的思路，在没有指导

的情况下不知道每步应该安装什么组件。

注：个人认为机械能力弱到如此地步不应将锅推在组队时不注意或者管理无力上，建议官方对机械方向招募营员进行更严格的审查，避免能力过差的学生进入冬夏令营，毕竟对于研发机器人来说，机械结构的稳定是其他一切的基础，没有机械结构，再快的算法也无法被实践，再厉害的嵌入也无法让车辆正常运行。（刘臣轩注）

(三) 嵌入式方面：嵌入式方面的框架搭建很早就完成了，而整车调试一直受到机械方向进度的影响，机械结构搭建完成后很快就完成了数据调试，这主要归功于官方对stm32 底层代码优秀的封装，各项操作只要逻辑清晰就能够实现，避免了太贴近底层而容易迷失本质的问题。

八、感想与感悟

(一) 龚楚越

其实对我个人来说，参加夏令营的目的更多是去锻炼自己的管理才能，在技术方面并没有太多新东西需要学习。在这短短 20 天的时间，管理的紧凑以及完善其实是一件非常不容易的事情，在冬夏令营有过体会的我在这次的体会要比上一次更深一个层次。本次的队伍在刚开始组队的时候是比较和谐的，但是在实际合作往同一个目的地前进的时候，还是遇到了各种各样的队伍配合问题。或许是因为磨合时间太短，对各自的了解都不甚足够，队员们曾出现过比较大的争吵，也有赌气、闹小脾气的情况出现。但是其实在夏令营的结尾，大家都学会了相互适应、相互包容，小摩擦虽然必不可少，但是真感情的促进我看在眼里。

其实作为新生代的年轻人，能来到这个夏令营就是有能力的表现。而相信大部分这样的人不甘心用自己比同龄人强大的能力来为未来的老板打工。于是，创

业成为了崛起的唯一机会，而创业最危险的时刻很大一部分就是公司初创的时期，挺过这段时期，等待事业走上正轨，一次创业也可以说算是成功了。在这个初创团队刚刚萌芽的时期，一个有凝聚力、有能力并且可以处理好各种摩擦的领导往往能够给整个团队带来成功的希望。所以我在这次夏令营选择了作为一个组长，来处理、磨合整个团队。这次夏令营虽然不知道最终结果如何，但是依旧是一次难得的机会，一次在实战中历练自己的机会。

希望整个团队的每一位，在未来都有个美好的前程吧。

(二) 刘臣轩

从技术层面讲，本次夏令营我并没有太多的接触什么新的技术，算法方面只是复习了深度优先搜索这一基础算法，练习了断点调试的能力；嵌入方面的内容基本没有脱离实践指导书，而这些内容在学校校本课程中也都有过接触了；机械方面没敢对自己不太熟悉的结构进行尝试，没有做创新的结构设计，基本是在借鉴与参考，还频繁麻烦东流哥，一出现问题就以自己机械方向知识不足为借口张口就问，缺乏自己的独立思考，这需要反思。

但本次对车辆的整体设计很好的锻炼了我的系统思维，集成化系统的设计是一门复杂的学问，底盘车的整体设计让我体会到了系统设计的难度，让一个系统和谐运行需要思考的问题是非常多的，只有从简单的系统做起，最后才能逐步具备设计复杂系统的能力。

在与队友合作方面，我没有评估好队友的能力，例如机械方面我没有实际参与画图，却在审图时没能克制自己的情绪，对队友的一点点错误吹毛求疵，造成队友对画图丧失了信心，对画图产生绝望，这是我应该反思的，明明自己能力以及知识储备都高于队友，就应该承担更多的任务，而将较简单，不那么紧迫重要

的任务交给队友，我在与人合作时不应以自己的极致标准苛求别人，需要多考虑到队友自身能力的限制，多照顾队友的心理感受与体验，不能仅站在自己的角度思考问题。

与冬令营相比，我从参与项目的组员逐步成为了领导项目开发的技术领导地位，这首先对我各方向的技术能力是一个不小的考验，客观评估，我的机械技术水平远低于其他机械方向的老营员，我对材料特性的了解明显不足，对实现常用功能的一些常用结构仅有概念上的基本认识，缺乏实践经验，在这方面我更应多学习新的知识，提升自己的能力。成为项目领导者更是对我与人合作能力的考验，我的性格使得与我合作的人会有很大压力，我总是以为所有人都具备跟自己一样的技术能力，一些事情能理所应当的完成，在队友没能完成时往往以责备的语气表达自己的失望，客观的评判，今后和我合作的人不可能人人技术能力都与我相当或高于我，我需要学会擅用他人的长处，在他人无法完成预定任务时多提供理论和方法指导，避免盲目的无能狂怒，才能够更好的解决问题。

大学我希望进入港科大的 ISD 系，学习更多与综合系统设计相关的知识，继续锻炼自己的能力，在未来成为优秀的工程师，做出有创新、有技术的改变世界的产品。

(三) 段权耕

在来到 RM 夏令营之前，我对 Solidworks 是一窍不通的。以前所做的加工设计及成车完全通过铝型材的搭建和对平面画图软件（如 CorelDRAW）的使用进行的，对 Solidworks 只有非常浅薄的理解，但是来到 RM 夏令营之后，我发现，Solidworks 的使用对于机器人的设计及加工和实用程度的检测都有非常大的帮助，极大的提高了机器人制作的效率和质量。于是，在夏令营的制作要求下，

开始学习起了 Solidworks。一开始，也能自己摸索的画出草图，建出模，完成装配，但是觉得无比的麻烦，速度也非常的慢。直到有一天，东流哥来看我们组的画图，看到我的画图之后对我的画图非常不满意，然后教了一些 Solidworks 的画图基础给我，我才知道，Solidworks 画草图需要定义，可以利用中心线和特征来更快的进行画图。第二天，东流哥又教了我画图时思路要清晰，要明白如何切除，拉伸才能更加有效率的画出正确的图，还教我将各个特征分离开进行绘制，在更改起来的时候会更加的方便。经过了东流哥的指导，对 Solidworks 有了更深的理解，用起来也更加顺畅了。

我在夏令营学习到了很多没有从前没有了解过的新知识，如 Solidworks 的绘制，链条、丝杆特性与属性的了解等等，感觉在同学之间的交流之中学到了很多。同时，在夏令营之中，认识了各种各样的夏令营营员，生活在一起，并结识了一些朋友。在夏令营之中，在这一个比学校社团专业的得多的环境中，对以前的一些由经验得出的结论进行了修正与补充。总的来说，这次的夏令营对我来说是一个学习的过程，我在其中学到了很多，下次如果有时间，我还会再来。

(四) 张捷

这次夏令营，我主要学习了 SolidWork 的使用。在方案讨论完成后，我和段权耕共同完成了机器绘图的任务。在刚开始时，我们对 Solidwork 中绘制草图并不规范。因此，我们在装配过程中遇到了很多问题。之后，经过了东流老师的指导，我们规范的绘制草图，使得我们会快的完成了机器的总装。

除了技术上的收获，我同时认识到了许多志同道合的同学。在紧张而充实的夏令营期间，我们共同学习了技术知识和团队配合。

在未来，我希望还有参加 Robomaster 夏令营的机会，能继续学习技术知识，认

识更多的同学。

(五) 蔡子逸

本次来 RM 夏令营最大的感想是：人外有人，天外有天，之前沉浸于自己的舒适圈，在学校当队长的时候认为自己已经做得不错了，但来到 RM 夏令营后，发现自己与他人有很大的差距，在认识到这个问题后，我还是没有改正自己容易骄傲的坏毛病：在一次加工丝杆的过程中，当时我认为自己已经加工的很好，在组员面前吹嘘，但是辅导老师的教导将我从泥潭中拉起，我认识到了自己的缺点：容易骄傲。我开始反思自己的问题，在接下来的工作与实践中，我放平了心态，扎扎实实做好我的本职工作之外，也帮助队友做其他机械方面的事情，不轻易自我满足，戒去了骄傲的心。除自我修养之外，这次夏令营我也学习到了很多知识：包括 SW 的使用、装配、受力分析、实体渲染等，各种材料的专业名词，各种机械的搭配使用，都是我之前没有注意到的知识盲点，在以后的实践中都是十分受用的知识。除此之外，我也十分感谢我的队友，一次次的醍醐灌顶，让我认识到了正确的工业流程和作为一名机械应该做的事情，在每一次批评之后，我深刻反思自己的问题，与他人交流，改正自己的问题。

感谢 RM 夏令营给我这次学习的机会，让我受用良多，我们今天冬天再见！

(六) 罗匡

首先，我一只菜鸡能来到这个 dalao 云集的夏令营我十分感动，在这里我学习了各种新的知识 (OpenCV , unity , STM32 开发 , PID 算法) , 也结交了许多朋友。

在队中主要负责后勤工作，点出了高速螺丝寻找的技能 (雾) , 并且几乎每天都去拜访仓库，虽然觉得这样有点骚扰仓管大叔就是了。(/▽ \)

在组内研发的过程中，各种各样的经历也让我明白了团队协作的重要性，纵然因为一些原因比赛前我们组仍在装车，但我觉得这并不是什么大事，

只要大家能高效的协作，总是能赶上 deadline 的吧(。·v·)ノ

小矛盾总是会有的，大矛盾也不是不能调和的，再不行找找旁观者调和一下嘛 o(*^▽^*)o

最后送一个简短的 ABC 理论给在座的各位：

“通常人们认为，人的情绪的行为反应是直接由诱发性事件 A 引起的，即 A 引起了个体的情绪及行为结果 C

事实上诱发性事件 A 只是引起情绪及行为反应的间接原因，而人们对诱发性事件 A 所持的信念、看法、理解 B 才是引起人的情绪及行为反应 C 的更直接的原因。”

(七) 邢庆宇

这次夏令营，我在开始阶段调试了 M3508 和 M2006 两类电机的一致性以及 PID 的初步调整，为下一步的嵌入式的开发提供便利。随后我又与刘臣轩、龚楚越分别完成了三套视觉读取题目纸的系统，实现方法各不相同，我用的方法是多边形逼近然后判断边和角，这种一题多解考验了我们独立思考的能力，让我懂得了团队协作和集思广益的重要性，便于筛选出更优的方案。经过这次夏令营活动我明白了从构想到实现是存在有距离感的，这种距离感需要多技术方向的成员共同攻克，实现硬件与软件，软件与算法更好的结合。

同时我也认识到了自己与其他组员相比存在的经验上的欠缺。我之前一直比较注重科学方向，这是第一次协作完成多技术方向的工程项目，我也充分认识到在实践中更能发现科学方面的问题。其次，我完成的一些工作，例如课上讲的

FFT,SVM 等其实是不会用到的，我把这些工作看作知识储备。也在此感谢组内成员的包容和理解！

未来我希望能将科学方向与工程方向实现结合，多参加一些有关的活动，锻炼利用技术来协作完成项目的能力。