



VEX 24-25 工程笔记

6353B

北京中学国际部高中组

December 2024



目录

1 团队介绍	4
1.1 团队文化	4
1.2 队员培训	5
1.3 合作方式	5
1.4 队员介绍	6
2 赛题分析	7
2.1 0.1 规则	7
2.2 0.2 规则	9
2.3 1.0 规则	10
2.4 2.0 规则	12
2.5 2.1 规则	13
2.6 技能赛	14
3 机型迭代	15
3.1 0.1 版	15
3.2 1.0 版	16
3.3 1.1 版 (1.0 改)	18
3.4 2.0 版	23
3.5 3.0 版	28
3.6 3.5 版	30
3.7 4.0 版	33
3.8 4.5 版本	34
3.9 整体研发日志——问题与改进	36
4 打法策略	40
4.1 针对 0.1 和 0.2 规则的策略设计	40
4.2 1.0 规则的策略改动	40
4.3 2.0 规则下的策略	41
4.4 2.1 规则后备战上交比赛的策略	42

4.5 技能赛思路	43
5 操控算法的设计思路	44
5.1 基本控制	44
5.2 PID 底盘	45
5.3 半自动设计 (3.x 版本)	48
5.4 半自动设计 (4.x 版本)	49
6 自动程序思路	50
6.1 1.0&1.1 版机器	50
6.2 2.0 版机器	54
6.3 3.0&3.5 版机器	59
6.4 4.0&4.5 版机器	65
7 历次比赛	70
7.1 北京延庆城市赛	70
7.2 北京海嘉城市赛	72
7.3 秦皇岛华北大区赛	76
8 VEX 机器人选修课周记	81

1 团队介绍

北京中学国际部 VEX 高中校队，队号 6353，成立于 VEX24-25 赛季初期，同时从 24-25 High Stakes 赛季开始正式设计研发、训练、参加比赛。北京中学国际部 VEX 高中校队虽然是刚刚成立的新战队，但是我们有信心在 20-25 High Stakes 赛季打出我们想要的成绩！Come on BAID！

1.1 团队文化

北京中学国际部 VEX 高中校队的目标是变得跟钢铁侠一样做一套自己的能飞的外骨骼……

在人类的历史长河中，科技突破的速度随能实现的事情越来越多而呈指数上升，这般加速度的动力正是来自人类在近百年所实现的数不胜数的科技突破，正是这些数不胜数“奇迹”推动着人类的快速前进。这是伟大的长征，向着未知领域的伟大长征，承载着曾经出现在地球上过的 1060 亿人的共同愿望。这样未知的目标也没法找什么做个参考，那就只好参考同样未知的事物来做比较吧：死亡/死亡是人类已知的最未知的事物了。那么，人类在追求“死亡”的过程中经历了什么呢？孩童时期最真挚的梦想——想变得跟钢铁侠一样做一套自己的能飞的外骨骼；少年时期在有了基本知识后有了为所处的世界做出贡献的梦想；成年时期有了物质/知识基础之后达到功成名就的追求；老年时期追求安稳/平凡/幸福度日的追求。那在这之后呢？在那无限逼近死亡的时候，所追求的？我想大概会回到孩童时期那最真挚的梦想吧——想变得跟钢铁侠一样做一套自己的能飞的外骨骼……恐怕只有到那时候才能明白，这才是独属于自己的意义，是在不受外界影响的前提下，由自己赋予自己的意义。我们北京中学国际部 VEX 高中校队所追求的正是如此：队员渴望通过机器人学习（一个埋藏于每个 VEX 参赛队员心底最诚挚热爱的东西），制造出孩童时期仰望天空时幻想的钢铁侠一样的外骨骼。当然，每个人（个体）追求的死亡是 1600 亿人类追求的未知领域的微观体现，每个人（北京中学国际部 VEX 高中校队队员）现在和未来做出的努力（机器人学习）也是人类科技进步发展的微观体现。综上所述，北京中学国际部 VEX 高中校队所追求的童心未泯——像钢铁侠一样做一套自己的能飞的外骨骼——恰恰是雄心壮志最直白的体现。

——那未知领域的样子必定似于世界创始之初的样子。



1.2 队员培训

在北京中学国际部 VEX 高中校队的不懈努力下, VEX 机器人课程已经成为了北京中学诸多选修课之一, 每周的周一和周四进行时长 2 小时的机器人训练。北京中学国际部 VEX 课程的学员均来自于北京中学 7-11 年级的中学生, 将同时参加的 VEX 机器人比赛的初中组和高中组: 北京中学国际部 VEX 高中校队希望借此课程产生持续且高质的 VEX 机器人比赛队员以及优秀的机器人工程师。由于参与人数众多, 学员们被分为两组进行学习: 有较好机器人基础的直接进行 VEX 机器人的学习, 并专注于比赛而非基础的机器人知识, 从规则解读开始由浅入深, 一步步熟悉 VEX 比赛。另一组机器人基础较弱的, 进行基本的机器人知识学习, 并且准备低级别的机器人比赛, 在积累了足够的机器人比赛经验之后尽快投入到 VEX 机器人比赛的学习之中, 当然, 学习表现好的可以直接快进到 VEX 课程组。



(a) VEX 机器人课程



(b) 基础机器人课程

1.3 合作方式

在 24-25 赛季中, 北京中学高中组 VEX 机器人队伍主要成员有三人, 职务分别为队长兼驾驶员 (蒋博涵), 观察手 (高佑承), 计算机工程师 (马睿皓)。在进行比赛准备的时候, 首先由队长分派任务: 一般来说驾驶员会进行比赛的技术训练, 并在训练的过程中尝试各种技术要领, 在过程中寻找机器的问题, 随即对其进行调试。对于观察手来说, 一般在不进行对抗赛训练的时候着手于工程笔记的撰写或者说协助驾驶员一起进行机器的修理和搭建工作。对于计算机工程师, 主要任务就是工程笔记的撰写和机器人自动化程序的设计, 在对抗赛训练的过程中职责较轻, 负责一部分场面赛况的分析。以上为分工阶段的训练内容, 在日常训练中, 紧接着的便是全员一同参加的对抗赛训练。在这一阶段中, 北京中学高中组 VEX 机器人队伍会和其他一同参加的队伍进行模拟比赛, 队员们在此环节中熟悉规则解读, 战术部署, 队内合作: 驾驶员负责机器人的操控并练习和观察员的配合, 观察员练习制定战术, 分析场面情况并用尽量简洁且易懂的语言报告给驾驶员。在模拟比赛中, 另外一个很重要的一环就是发现机器上的问题, 驾驶员发现问题后并报告给计算机工程师。以上便是北京中学高中组 VEX 机器人队伍的基本合作方式。

1.4 队员介绍

注：排序先后顺序无任何表张。



蒋博涵 Benjamin Jiang

- 年级: 高一

- **相关经验:** 自小学开始参与机器人学习，连续多个赛季担任团队策略设计与项目协调。
- **职责与贡献:** 在 24-25 赛季中，负责操作机器人进行比赛，指导团队成员执行建造任务，并与编程/建造团队合作确保机器人高效运作。
- **兴趣与动机:** 对物理和工程充满热情，对机器人充满热情



马睿皓 Harry Ma

- 年级: 九年级

- **相关经验:** 自小学开始参与机器人学习，有过打 MakeX 比赛写程序的经验，熟练掌握多种编程语言，有耐心。
- **职责与贡献:** 在 24-25 赛季中，作为年龄最小的队员参与自动程序的编写、工程笔记的撰写与 3D 打印队牌的设计制作。
- **兴趣与动机:** 对机器人与编程都很感兴趣，希望能在 VEX 机器人比赛中更进一步，提升自我

2 赛题分析

2.1 0.1 规则

2.1.1 场地与目标

比赛在一个的 12ft. x 12 ft. 的正方形场上进行，场地内的道具包括 5 个 Mobile goal，两边的两个中立 Stake，每一方联队各一个的联队 Stake，场地中央的大型攀爬架 Ladder 还有红蓝双方包括预装各 24 个 Ring，总共 48 个 Ring。

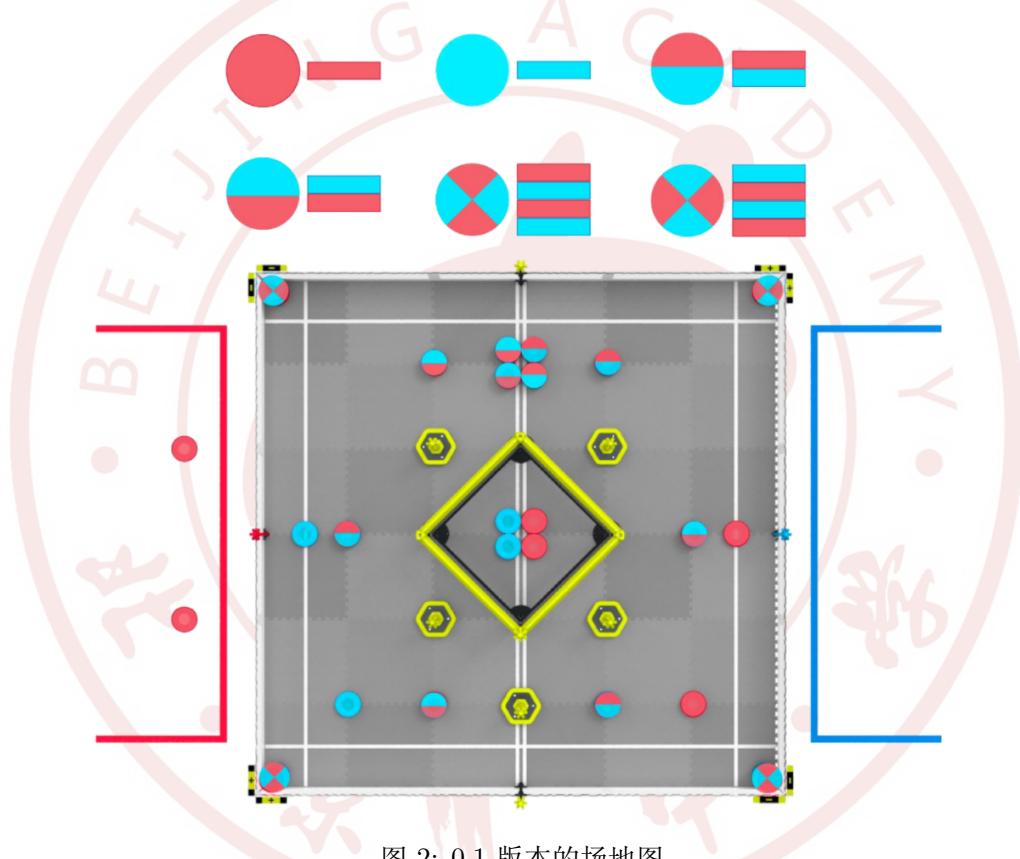


图 2: 0.1 版本的场地图

比赛分为自动阶段和手动阶段。前 15 秒为自动阶段，自动阶段规则与先前赛季没有重大改变/本质区别，重要关注点为本赛季 AWP 的判定。本赛季的 AWP 需满足以下条件：

- 至少放置 3 个记分的 Ring(联队在 Stake 上至少放置了 3 个得分 Ring，这些 Ring 必须符合得分规则)
- 至少有两个 Stake 上放置了 Ring(这两个 Stake 上至少各有一个 Ring，Ring 必须放置在不同的 Stake 上，并符合得分判定条件。并且，联队的 Stake 和中立 Stake

也算作判定范围内)

- 机器人保持在起始线后 (所有机器人在自动赛段期间都未越过场地上起始线 (起始线由白色胶带标记, 自动赛段时机器人不能越线, 否则将无法获得 AWP))
- 至少一台机器人接触 Ladder(联队中至少有一台机器人在自动赛段结束时接触到 Ladder (即比赛场地中央的结构), 但不需要完成攀爬任务, 只需简单接触。))

本赛季中, 和先前的赛季相似, AWP 的获得至关重要, 会直接影响到排名以及联队选择。同时自动的获胜分 6 分相当于两个顶套环的分值, 因此争夺自动阶段的胜利也是至关重要的。

2.1.2 套环的得分

在 0.1 规则中, 对于如何判定套环的得分, 有:

- 不接触与该 Ring 同色的机器人
- 不接触泡沫垫
- 环绕 Stake
- 符合 Stake 上限数量 (Mobile Goal Stake 为 6 个, Wall Stake 为 2 个, High Stake 为 1 个。如果超过允许的数量, 多余的 Ring 将不计分。)

同时, 对于套环的加分情况为, 一个符合上述条件的普通 Ring 被正确放置在 Stake 上时, 将为该联队获得 1 分, 可是如果该 Ring 是 Stake 上最顶部的 Ring, 它将被视为“顶套环”, 此时该 Ring 额外获得 3 分。这个分数不会与普通 Ring 的得分叠加, 而是只计算 3 分 (而不是 $1+3=4$ 分)。

对于加分区和减分区, 如果一个 Mobile Goal 被成功放置在加分区中, 那么该 Mobile Goal 上的所有得分 Ring 的分数将翻倍, 意味着普通 Ring 得分从 1 分变为 2 分, 顶套环得分从 3 分变为 6 分。如果一个 Mobile Goal 被放置在减分区, 那么该 Mobile Goal 上的所有得分 Ring 将不计分, 并且相应方的得分也会相应减少: 普通 Ring 的得分将减少 1 分, 顶套环的得分将减少 3 分。因此, 基于这个规则, 我们初步判断, 最后赛队间争抢得分的战略会集中注意在争抢加分区上。实际上对于减分区的打法是有风险的, 因为要想要发挥减分区的作用需要打对方联队颜色的 Ring, 可是如果出现失误, 需要承担的风险是对方联队相应分数的 Ring 的得分。对于加减分区的是否生效的判定, Mobile Goal 需接触加减分区的边界 (即地面或标记加分区的白色胶带线)。保持直立状态, 即 Mobile Goal 的 Stake 与地面或围栏没有接触。与此同时机器人可以与 Mobile Goal 接触, 但这不影响得分的计算。

当然还有一些其他的比较细微可是在场上却能发挥至关重要的作用的一些其他注意事项。首先是当 Ring 得分时, Mobile Goal 不必保持直立状态。只要 Ring 符合得分条件 (不接触机器人、不接触泡沫垫, 并且环绕 Stake), 即使 Mobile Goal 倒下或处于倾斜状态, Ring 仍然可以得分。

在场上额外需要小心的是“双持”的规则, 规则中描述的是如果机器人的方向改变会导致得分道具受控运动, 则视这台机器人持有该得分道具。。这句话的判定满足任意一项:

1. 得分道具完全被机器人支撑
2. 机器人利用其凹面 (或在多个机构/面形成的凹角内), 沿首选方向移动得分道具
3. 机器人将得分道具固定在地板或场地要素上

注释: 对于 Ladder 的攀爬功能, 由于其技术难度, 在赛季初的机型设计中我们并不考虑, 可是在后期, 由于其巨额得分, 这是一个很值得考虑的技术方面。但还是, 在初期的策略设计中不会对于攀爬进行任何考量。

2.1.3 机器人的设计要求与规则

机器人的合规起始尺寸要求是 18" x 18"，即在折叠状态下机器人的尺寸需要在此尺寸以内。但是，机器人可以且只能沿着一个 x/y 方向展开构型，总占地面积不能超过 24" x 18"，即向选定方向展开至 24"。除了水平方向之外，机器人也可以向垂直方向展开，但是不能超过两层以上的 Ladder，即 32"。

2.2 0.2 规则

相较于 0.1 规则，0.2 规则只是进行了一些微小的改动和细化，细化重申了一些对于如 AWP、预装的规则描述。因此，在综合考虑了 0.2 版规则的改动后，并没有直接对我们的机型设计和场上打法策略产生直接的影响和改动。



图 3: 1.0 版本的竞赛手册

2.3 1.0 规则

2.3.1 场地变动

1.0 规则对于 0.2 规则的改动可以说是巨大的。首先，最为直观的是场地布局的改变。**加分区和减分区由原来的对角排布改变为了在一边的平行排布。**新版的场地图如图所示：

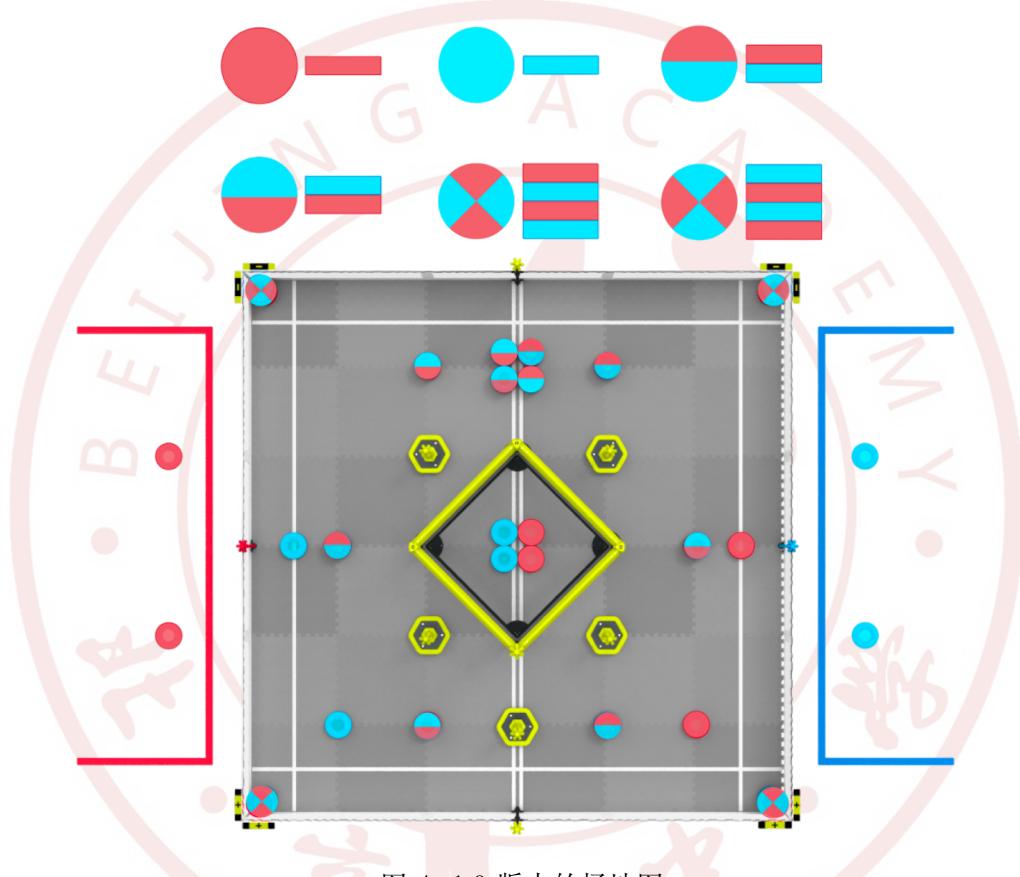


图 4: 1.0 版本的场地图

这一次场地的改动对于打法的影响可以说是又大又不太大。在面对这个改动时，主要需要改变的是操作手的主观意志。因此，我们相信多加练习后即可适应该改变。

2.3.2 攀爬加分

这一项规则引入了一个新的加分机制。根据 <SC9>，如果一支队伍在比赛结束时有一个环在高桩上得分，每次攀爬可额外获得 2 分。这表明着对于大部分机器已有的 1 级攀爬，如果条件成立，得分将由原来的 3 分变成 $3+2=5$ 分。而且在高桩上的套环也

有 3 分，如果可以做成这样的车型，将会在场上拥有极大的优势。除此之外，想要达成这个目标，机器人也需要有 3 级攀爬的能力。这就说明着，单一机器人在完成这个任务后最少可以获得： $3 + 12 + 2 = 17$ 分。

奈何，在赛季初期，我们这支队伍由于时间和技术难度的问题，暂不考虑 3 级攀爬 + 高桩套环功能的实现。但我们仍旧保留对这一可能性的认可，可是实现这一功能也意味着对于其他已有功能的放弃，在经过细致的考虑和研发后，在后期也许会考虑这个功能的实现。

2.3.3 AWP 改动

规则 <SC8> 进行了修订，澄清了赢得 AWP 的要求，现在需要至少有三个盟队颜色的环得分，且至少有两个桩在盟队一侧，其中至少有一个环得分。实际上，这一条规则的修订对我们车的影响不大，鉴于前期我们已经研发出了单车实现 AWP 的自动程序。

2.3.4 保护区时间

在比赛结束时为加分区引入了 10 秒保护规则，防止在这段时间内与这些区域的移动目标进行互动。在面对这一项规则是要谨慎的，因为 10 秒内进入加分区或者移动加分区内的道具都会导致 DQ，因此在 10 秒左右，比如 11 秒时开始，需要尽可能地远离加分区，谨慎被 DQ。这一个规则实际上对比赛的走势有影响，因为 10 秒内空出的时间可以给队伍更多的选择，比如将对方的底座拖入减分区、进行边桩占领的争夺、高挂等。实际的策略决定和执行视场上情况二调整和决定。

2.3.5 其他调整

除了上文描述的规则改动，对几个规则也进行了调整和增补，以更好地澄清游戏意图并确保参与者的公平竞争。这包括对各种规则的更新和引入新的图示等。这些规则的改动对我们队伍并没有任何实质性的影响，因此在这个部分并不会列出。

2.4 2.0 规则

2.0 版本的规则相较于前几个版本是改动最大、对于场上比赛表现影响最重的一次。2.0 规则的改动在场上的很多方面和判定都有体现。以下是 2.0 规则对我们产生重大影响的几点改动：

2.4.1 场地变动

2.0 版本的场地图如下图所示：

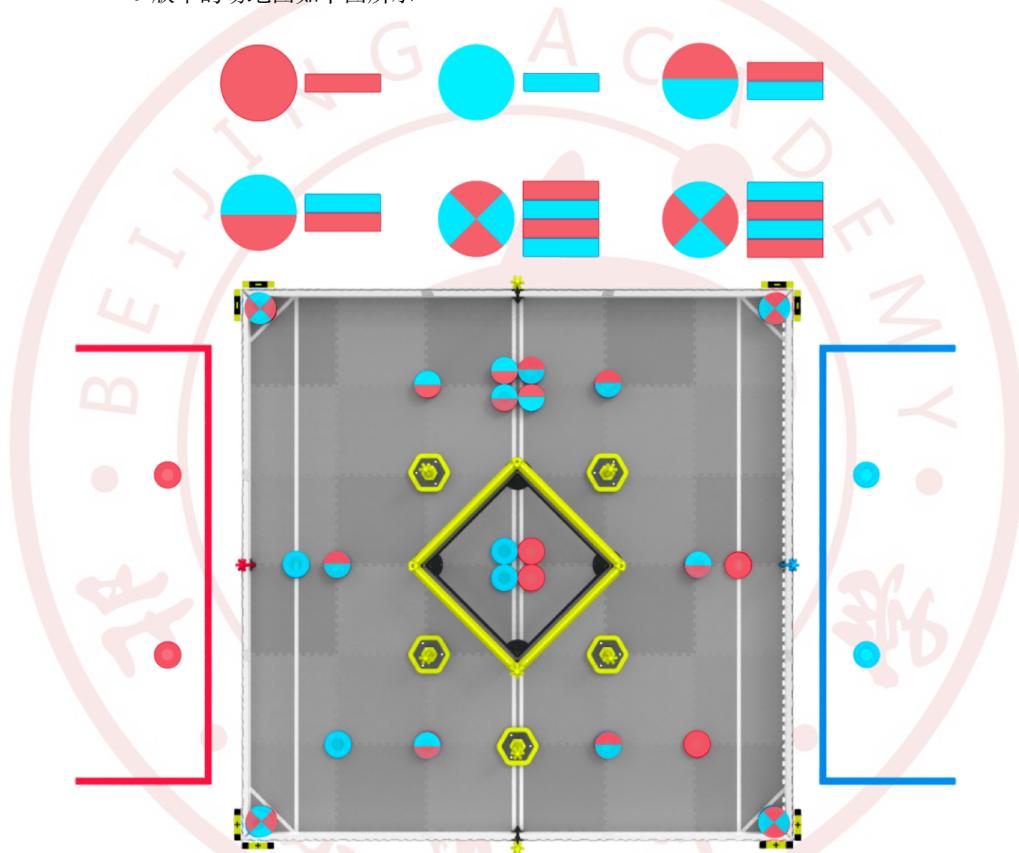


图 5: 2.0 版本的场地图

相较于 1.0 版本的场地，2.0 规则的场地的主要改动是加分区与减分区初始套环摆放。在 2.0 规则中，对于蓝方的加分区减分区的两摞套环的最上方的套环都是红色的，相反，对于红方的加分区减分区的两摞套环的最上方的套环都是蓝色的。这一个改动对我们产生了较大的影响，由于我们一开始的自动程序会拾取一摞套环中与联队同色的最顶部套环，改动摆放位置后我们不得不放弃了这一自动程序的思路。同时，加分区和减分区的白色胶带也由原来的正方形改为了三角形，减少了加分区与减分区的面积。

2.4.2 环的得分

2.0 版本规则改变了对于环的得分状态的判定。2.0 规则对 $\langle SC3 \rangle$ 、 $\langle SC5 \rangle$ 进行了重写，即 Ring 接触与该 Ring 同色的连队机器人或者地垫仍可以在 Stake 得分。简而言之，在 Mobile Goal 上面的套环不论是与机器人接触又或者是与地面接触都可以算作得分。这一项的改动至关重要，从此往后对于机器人的操作可以更加宽松，改变了前些版本比赛结束前放下夹住的 Mobile Goal 的习惯，而且放倒对方的 Mobile Goal 的动作也将失去作用。而且，即使套环触地也不会影响得分情况的判定。

对于加分区和减分区的判定也出现了相关改动。移动桩顶部的伞帽高出场地围边的上沿且桩侵入了区的平面即可算作放置在加分区、减分区区内。简而言之，**判定为使用垂直投影决定是否进入加减分区的边界**。此外，新的 $\langle SC5 \rangle$ 对多个移动桩放置在同一个区内的情况进行了新的判定，即：

1. 底座延伸至区内最远的移动桩
2. 尖桩最垂直于地面的移动桩。
3. 顶部的伞帽延伸至区内最远的移动桩。
4. 如果以上标准都不能打破平局，那么所有移动桩都不被视为放置在区内。

此规则虽然额外给予了一定的得分空间，但仍再次强调了保护加分区内移动桩的重要性。若己方放置的移动桩被对方移开或对方也放置了自己的移动桩但满足上面四条规则，则己方的竞赛分数将会受到极大的影响。因此待己方机器人将一摞套环置入得分区后，联盟中的两台机器人有一台机器人负责时刻注意并保护己方的移动桩的行为是有必要的。

2.4.3 保护区时间

在这一规则下，保护区的时间由 10 秒变动为了 **15 秒**。这一改动赋予了双方队伍机器人更多的时间去专注于 Wall Stake、高挂、减分区等动作，实际上增加了比赛中不同策略的决定，一定程度上加大了操作手与队友沟通、自己看场上环境而决策的难度。

2.4.4 AWP 改动

2.0 规则下又一次对 AWP 进行了改动。但实际上这一次改动是比较微小的，即对于有世锦赛名额的比赛，需要有 4 个得分套环，并且占领 3 个得分杆，而且至少有一个环在联队的边桩上。这一个变动更加考验了队友间的配合，客观上增大了世锦赛名额比赛的激烈程度和世锦赛名额的难度。

2.5 2.1 规则

相较于 2.0 规则，2.1 规则只是进行了一些微小的改动和细化，细化重申了一些对于如 AWP、预装的规则、“攀爬”、 $\langle SC3a \rangle$ 中的“环绕的”套环概念的描述。比较重要的是新增的套环概念，即在多个尖桩被一个套环套住的情况下，将不会出现顶套环的 +3 的得分。同时，在区内的尖桩将不受加减分区的影响。

2.6 技能赛

时间分为 1 分钟自动阶段和 1 分钟手动阶段，每个阶段重新恢复场地，且初始位置不同。技能赛阶段只能代表红方发车，选手站在红方站位。发车位置与常规赛成绩一样，只有一个预装环。值得注意的是，蓝方的联队边桩也可以得分。比较重要的是 $\langle RSC4 \rangle$ 规则，规定蓝套环只能作为 Top Ring 而且只能在所有红色套环得分后才可以在 $\langle RSC4 \rangle b, c, d$ 规则上计算得分。根据 $\langle RSC5 \rangle$ 规则，蓝色套环上方的红色套环不得分。因此，在初期，可以不用考虑蓝色套环的得分，并且可以把带有蓝色套环的 mobile goal 推入角落，以免影响得分。



3 机型迭代

3.1 0.1 版

0.1 版本我们采用了“夹子车”的设计，先采用简单的结构以测试性能。由于是初次进行该项目的竞赛，我们使用这样的设计作为模版，先实现一些基础的功能，方便后面对速度与精准度的相关优化。该版本机器人由于是初期设计，并没有配备相关对于移动桩的加持装置，但是预留了相关的空间，方便后面进行移动桩夹持装置的设计与安装。设计该机器人的主要目的是为了能够测试机器人的移动效果以及方便后期升级。0.1 版的夹子车如图所示，功能单一，是日后继续研发的思路的雏形。



图 6: 0.1 版本的机器

3.2 1.0 版

3.2.1 总体概要

1.0 版本我们对 0.1 版本进行了重大翻新与改进。我们在机器人一边增加了移动桩加持装置，方便对移动桩进行操作；在另一边增加了套环的吸收装置与一个可上下移动的机械臂。我们使用链条配合塑料打印的钩子以抓取套环放置到移动桩上，而机械臂的顶端同时用作套环的吸入装置，通过橡胶轮的摩擦将套环放吸收到钩子上。同时机械臂也负责将套环放置在边桩上。在吸入装置将套环吸入后再反转链条，套环便可以到达机械臂顶部。随后机器人移动到边桩后便可以抬起机械臂将套环套在边桩上，此时再进行后退便可以将套环套入边桩。

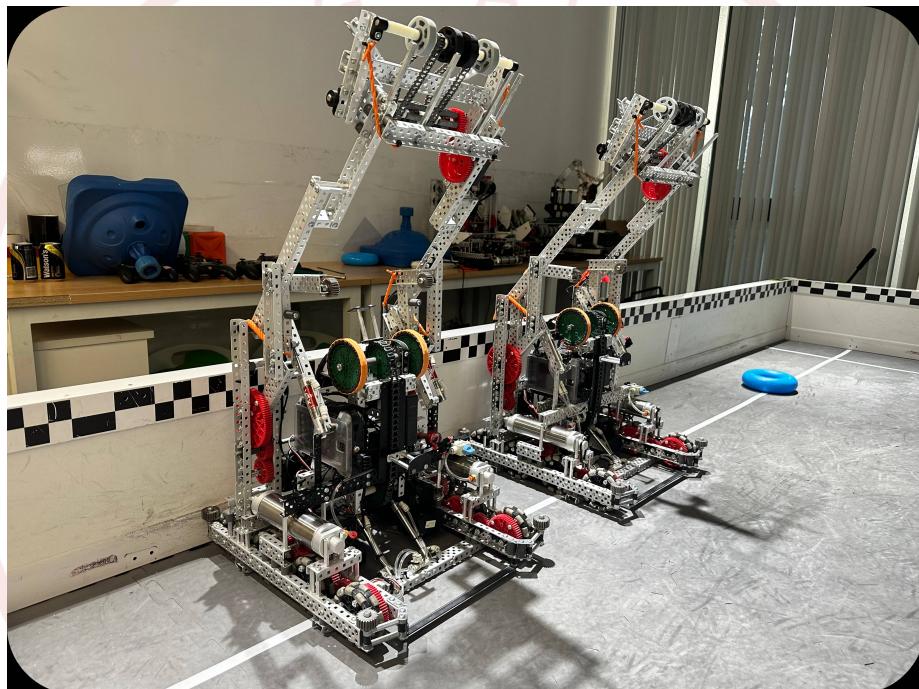
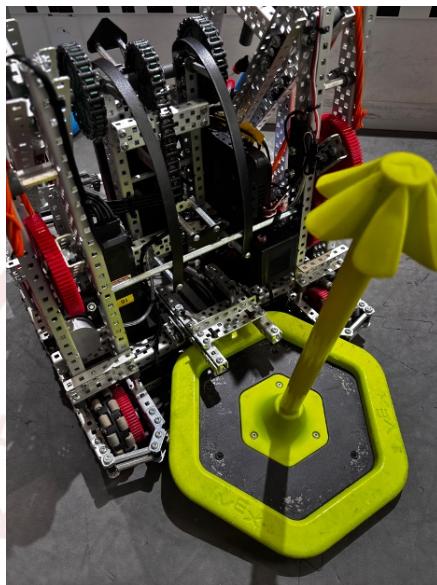


图 7: 1.0 版本的机器

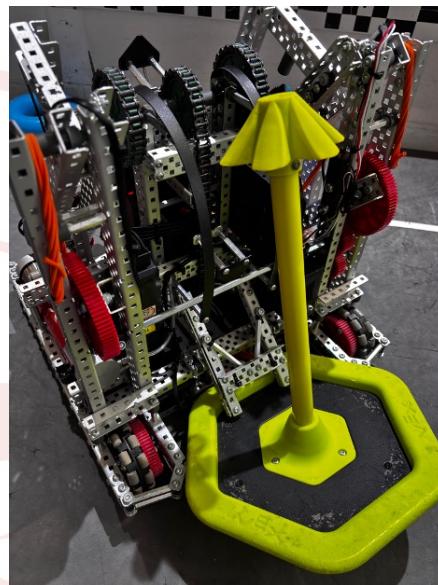
1.0 版本的机器的一大特点就是“沉”和“有劲”。六个蓝色电机驱动的底盘，10 余公斤的质量，在场上没有任何一个机器可以顶得动我们 1.0 的车，这也是为什么即使最新版本的车辆已经投入使用，1.0 版本系列的机器仍然能在首发阵容中有着一席之地。在我们参加的第一次比赛也就是北京延庆的比赛，用的就是这一台车参赛的。最终在延庆赛中获得了亚军的成绩，这一个“功勋老将”1.0 车在随后也迎来了升级，可其本身也是功不可没的。

3.2.2 技术细节

首先是对于 1.0 代车的初代夹子的介绍:



1.0 夹子放开情况



1.0 夹子锁住情况

对比上方两图，展现了 1.0 机型的夹子的工作原理。1.0 机型的夹子结构十分简单，核心就是两条钢，末端被气缸展开控制向上，前端因此被驱动向下夹住底座。当然，为了很好地夹住底座到一个比较合适的角度，需要进行很多的微调，在调整后应用连接两边底盘的钢进行限位，达到最终的效果。

但是这一代的夹子有很多现存的问题，最主要的问题（也是最影响我们的问题）就是对于底座的夹取需要很准确的对准。为了可以很好的上环，夹子必须对准底座的平面的正中央，如果夹取了平面的靠侧边或者是底座的角，这个夹子的设计就会导致底座的夹歪，因此无法正常上环。

如果在场上遇到了这个问题，首先心态会很焦急受到影响，影响接下来的操作，而且需要把底座放下、重新对准。在急迫的心态条件下，（至少从我们队的操作上）经常会在第二次尝试夹取中继续失败，因为内心会放大对于时间的敏感度，就想尽可能地更快速地重新夹取，导致时间上的巨大损失和心态的崩溃。

这一代的车参加了在延庆举办的北京城市赛，在过程中即使在练习下有了熟练的操作，还是因为夹子设计考虑的不周全出现了一些失误，这更加坚定了我们队对于赛后改变夹子的需求和想法。

3.3 1.1 版 (1.0 改)

3.3.1 总体概要

相较于 1.0 版本，我们发现机器人在机械臂与吸环装置等方面有一些欠缺。于是在 1.1 版本的微小升级中，我们简单改进了抬臂的电机和齿轮比，使得抬臂的速度更快。可随着速度的加快，抬臂的力道也会有显著的降低，因此相比会更加的难以控制。我们对底盘的速度也进行了加快，原因是减重减掉了部分用处不大的结构件和钢材。同时，在 1.1 版中改进最大的是车辆后部的夹取 Mobile Goal 的夹子，我们又原来的一块钢被气压驱动压紧底座的设计变为了有两个气阀的反锁结构，这一结构的设计大大提高了对于底座的夹取的成功率，是一个重大的改进。

总的来说，1.1 版车并没有参加任何比赛，只是一个过渡的车型，在随后也被 2.0 新版的机器完全取代。1.1 车型是老车型（包含一个巨大抬臂的车型）的最后一个版本。下图是 1.1 版车的样式：

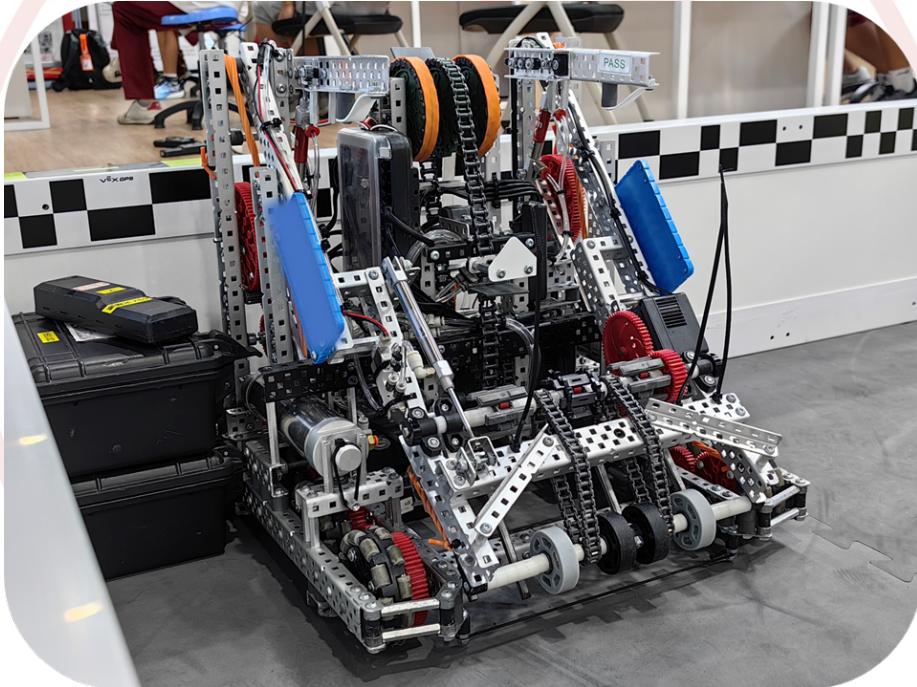


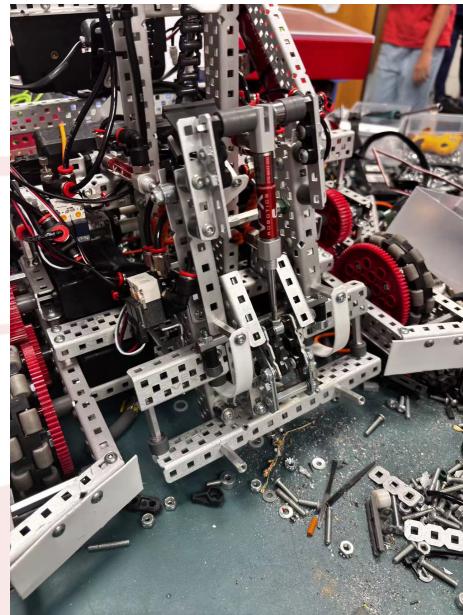
图 8: 1.1 版本的机器

3.3.2 技术细节

从 1.1 代车开始，我们的车型设计采用了双气阀的反锁结构，顾名思义“反锁”可以更牢固地锁住底座，这一点可以在比如说自动程序的抢中立底座阶段比其他队伍更有优势。

细分这个结构，主要为两个功能：“夹住”和“抬起”。

- “夹住”的功能是用一个气阀控制的，气阀带动两根钢，每根钢的尽头都有一个自锁螺母以更好地夹住底座。当夹子的气阀展开到最大角度，夹子就可以进入反锁的结构，在这一角度下，如果气缸不主动控制杆缩回，单靠气缸的杆的切向力是无法解开锁的。
- “抬起”的功能是由两个气缸控制的，并且有皮筋的力量加持，可以让其有更大的力道来抬起底座。在夹子的后方也有钢作为限位，使夹子可以把底座抬起到特定适当的角度，使得吸子可以把套环顺利地套入到底座当中。



双气阀反锁结构

其次是吸子的改动，在先前的版本中，吸子设计的最大高度容量只可以放 1 个环进去，因此在一摞两个的环中，前代车只能吸进下方的环，但是在每一方中间的那一摞环的上方的环是和联队同色的，因此有需求做出可以吸入上方套环的结构设计。在延庆的城市赛，看到其他队伍类似的设计，导致我们队终于意识到了该结构的重要性。在此之下，可以变角的吸子的设计诞生了。如下图所示：

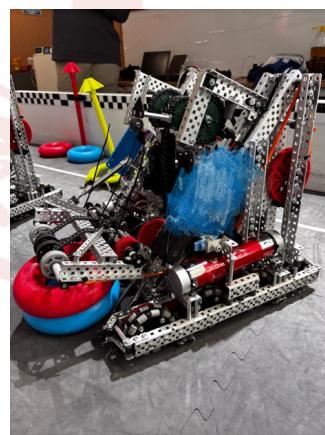
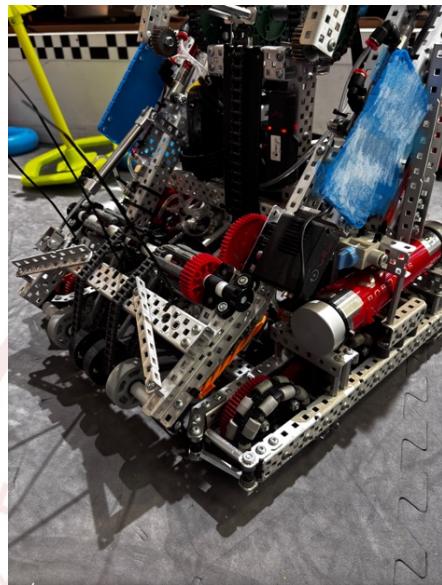
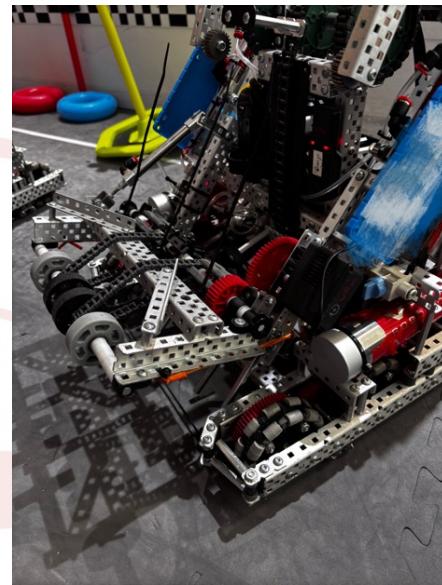


图 9: 可变角吸子

如图所示，这一版本的吸子可以吸入上方的套环，因为加入了一个气缸，控制吸子的角度变化。在下方的两个对比图中很直观地体现了可变角吸子的功能：



1.1 可变角吸子下方档



1.1 可变角吸子上方档

除了气缸的控制，这一项功能的另一核心是其内部的链条结构，模仿了自行车变速的设计，使得升高的吸子带来更短的直线距离并不会导致链条的松弛。如下图所示：



图 10: 可变角吸子的链条结构

套环的上升运输结构

接下来描述的是 1.1 代车对于套环向上运输的方式，大体来讲和 1.0 代车的方式一样，链条驱动螺母柱，螺母柱前端用两个螺丝固定连接激光切割的 PP 材质三角形的拨片，每个拨片下方都有一个垫片，以将套环顶在下方垫片运送到上方套在夹住的底座中，具体的样式如下图所示：

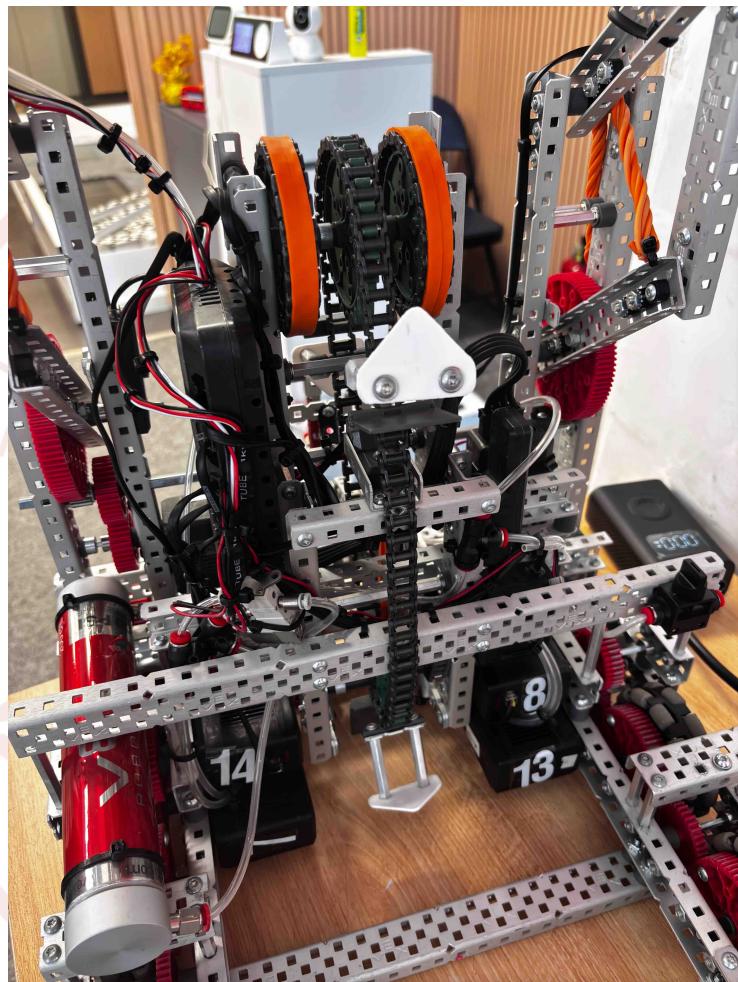


图 11: 1.1 版机器的抬升结构设计

在后续的测试当中，这一版本的吸子对于套环的吸收效率很高而且效果和稳定性相对较高。可是，在队内练习赛中也显露了一个相当严重的问题：具体来讲就是在高挂杆下方进行吸环时和在小概率卡环的时候链会断。其实我们在设计时也想到了，在这个结构上使用履带其实是一个更好的选择，当时因为从材料数量角度考虑没有使用履带，当时一定程度上忽视了断链的可能性，这使得我们在后续的结构调整加入了优化链条的考虑。

底盘的设计

我们底盘的设计为六电机底盘，每个底盘使用 6 个蓝色的电机。两边的底盘一边 3 个电机，分为前、后、上电机。每边底盘 4 个轮子，总共为 8 个轮子，轮子间由齿轮传动。具体的排布由下图所示：

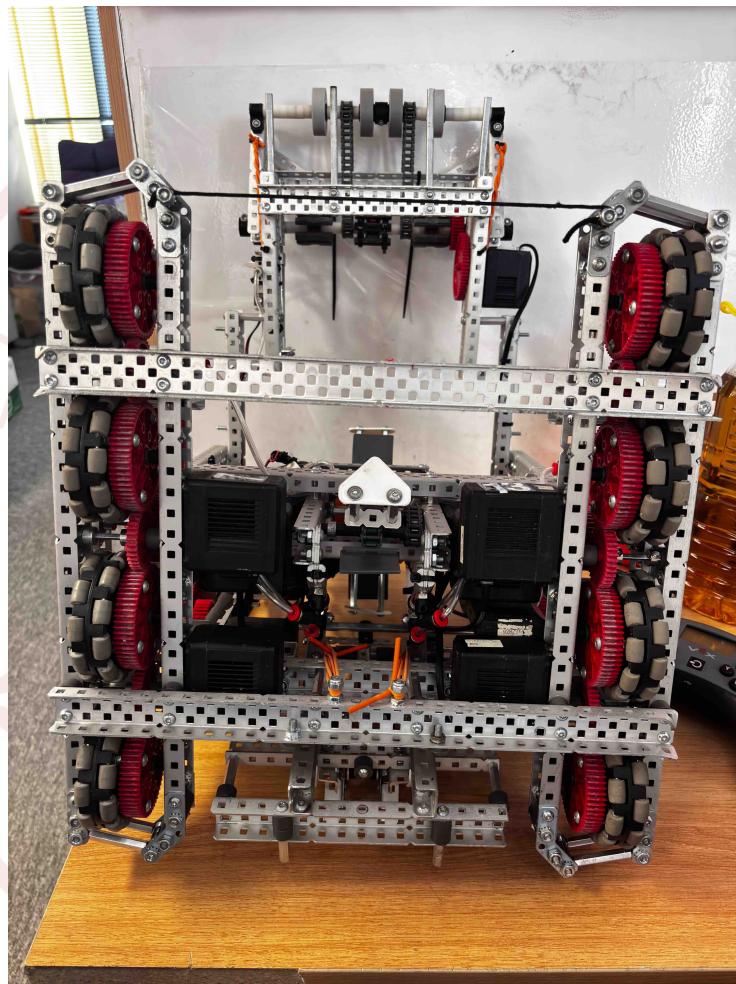


图 12: 1.1 版机器的底盘设计

当然，这个底盘也有相应的问题，虽然也和自身重量有关，这个底盘的齿轮比和每个电机内部没有添加润滑油使得这个底盘的移动速度相对较慢，因此在场上的底座的争夺中和打套环的过程中会有相当的劣势，因此这个版本的底盘也有后续的改进需求。

3.4 2.0 版

3.4.1 总体概要

2.0 版本的新车改动相比 1.1 可谓是巨大的。2.0 的车的设计完全舍去了车前方的巨大抬臂和车的高挂功能。2.0 也是由 6 个电机驱动的底盘，相比 1.1 车，由于重量的极大降低速度有了一个质的飞升。而且由于新的车前吸子架构，2.0 车对于套环的吸收相比前版本的车也有极其巨大的提升。在先前的测试中，甚至可以出现原地把环吸进上升的轨道的经历。还有一点的改动就是在吸子上方加入了气阀可以使吸子向上抬起一段距离，从而可以使我们在自动阶段吸收已方放置于对方套环上方的第二层套环。2.0 车的设计也改变了挂高杆的方式，我们研发了一个半自动矫正系统，车上方有两个抬臂可以将套环正好卡在抬臂上，然后抬臂向下压到高杆，即可把套环套进两边的杆中。下图是最初版的 2.0 的车的设计：

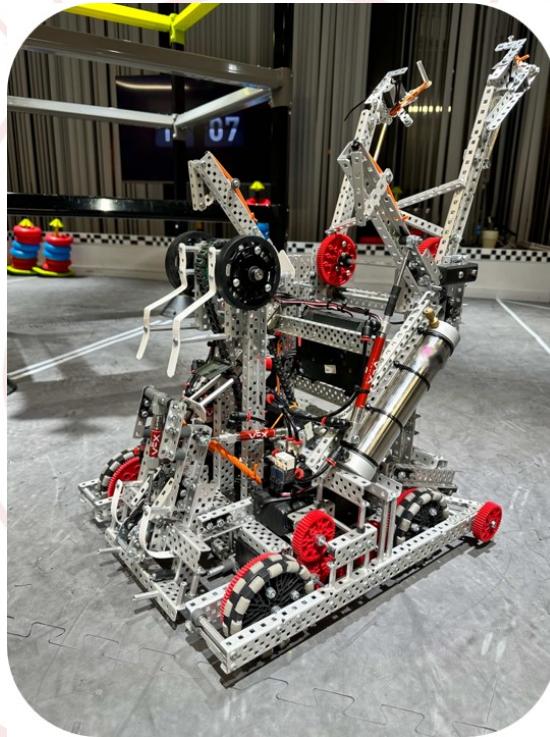


图 13: 2.0 版本的机器-1

当然，在 2.0 车基础上的后续改进也有不少，比如将高挂的功能取消掉，同时也拆除为了高挂的后方导轮，这也减轻了这辆车的一些重量。我们分析，在场上实际上高挂的机会并不是很多，所以干脆省去了高挂的功能，因为对于 2.0 版的车的设计来讲，在操作熟练后，高挂的时间足以占领一个侧边的高杆，分值是相等的，再加上这个赛季对于高挂保护的不重视，高挂的风险较高，于是决定彻底取消高挂在我队伍的车上的安装。

3.4.2 技术细节

整体设计

2.0 版本车的底盘由 6 电机 6 轮的布局改为了 6 电机 4 轮的设计，这大大加快了底盘的运行速度，速度的基础对我们机型在打法的提升起到了很大的作用。

其次，我们的车型取消了抬臂的结构，吸子放在车的前方并且结合了可变角的吸子。我们的车上加上了“摇臂”的设计，因为我们坚信摇臂在经过熟练的练习后，套边桩的速度一定会比先前的抬臂车快出一个档。下图是 2.0 车的整体布局图。

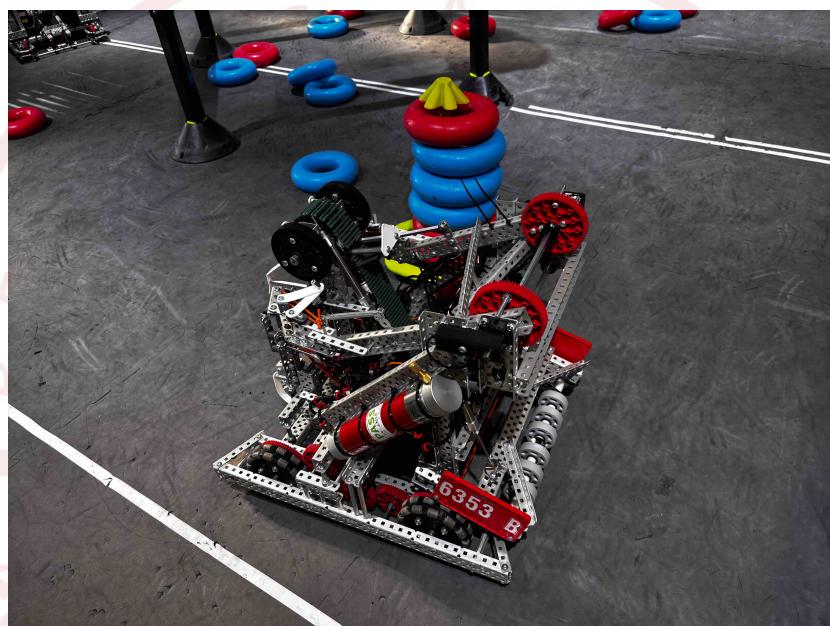


图 14: 2.0 版本的机器-2

2.0 的机型在操作方面有整体巨幅的提高。而且与 1.0 系的机型不同，2.0 作为一个轻量的平台自身存在很多的优化空间：

- 清角的优化
- 筛环
- 夹子的测距传感器
- 摆臂牢固性的持续升级
- 夹子结构的持续优化
- ...

挂杆摇臂

为了追求更高的效率，挂杆的方法由原先的抬臂变成了摇臂。摇臂引入了一套半自动程序，就是当 L1 和 L2 按键被同时按下后，摇臂会通过角度传感器锁定一个倾角。在锁定角度后，当吸子吸入套环后将通过摇臂上的防滑结构（防滑帆布 +PP 材料 + 皮筋）使得套环可以卡在摇臂上。摇臂通过特定角度和长度的设计，摇下后可以准确地套在高杆上。

具体结构如下图所示：

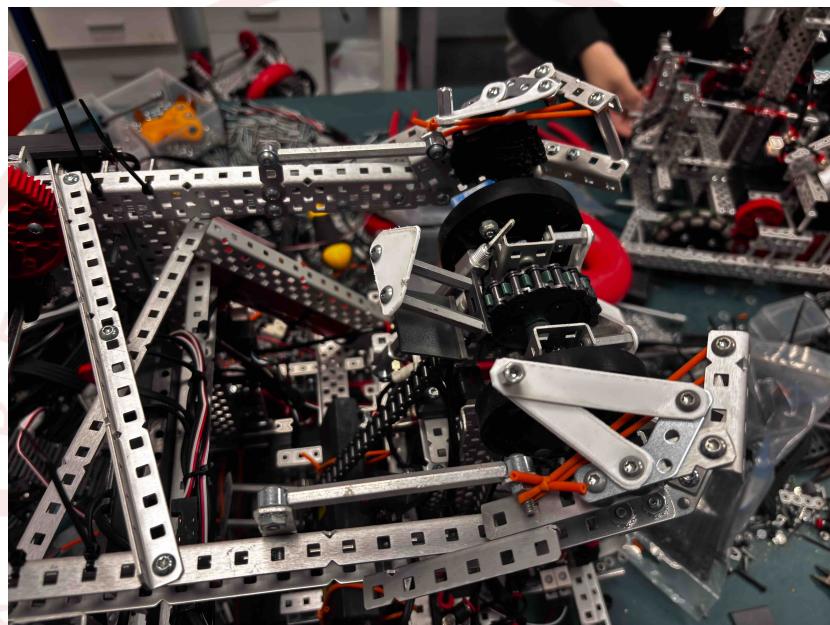


图 15: 摆臂设计细节图

在改为摇臂的设计后，我们曾经做了一项测试，就是场上随机散落套环，没有干扰，看需要多久可以在一个高杆上套满 6 个本方颜色的套环。对于 1.0 系的老车型，完成这项测试普遍需要 50 秒至 1 分钟，可是对于摇臂的设计，最快的一次一共只花了 26 秒便完成了 6 个套环对于高挂杆的套。

筛环

其实对于很多队伍，在赛季初期为了避免失误在手动和自动阶段都加入了筛环的结构和程序。我们车型在之前都没有考虑加入筛环，在目前的版本加入了手动阶段的手动开启的筛环还有自动阶段的自动筛环。筛环的结构如下，简而言之就是一个气缸和颜色传感器控制，再控制气缸旋钮的松紧以调整气缸收回的速率以完成筛环功能。

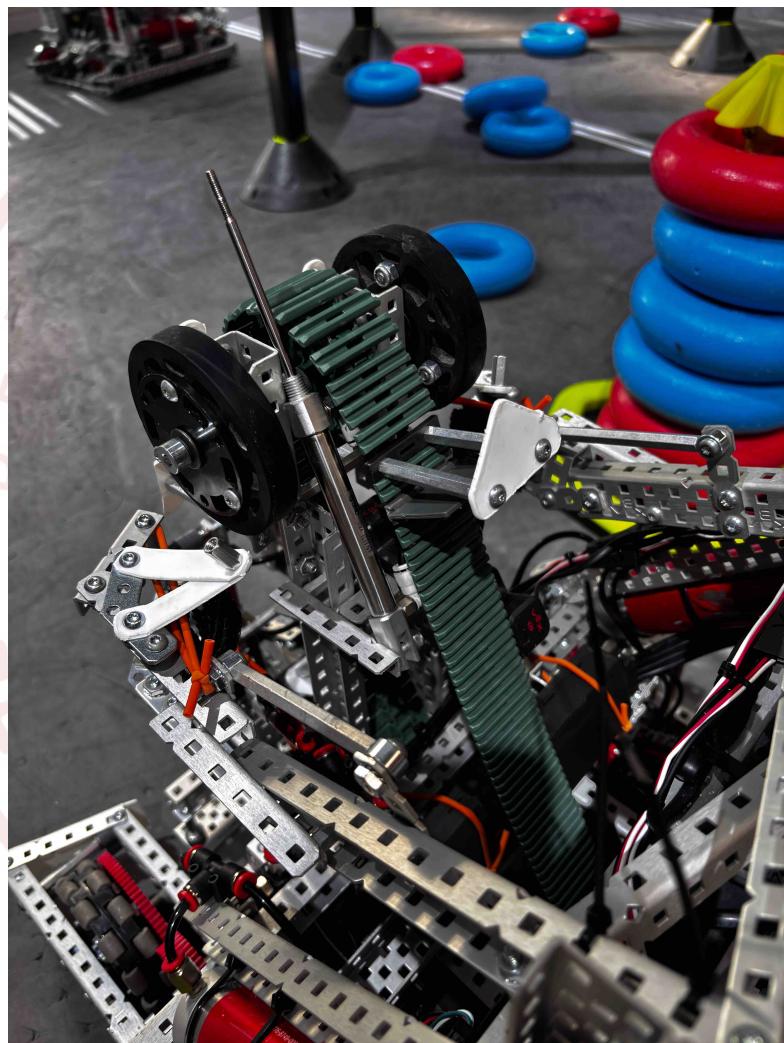


图 16: 篩环设计细节图

以下是筛环程序在手动程序中的代码:

```
1 int SelectColor()
2 {
3     while(1)
4     {
5         // IF(吸子COLOR.HUE() < 25 OR 吸子COLOR.HUE() > 350)
6         //
7         //     HANG.SET(1);
8         //
9         // ELSE IF(吸子COLOR.HUE() > 25 AND 吸子COLOR.HUE() < 350)
10        //
11        //     HANG.SET(0);
12        // } //筛红环
13
14        if(吸子Color.hue() > 180 and 吸子Color.hue() < 240)
15        {
16            Hang.set(1);
17        }
18        else if(吸子Color.hue() < 180 or 吸子Color.hue() > 240)
19        {
20            Hang.set(0);
21        } //筛蓝环
22    }
23 }
```

3.5 3.0 版

3.5.1 总体概要

本车型大致可以分为五个主要子部分：底盘，进环，上环，夹子，摇臂；五个部分以轻便，机动，效率为行动纲领，和心协力为争取在赛场上拿下更多的分数。全车功率 **88 瓦**，其中 66 瓦分布给底盘；11 瓦分布给摇臂；11 瓦分布给进环和上环的联动系统。至于电机的选择，底盘上使用 **600 转的蓝色电机**，摇臂上用了转速较低但扭矩较大的 **100 转的红色电机**。底盘上的齿轮比为 **4:5**，摇臂上的齿轮比为 **1:2** 同时，本车还备有 **3 个电磁阀**和一个气缸。传感器有吸环系统上的光学（颜色）传感器；摇臂上的角度传感器；全场定位上的定位传感器；车身上用于定位的陀螺仪。全车进行了气动布局的优化，使单气缸的布局在比赛中足以发挥优势。另外地，**3.0 版本也是一个过渡版本**，未参加任何比赛，并在数周后进行大改成为 **3.5 版本**。

3.5.2 底盘

本车有四轮的联动驱动，使车子成为四驱，在运动中有更好的稳定性而使用的万向轮通过减小摩擦力也提升了车子的机动性和灵活性，相反地，底盘侧向的稳定性会大打折扣，这也是为了追求灵活而对稳定性难免做出的牺牲。底盘使用 600 转的蓝色齿轮箱电机来为轮子提供动力；但是，这 600 转的动力不能完全使用在轮子上，必须要有一些减速装置。于是，我们选择了在电机上连接 48T 的齿轮，在万向轮上连接 60T 的齿轮，形成 **4:5** 的减速比。相较于传统的 **3:5** 减速比，可见这次的设计对于速度和机动的额外追求。

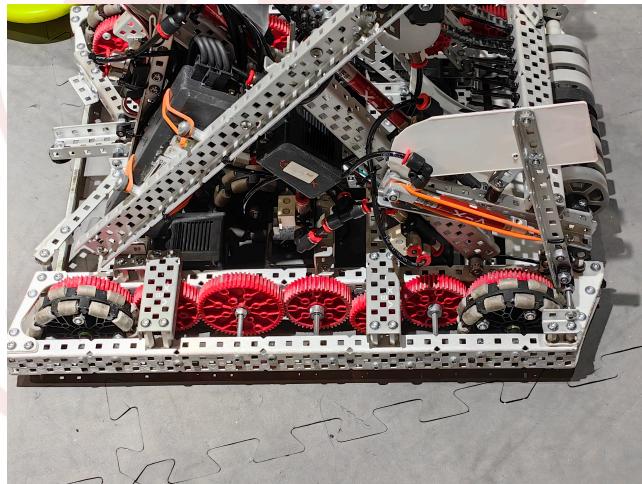


图 17: 3.0 底盘传动设计

3.5.3 进环 & 上环

这是由进环和上环共同组成的联动结构，主要目的是把套环运送到夹在后方的桩上。运输过程由二者之间的履带完成。其中的进环系统可以用气缸将其抬至二层，来完成自

动环节对二层环的吸取工作。本气缸只连接单个气管，也就是说气缸只能使其抬高，下落则由其自身重力完成，这是为了给吸环提供足够的空间。在上环附近的履带上安置的有光线传感器，可以感知传输上来的环的颜色，并通过短时间停止传送带将异色环排出。



图 18: 3.0 版吸子设计

3.5.4 夹子

本车的夹子由气缸驱动，整体结构可以总结为“两个三角形”。第一个三角形是气缸，底盘，夹子组成的三角形，进气后会锁死成具有稳定性的三角形，这在争抢底座时十分有利。第二个三角形时底盘在底座位置组成的三角形，可以在运动中将底座调整为正确角度，大大提升了夹底座和上环的稳定性，同时底盘上的粗轴也会配合着夹紧底座。

3.5.5 摆臂

揆臂的主要目的就是套边桩，不需要过高的转速。于是，揆臂的电机使用了 100 转的红色齿轮箱电机，并配上 24t 和 48t 的齿轮形成 1:2 的减速比，这是为了对操纵有更好的控制。

3.5.6 全场定位

全场定位是本车的一大亮点，位于底盘的正中央，由两根皮筋使其接触地面。接触地面的两个万向轮一竖一横，对应了判断位置的 x 和 y 轴。在车辆行驶中，这两个轮子被动而转，而轮子连接的两个角度传感器正好接受了转动的信息并将其记录下来。这些信息有什么用处呢？就实际情况而看，它可以记录一些预计转动移动与实际转动移动上的误差，最后换算出车辆在场上的具体位置，完成全场定位。

* 注：由于后期低挂的增加导致接触地面的两个外向轮在悬挂状态下仍接地，致使悬挂不得分，为了使车辆悬挂得分，全场定位于 3.5 版本停用。

3.5.7 细节

本车的摇臂上有一个用绳子控制随摇臂转动而收缩的装置：当摇臂收回时，装置伸出，辅助套环运输到上环系统；当摇臂上升过程中，装置回收，使套环固定在摇臂上；当摇臂伸长到一定高度准备套边桩时，装置会再次伸出，对套环施加一个从后方的力，将其压入边桩。



图 19: 特别设计的摇臂

3.6 3.5 版

在前期的自动手动测试中，3.5 车辆在吸子与筛环方面仍存在重大缺陷，因此我们大改了吸子部分以及筛环部分，小改了夹持部分，构成了 3.5 版。

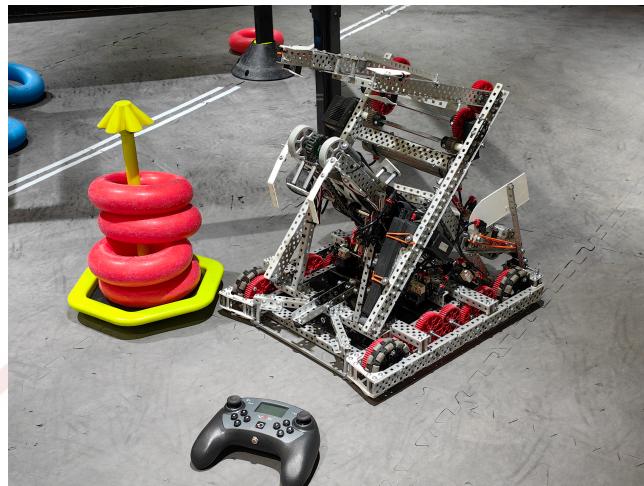


图 20: 3.5 车型外观

3.6.1 小改后的夹持

夹持装置方面我们进行了小改，拆除了原来的扎带限位，增加了两个限位 L 形钢片使得夹持移动桩的时候可以保持其竖直的状态，避免移动桩歪掉致使套环无法打进。此外，夹持的可行范围也来到了几乎全角度，增加了比赛时的稳定性。

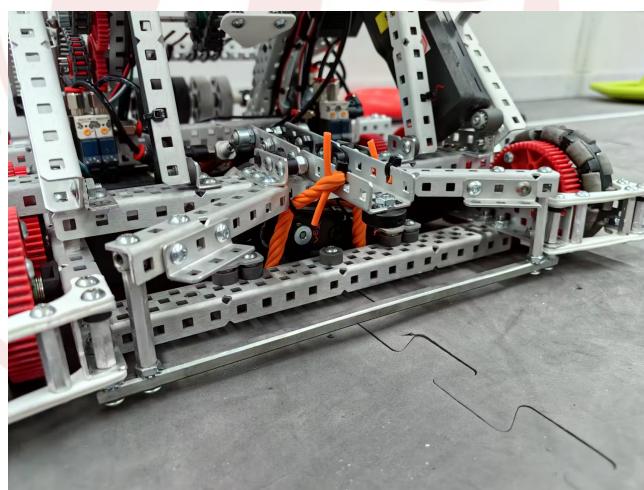


图 21: 3.5 小改后的夹持

3.6.2 新吸子

新吸子在原 3.0 版的基础上大幅增强了角直吸的能力。我们改进了中央的结构，使吸环部分由 6 个较软的白轮 +2 个较硬的黑轮混合组成，增加了其对角区的适应性。此外，我们还增大了下面的底板使其可以将环直接“铲”入吸子。这样方式不仅增加了效率，而且可以通过倒退吸子的方式插入环堆底部将最底下的本方环拖出吸入。另外，我们使用了扎带对套环进行线性引导，确保套环可以快速被吸入传送带中。

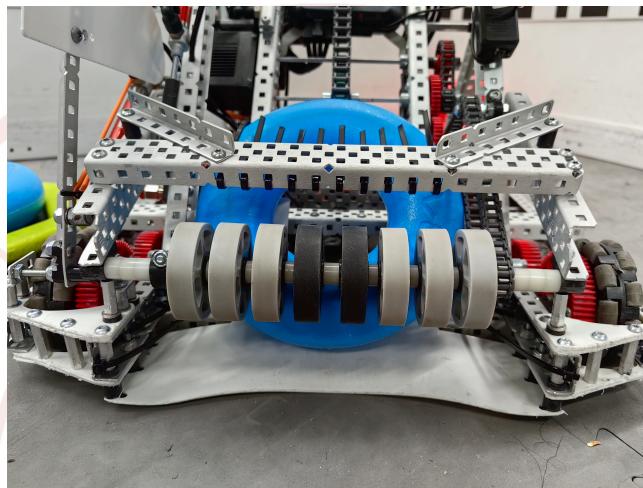


图 22: 3.5 版本的新吸子

3.6.3 新筛环

新筛环我们移动了光学传感器（颜色传感器）的位置，使其更好地识别吸入套环。此外，停止传送带的筛环思路也因不稳定改成了气缸筛环，通过气缸的伸出干扰环的路线使其中途掉落从而达到筛环的目的。

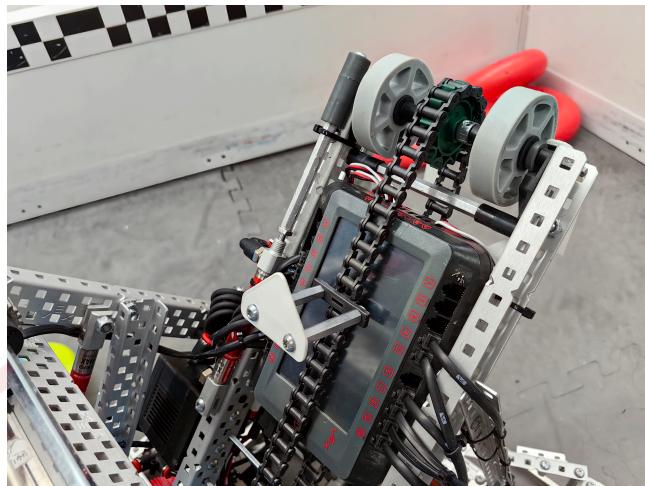


图 23: 3.5 车型筛环系统

3.6.4 低挂系统与 3D 打印队牌的应用

考虑到第一层的悬挂可以获得 3 分，而这三分也许可以起到决定性的作用，因此我们将以前放弃的低挂系统重新部署到了我们 3.5 版本的车上。由于 3.5 版本车具有轻量快速的特点，因此低挂系统也可以在比赛中得到更充分的利用，不管是拉开分数差距还是努力追赶分数都具有不错的实用性。

此外，我们在 3.5 版本的车上首次使用了全新的 3D 打印队牌。队牌由 2 个底座与 4 个红、蓝两色的号牌组成。号牌背面我们嵌入了强力磁铁，借此达到稳定并快速更换队牌的效果，省去了拧螺丝的麻烦。本套队牌由马睿皓设计并进行 3D 打印。

3.7 4.0 版

经过对上交比赛失利的仔细分析，我们总结了车辆方面如下的缺点：

- 底盘不稳定，易造成自动不稳定
- 底盘忽快忽慢又不善于在强烈撞击的环境下进行防守，既不适合进攻也不适合防守。
- 半自动仍存在缺陷，取边桩与套边桩仍然不稳定。
- 吸子的传动链条易断链，导致瘫痪

经过测试，我们在 2.0 版极为稳定的底盘之上进行修改并且增速，使车辆达到比 3.5 版本更快的速度及更高的打环效率。4.0 版本在吸子等各个方面均进行了优化。

3.7.1 改进后的吸子

为了防止断链的情况发生，我们将原先的细链改为粗链且通过更改底板增加了一套断链用特殊应对措施——在断链后将吸子抬起直接铲环仍能达到吸环效果。同时我们也将吸子前伸避免自动阶段过界。

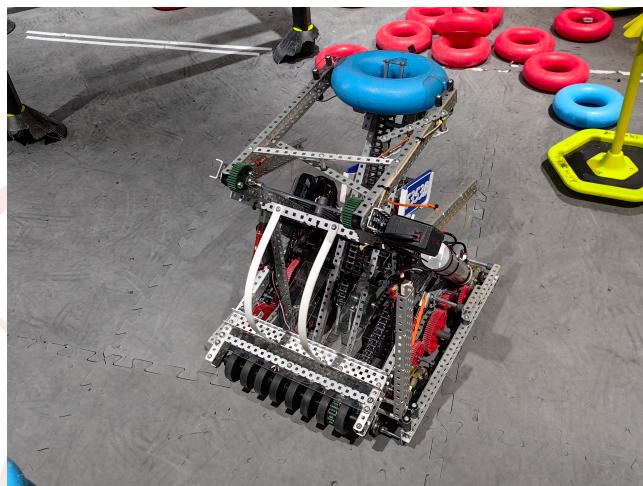


图 24: 4.0 的新吸子

3.7.2 新的摇臂系统

为了方便高挂、节省用气，我们用橡皮筋将摇臂与高挂联动。在赛局即将结束时，只要提前打开摇臂便可以迅速完成一级挂拿到 3 分，或许可以起到逆转赛局的效果。

3.8 4.5 版本

在我们对于 4.0 版本机器产生的一些问题进行改进后，我们成功研发出了 4.5 版本。

4.5 代车最大的亮点（和 3.5 以及 4.0 区别最大的）是更加轻量化的车身以及全新的操作模式。这一改变会使操作在一开始需要大量的适应，可是在适应后操作会提升一个新的档次。
新的轻量化的底盘带来的就是更快的加速度，这使我们的机器在自动阶段有了更多的优势，在道具的争夺中（比如中立的环区）占更多的优势。在吸子方面，我们将 4.0 原有的黑轮更换为白轮，使其在吸角中能有更大的优势。此外，位于吸子上方的 PET 板材被更换为更加坚硬的 POM 板材，避免触碰吸子造成吸角失败。

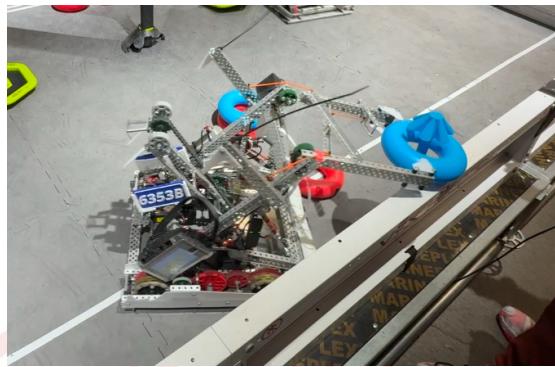


图 25: 4.5 版车

此外，我们还利用 onshape 和 PPT 为车辆设计了好看的装甲并打印了出来。



图 26: 4.5 版车辆装甲与贴画

3.9 整体研发日志——问题与改进

3.9.1 1.0 版本问题

1. 过重

- 底盘和结构使用了过多的材料，导致整体重量偏高，影响了速度和操控性。
- 重量的增加也导致电机负载过高，降低了机器人的运行效率和稳定性。

2. 效率低下

- 能量转换和运动方式不够优化，执行功能（例如清角和筛环）时存在较高的时间成本。
- 吸环、套边桩的操作速度较慢，难以满足比赛中的高效性要求。

3. 功能需求不足

- 清角功能：机器人在角落处理障碍物的能力有限，容易在赛场上被卡住。
- 筛环功能：对场地内环的识别和分类能力不足，缺乏相关传感器辅助。

3.9.2 2.0 版本改进

1. 减重设计

- 通过将轮子从 8 个减少到 4 个，降低了整体重量，提升了速度。
- 去掉了高挂功能，优化了整体重量分布，减轻了不必要的负担。

2. 提高效率

- 将抬臂设计改为摇臂结构，简化机械动作，提升动作响应速度和稳定性。
- 吸环和套边桩的速率得到提升，能够在比赛中更快完成任务。

3. 功能性增强

- 清角功能：在底盘结构上增加了斜角设计，使机器人能够更有效地清理赛场角落的障碍物。
- 筛环功能：在传送带上添加了光学传感器，提升了机器人对环的识别能力，使得筛选效率和准确性更高。

3.9.3 2.0 版本问题

1. 仍然过重

- 虽然相较 1.0 版本有所减重，但整体重量仍然偏高，速度和灵活性受到牵制。

2. 灵活性不足

- 结构设计复杂，底盘的机动性有限，导致机器人在赛场中不够灵活。
- 操控精准度不足，容易因为惯性较大而错过关键动作。

3. 功能需求不足

- 自动准确性方面存在缺陷，机器人在自动化运行时的精度无法满足预期要求。

3.9.4 3.0 版本改进

1. 进一步减重

- 减少气缸的数量，使用单气阀设计及省气管路布置，进一步降低整体重量。
- 优化整体结构，减少不必要的部件，提升速度和灵活性。

2. 提升灵活性

- 对底盘结构进行简化设计，提升机器人的机动性和操控性，使其能够更灵活地应对赛场环境的变化。
- 调整重心和轮子布局，减少转弯和快速移动中的惯性影响。

3. 功能性增强

- 准确性：新增全场定位功能，通过定位模块实现对机器人在赛场上的实时精确位置追踪，提高操作的精准度。
- 自动化性能：结合光学传感器和全场定位，进一步提升自动运行的准确性，减少人为操控的误差。

3.9.5 3.0 版本问题

1. 精益求精

- 随着比赛难度进一步上升，为了精益求精，之前为了简洁而放弃的一级挂又变得重要了
- 吸环的速度、角度兼容性以及夹子的夹持角度兼容性更需要被考虑。

2. 吸环稳定性

- 吸环的稳定性和效率仍需提高

3.9.6 3.5 版本改进

1. 一级挂的得分

- 车辆再次选择使用了新的低挂。

2. 吸环的改进

- 吸子位置增加了扎带，辅助环进入传送带。
- 吸子下方装备了底板：底盘的曲线与环相吻合，同时底板设置成了固定的角度，在与吸子的配合下，将环送入传送带。同时利用扎带引导环以正确的角度进入吸子从而节省时间。
- 吸子改为 2 黑轮 +6 白轮，增强稳定性。

3.9.7 3.5 版本问题

1. 精益求精

- 随着比赛难度进一步上升，为了精益求精，之前为了简洁而放弃的一级挂又变得重要了
- 吸环的速度、角度兼容性以及夹子的夹持角度兼容性更需要被考虑。

2. 吸环稳定性

- 吸环的稳定性和效率仍需提高

3.9.8 4.0 版本改进

1. 改进吸子

- 改进吸子，使其具有可以吸角落 1、3 环的特点。
- 改进传动设施避免断链

2. 高挂的改进

- 高挂与摇臂联动，方便进行快速的高挂。



4 打法策略

经过细致的赛题分析，我们判断这个赛季的场上打法需要很高的团队配合和极其明确的思路。因此，在赛前的训练赛中，我们也需要培养相对很成熟的打法思路，以应对场上千变万化的情况以及慌乱下的混乱思想。具体的实施情况会随场上的具体情况而变化，但是大致的思路如这一章节所示。

4.1 针对 0.1 和 0.2 规则的策略设计

基于在上文描述的我们认为对我们产生绝对性影响的规则，我们对于策略的设计主要在于对于加分区的争夺。在自动环节结束后，根据双方两台机器的位置会有细微调整，总体来讲还是先一台机器打一摞套环，同时场边队友需随时注意对方机器的位置，因为保护住加分区是至关重要的。在两台机器都分别打完一摞套环时，这时就存在两种情况，第一种是两个加分区都无人占领，这时两台机器就可以分别保住一个加分区什么都不需要做，根据对于分数的计算，这种打法除了极端情况来讲已经足以获胜。另一种情况，也是大多数情况下，只有一方的加分区是容易获得的，这时就需要一个联队清理加分区的初始 4 个套环后另一个联队将 Mobile Goal 放置于加分区中，然后由一支联队背着 Mobile Goal 进行对于加分区的保护，另一队这时就可以去争夺尽可能多的 Wall Stake 上的顶套环得分。

所以基于这个大体的思路应该在场上完成以下几点：

首先，**加强自动阶段的编程与协同**。自动阶段的表现可以极大地影响后续比赛的战略布局。根据初始位置和预设的程序，两台机器应该能够迅速定位并尽可能多地放置套环，确保在自动环节结束时能占据有利位置以便于转入手动操作阶段。这一部分的内容将在下一部分**程序思路**中更加详细地阐述。

其次，**灵活调整双机协作模式**。在两台机器完成各自的套环任务后，应根据实时比赛情况灵活调整策略。如果对方开始向加分区施压，一台机器可以转变角色，专注于干扰对手，阻止对方将 Mobile Goal 或其他得分对象移入加分区。此时，另一台机器则可以加快套环的放置速度，或者转移到另一加分区尝试占领。**总之，对于加分区的加速争抢和干扰对面对于加分区的争抢我们认为将会是这个赛季的核心竞争点。**

最后，**优化对 Wall Stakes 的争夺**。在第二台机器空出来后，除了争夺 Wall Stakes 顶部的套环外，还应评估是否有机会进行“清扫”，即快速移除对方已经放置的套环，减少对方的潜在得分，同时增加自己的得分机会。这要求操作者具备高超的操作技巧和对场地布局的精确判断。顶套环的争夺是极其至关重要的，一个顶套环称为一个“占领”，每个占领就可以有 3 分。因此，在一些比分相近的时候，占领杆是可以引领队伍队伍获得胜利的。

4.2 1.0 规则的策略改动

在 1.0 规则的改动后，改变了加分区和减分区的排布，这一点对于策略本身的影响不大，可是基于这个改动我们需要更多的训练，以把场地的改动更加熟练地记在我们的脑子里，以免在场上时因为习惯了先前规则中分区的排布从而进入到了错误的得分区影响比赛的走势。

这次规则的改动中对我们的打法思路影响最大的是加分区保护时间的出现。1.0 规则中规定了加分区在最后的 10 秒中有保护，因此 10 秒左右后就可以停止对于加分区的防守而去做一些别的事情。因此，比赛的思路就变化为了：

比赛开始时，队伍首先需要执行精确的自动程序，这通常涉及快速定位并占领关键得分点或执行特定任务以获得自动阶段的加分。进入手动操作阶段后，队伍应该分配两台机器人的角色和任务，一台专注于快速有效地操控 Mobile Goals 以占据得分区域，同时另一台机器人则集中在收集和堆叠套环，特别是在策略性地位重要的 Wall Stakes 上。此外，防守也是比赛中不可忽视的一环。队伍需要有能力快速识别并反应用于的攻

势，通过战术阻拦或在关键地区设置障碍来阻止对手得分。同时，团队成员必须保持高度的沟通和协调，以便迅速适应比赛的动态变化和对手的策略调整。在整场比赛中，策略的灵活性和队伍的执行力是赢得比赛的关键因素，每个动作和决策都应旨在优化得分机会，同时尽可能地限制对手的得分空间。其实这一次策略制定的核心还是一支队伍进攻加分区，一支队伍打套环并防守加分区，同时兼顾 Wall Stake 上面的占领，最后时刻根据机型决定是否进行高挂。

注释：在 1.0 及之前的规则中，用上文提到的 1.0 及 1.1 版机器，可以通过前方的吸子抬臂将对方的 Mobile Goal 撞倒以进行防守。

4.3 2.0 规则下的策略

2.0 规则的改动对于策略的影响非常大。首先是对于套环得分条件的重申，在 2.0 规则下，2.0 规则对 <SC3>、<SC5> 进行了重写，即 Ring 接触与该 Ring 同色的连队机器人或者地垫仍可以在 Stake 得分，而且即使套环触地依旧判定得分。这一点的规则改动很大程度上修改了操作的思路，因为在 2.0 规则发布之前，进攻方需要注意在比赛结束前一小会儿放下底座避免不算得分，同时防守方可以通过将对方的 Mobile Goal 撞倒而使对方的得分取消。在新规的加持下，相关的动作将失去意义。因此，实际上在新规内可以在场上“考虑得更少一些”，或者是可以把考虑这些的心思转移到争取高杆的占领、不同等级的高挂等。

其次是加减分区内套环的初始摆放位置的变化。这一点在手动阶段的影响并不大，可是对于自动阶段有着很大的干涉。由于前期的自动路线有设计把己方加分区内上方的同色套环吸入并且套入 Mobile Goal 中，可是这个变动使得最顶部的套环是对方颜色的套环，由此相关的自动路线将无法成功实施。进而需要设计新的自动程序路线思路。

最后一点是保护时间的修改。从 10 秒增加到 15 秒的多余的 5 秒时间可以增加很多操作手对于其他行为的可能性。这一个规则的改动增加的 15 秒时间说不定可以让后续的攀爬车有足够的时间进行攀爬。

秦皇岛华北赛时期的思路为：

15 秒：

底座区：手动开始直接占角，角落清空，机器上带一个底座

环区：第一摞打满，放加分区

(模拟环区 2 环，底座区单底座 2 环)

30 秒：

底座区：将 6 环底座放入角落，继续持有 4 环底座

环区：抢第三底座，打至少 2 个环

45 秒：

底座区：加分区防守，给第三底座留出位置，没打满的底座准备给队友

环区：打满第三底座，放加分区

1 分钟：

底座区：拿 6 环底座继续防守加分区，持续关注边桩情况，适时进行防守

环区：换底座，处理边桩

1 分 15 秒：

底座区：确保加分区 6 环加分成功，准备处理边桩

环区：争夺边桩顶环

1 分 30-1 分 45 秒：

底座区：争夺加分区边桩

环区：争夺减分区边桩

4.4 2.1 规则后备战上交比赛的策略

随着秦皇岛大区赛的结束，我们总结了比赛上出现的很多问题。一般来讲，问题主要还是出在了对于第三个 Mobile Goal 的争夺以及 Wall Stake 的战术地位。
在我们队内经过多次讨论后，我们一致决定：

- 优化自动路线更直接地摆好位置以更优势地抢夺第三桩
- 手动阶段刷分的同时也要刷边桩
- 优先第三桩的争夺!!!

因此，我们为上交的国赛制定了一份训练计划。

4.4.1 第三桩的争夺

无论如何，在自动阶段结束后，我们的第一优先级都是去争夺第三桩。
此时我们将己方的车分为加分区车和减分区车：加分区为抢夺中立 Mobile Goal 的一方，减分区为中立环区的一方。

对于减分区车：

第一优先级：防守加分区车放下的桩

进阶 1：满足第一优先级的情况下，清角

进阶 2：满足进阶 1 的情况下，打满自己的桩并放入加分区

进阶 3：满足进阶 2 的情况下，夹上加分区放下的桩准备换防

对于加分区车：

第一优先级：争夺最近第三桩，并威胁加分区

进阶 1：争夺到第 3 桩的情况下，在对方加分区打完易上的环，回己方加分区换车

进阶 2：争夺到第 3 桩的情况下，在对方加分区打完易上的环，回己方加分区清角

进阶 3：在满足进阶 2 后，打满自己车上的桩

第二优先级：

未争夺到最近第三桩，加强威胁对方加分区

进阶 1：取得加分区，对方无加分区，我们不是双加

进阶 2：取得加分区，我们偷鸡拿到第三桩，但上面 top ring 为对方环，盖一环放加分

进阶 3：取得加分区，我们偷鸡拿到第三桩，但上面 top ring 为对方环，盖环到我方 double score 对方，取得双加

第三优先级：

未争夺到最近第三桩，无法威胁对方加分区，无法造成双加，占领加分区边桩

进阶 1：在第三优先级基础之上，加分区边桩得分比对方高 3 分

进阶 2：在进阶 1 情况下，减分区边桩占领

进阶 3：在进阶 2 情况下，减分区边桩刷满

基于这一项，也就是第三桩的争夺，我们制定了相应的 2 个专项练习用来提高我们在场上的表现。

专项练习 1: 一支队伍作为对手进行练习，模拟自动结束第三桩在对手半场，对手持有本方区底座，第三桩在中线附近 1 格地垫左右的位置，进行第三桩的抢夺并威胁对手加分区，在对手加分区打满环。

专项练习 2: 一支队伍作为对手进行练习，模拟自动结束第三桩被对手持有，对手本方区底座在加分区附近，进行第三桩的抢夺并威胁对手加分区，在对手加分区打满环。

4.4.2 加分区攻防

此时我们将两辆机器分为进攻车与防守车，并分别提供相应打法指令。

首先讨论**防守方车辆**: 基本指令为手动开始减分区机器拿一个底座守一个底座清角，队友留下来的底座，将多环底座放置在加分区内，斜 45 度防守，可以适当在角落内囤积对手的环，让总环数（总分数）处于优势。

第一优先级: 守住自己和队友留下来的底座

接下来是对于**进攻方车辆**的指令：第三桩抢到的情况下不需要刻意抢夺对方加分区，在吸环过程中顺带干扰对手防守机器即可；第三桩丢失的情况下可以采用挂中立边桩诱导对手离开加分区，配合转身 + 挤的方式尝试占领对手加分区或者偷对手底座拖到减分区。

这相关的专项练习有：

专项练习 1: 一支队伍作为对手进行练习，教练为防守方，加分区内有 1 个底座，教练机器上 1 个底座，队员不带底座进行进攻，目标抢到底座并占领加分角。

专项练习 2: 两支队伍攻守易型，但底座在加分区外

4.4.3 边桩攻防

边桩进攻以占领为主，尽可能多的达成占领，刷满边桩一般只在丢失第三桩时打。防守车针对加分区边桩进行防守，至少做到不让对手占领，有余力可以打占领。

针对边桩的攻防我们有两个专项练习：

专项练习 1: 教练作为对手进行练习，队员为防守方，加分区内有 1 个底座，队员机器上 1 个底座，教练对加分区和边桩进行进攻。

专项练习 2: 教练作为对手进行练习，队员为进攻方，对加分区边桩和加分区进行进攻，以占领加分区边桩为主，教练机器上 1 个底，加分区内 1 个底座。

4.4.4 高风险打法

高风险的方式有三种打法：

- 加分区攻防中第三桩丢失的进攻打法；
- 第三桩丢失减分区边桩刷 6 环；
- 抢对手底座塞减分区。

4.5 技能赛思路

如下图所示为技能赛的手动思路图，从联队桩开始路线由红线表示，确保每个占领都必须打上，最后再进行 1 级挂，可以得到较高的分数。

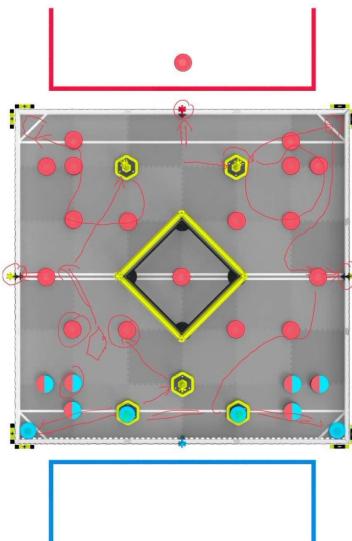


图 27: 技能赛思路图

5 操控算法的设计思路

5.1 基本控制

对与 VEX 车辆来说，基本控制相对好写。我们结合了车上各部分硬件，再结合对操作手柄的判定，先写出了一套基本移动以及手动操控程序方便调试。

```
1 void Move(float lPower, float rPower){  
2     LeftChassis1.spin(fwd, 0.128 * lPower, volt);  
3     LeftChassis2.spin(fwd, 0.128 * lPower, volt);  
4     LeftChassis3.spin(fwd, 0.128 * lPower, volt);  
5  
6     RightChassis1.spin(fwd, 0.128 * rPower, volt);  
7     RightChassis2.spin(fwd, 0.128 * rPower, volt);  
8     RightChassis3.spin(fwd, 0.128 * rPower, volt);  
9 }  
10  
11 void Drive(double l, double r) {  
12     LeftChassis3.spin(fwd, l, voltageUnits::mV);  
13     LeftChassis2.spin(fwd, l, voltageUnits::mV);  
14     LeftChassis1.spin(fwd, l, voltageUnits::mV);  
15  
16     RightChassis3.spin(fwd, r, voltageUnits::mV);  
17     RightChassis2.spin(fwd, r, voltageUnits::mV);  
18     RightChassis1.spin(fwd, r, voltageUnits::mV);  
19 }  
20  
21 void Stop1(brakeType type){  
22     LeftChassis1.stop(type);  
23     LeftChassis2.stop(type);  
24     LeftChassis3.stop(type);  
25  
26     RightChassis1.stop(type);
```

```
27     RightChassis2.stop(type);
28     RightChassis3.stop(type);
29 }
30
31 void ArmControl(int power){
32     if(power == 0){
33         Arm.stop(hold);
34     }
35     else{
36         Arm.spin(fwd, 0.128 * power,volt);
37     }
38 }
39
40 void IntakeControl(int power){
41     if(power == 0){
42         Intake.stop(brake);
43     }
44     else{
45         Intake.spin(fwd, 0.128 * power,volt);
46     }
47 }
```

5.2 PID 底盘

5.2.1 v0.1

v0.1 是试行版 PID。在以简单形式编写了 Move 后，我们先在大致车辆框架的基础上压上一定配重来进行 kp、ki、kd 的调试，毕竟这三个值在每一辆车上可能都会有些许的不同，压配重也是为了模拟后面全功能完备的时候使最终车辆的性能仍然保持得较好。

5.2.2 v0.5

v0.5 是我们用于 1.0 版车辆的 PID。该套算法功能比较均衡，速度和准确性都可以很不错的保持。虽然这版 PID 参加了正赛并取得不错的战绩，但仍有较大的优化空间。

5.2.3 v1.0

v1.0 是我们用于 2.0 版车辆的 PID，各项方面都得到了相应优化与提升，包括速度、准确度在内都相较于 v0.5 有了重大改进，从这一版 PID 开始，我们逐渐转向了“速度优先”的优化思路。直至 v2.0 版本，里面也有很大一部分（甚至是说完全包含）v1.0 版本的源代码。本套 PID 的优势在于速度，机器人可以更灵活地在场地上穿梭打环，无疑是对手动体验的巨大提升。缺点是有着较大的误差，自动也是“勉强可用”的那种。

以下是 v1.0 的转向与前进部分。

```
1 void GyroTurn(float target,float kp,float kd) //PD转向
2 {
3     int timeused = 0;
4     float dtol = 0.1;
5     float errortolerance = 2;
6     float lim =127;
7     float error = target - Gyro.rotation();float lasterror;float v = 0;
```

```
8  bool arrived;
9  float timetol = fabs(error) < 300 ? 1000 : fabs(error) * 1.2;//2.7
10 float pow;
11 int endp = 0;
12 lasterror = error;
13 arrived = error == 0;
14
15 DrawFrame();
16
17 if(target>=0)
18 {
19     endp=-5;
20 }
21 else
22 {
23     endp=5;
24 }
25 while (!arrived)
26 {
27     timeused += 10;
28     error = target - Gyro.rotation();
29     v = error - lasterror;
30     lasterror = error;
31     //IF ((fabs(error) < ERRORTOLERANCE && fabs(v) <= DTOL) || TIMEUSED > TIMETOL)
32     {ARRIVED = TRUE;}
33     if ((fabs(error) < errortolerance && fabs(v) <= dtol) || timeused > timetol)
34     {
35         arrived = true;
36         Brain.Screen.printAt(200,200,"time:%dms",timeused);
37         turn(0,0);
38         pow = kp * error + kd * v ;
39         pow = fabs(pow) > lim ? sgn(pow) * lim : pow;
40         turn(pow,0);
41         Brain.Screen.drawPixel(20+timeused/4, 120 - LeftChassis1.velocity(rpm)/6);
42         sleep(10);
43     }
44     turn(0, 0);
45 }
46
47 void GyroMove( double power, double enc , float g) //陀螺仪移动
48 {
49     //BOOL FINISH;
50     int time=0;
51     float menc;
52     float turnpower;
53     EncoderReset();
54     //INT TIMEOUT_ =ENC<300?500:ENC*1.5;
55     while(true)
56     {
57         time += 1;
58         // MENC = (LEFTCHASSIS1.ROTATION(ROTATIONUNITS::DEG) + RIGHTCHASSIS1.ROTATION(
59         // ROTATIONUNITS::DEG)+LEFTCHASSIS2.R
60         otation(rotationUnits::deg) + RightChassis2.rotation(rotationUnits::deg) +
61         LeftChassis3.r
62         otation(rotationUnits::deg) + RightChassis3.rotation(rotationUnits::deg))/6;
63         menc = (LeftChassis1.position(rotationUnits::deg) + RightChassis1.position(
64             rotationUnits::deg) + LeftChassis2.position(rotationUnits::deg) +
65             RightChassis2.position(rotationUnits::deg) + LeftChassis3.position(
66             rotationUnits::deg) + RightChassis3.position(rotationUnits::deg)) / 6;
67         turnpower = 1.16*(g - Gyro.rotation());
68         //IF (TIME>TIMEOUT){FINISH = TRUE;}
69         if (fabs(enc)-fabs(menc)<2 )
70         {
71             break;
72 }
```

```

69     if (enc > 0)
70     {
71         forward_(power+turnpower,power-turnpower);
72     }
73     else
74     {
75         forward_(-power+turnpower,-power-turnpower);
76     }
77     sleep(1);
78 }
79 forward_(0,0);
80 }
```

5.2.4 v2.0

在 v1.0 的基础上，我们引入了另一套—准确为主的 PID 算法，达到“速度与准确性兼存”，同时引入 Eigin 线性代数库为技能赛的自动（如全场定位）以及刁钻的场上自动（如曲线行驶）打好基础（虽然相关程序并未完善），做成适用于 3.0/3.5 版本车辆的 PID v2.0。得益于 3.0/3.5 版本车辆上安装有定位轮，2.0 版 PID 对其添加了一定的支持，但也正是因为如此 2.0 版 PID 仍然有极其广阔的优化空间，涵盖更精细的全场定位与曲线行驶等。因此，本套 PID 的优点是同时注重了速度与准确性，缺点是仍然有一定的改进空间。（毕竟是要先以课内学习为重，时间相对少一些）

以下是 v2.0 自动的转向与前进程序，通过与 v1.0 的相关内容组合使用形成自动程序。

```

1 void turnToHeading(double heading, double speedRatio = 1, double speedCap = 1,
                    double reversed = 0, double kP = 410, double kI = 25, double kD = 3300) {
2     resetPID();
3     double time = 0;
4     double turnError = heading - Gyro.rotation(rotationUnits::deg) - reversed *
180;
5     while (fabs(turnError) > 0.3 || fabs(Gyro.gyroRate(zaxis, velocityUnits::dps)
) / 100 > 0.3) {
6         turnError = heading - Gyro.rotation(rotationUnits::deg) - reversed * 180;
7         double driveOutput = angularPID(turnError, kP, kI, kD);
8         if (fabs(driveOutput) > 12000 && driveOutput > 0) driveOutput = 12000;
9         else if (fabs(driveOutput) > 12000 && driveOutput < 0) driveOutput =
-12000;
10        driveOutput *= speedRatio;
11        clamp(driveOutput, -12000 * speedCap, 12000 * speedCap);
12        // cout << "ERROR: " << TURNERROR << " DRIVE OUTPUT: " << DRIVEOUTPUT <<
DERIVATIVE: " << IMU.GYRORATE(ZAXIS, VELOCITYUNITS::DPS) / 100 << endl
13        ;
14        setDrive(driveOutput, -driveOutput);
15        time += 10;
16        vexDelay(10);
17        // cout << "SETTLED: " << TIME << endl;
18        setDrive(0, 0);
19    }
20
21 void driveStraight(double distance, double speedRatio = 1, double speedCap = 1,
                     double kP = 2000, double kI = 230, double kD = 57000) {
22     resetPID();
23     double error = distance;
24     double leftSum = ((LeftChassis3.position(rotationUnits::rev) * (
driveWheelDiameter * M_PI) * (dt_rpm / 600)) +
25     (LeftChassis2.position(rotationUnits::rev) * (driveWheelDiameter * M_PI) * (
dt_rpm / 600)) +
```

```

26     (LeftChassis1.position(rotationUnits::rev) * (driveWheelDiameter * M_PI) * (
27         dt_rpm / 600));
28     double leftCurrentReading = leftSum / 3;
29     double leftPrevReading = leftCurrentReading;
30     double distaceTraveled = 0;
31     while (fabs(error) > 0.2 || LeftChassis2.velocity(rpm) * driveWheelDiameter /
32         6000 > 0.02) {
33         // GET DRIVETRAIN DISTANCE READING
34         leftSum = ((LeftChassis3.position(rotationUnits::rev) * (
35             driveWheelDiameter * M_PI) * (dt_rpm / 600)) +
36             (LeftChassis2.position(rotationUnits::rev) * (driveWheelDiameter * M_PI) * (
37                 dt_rpm / 600)) +
38             (LeftChassis1.position(rotationUnits::rev) * (driveWheelDiameter * M_PI) * (
39                 dt_rpm / 600)));
40         leftCurrentReading = leftSum / 3;
41         double deltaLeft = leftCurrentReading - leftPrevReading;
42         distaceTraveled += deltaLeft;
43         // ERROR CALCULATION
44         error = distance - distaceTraveled;
45         double driveOutput = lateralPID(error, kP, kI, kD);
46         if (fabs(driveOutput) > 12000 && driveOutput > 0) driveOutput = 12000;
47         else if (fabs(driveOutput) > 12000 && driveOutput < 0) driveOutput =
48             -12000;
49         // cout << "ERROR: " << error << " DRIVE OUTPUT: " << driveOutput << endl;
50         setDrive(speedCap * driveOutput, speedCap * driveOutput);
51         leftPrevReading = leftCurrentReading;
52         vexDelay(10);
53     }
54     setDrive(0, 0);
55 }
```

5.3 半自动设计 (3.x 版本)

为了方便手动套边桩、取边桩的操作，我们撰写了一套对于摇臂的半自动程序以便于操作。在同时按下 L1 和 L2 按键后，摇臂会抬升至水平位置方便接收从传送带升上来的套环；在抬起摇臂后若松开相应按键，摇臂会回到初始位置；此外我们还针对可能出现的取环进行了设计，写入了一个取边桩第 6 环的位置，方便取环、套上自己的并完成逆转。

```

1 int ArmSpin()
2 {
3     if(ArmAngle_Bool == 0){
4         if(L1) ArmControl(100); //手臂抬起
5         else if(L2) ArmControl(-100); //手臂放下
6         else if(Angle_Arm.angle(deg) > 10 and Angle_Arm.angle(deg) < 350) ArmControl
6             (-100);
7         else Arm.stop(hold);
8     }
9     else if(ArmAngle_Bool == 1){
10        if(L1) ArmControl(100); //手臂抬起
11        else if(L2) ArmControl(-100); //手臂放下
12        else if( !L1 and ! L2 and Angle_Arm.angle(deg) >48 and Angle_Arm.angle(deg) <
12            95) ArmControl(-100);
13        else if( !L1 and ! L2 and (Angle_Arm.angle(deg) < 15 or Angle_Arm.angle(deg) >
13            350)) ArmControl(20);
14        else if( !L1 and ! L2 and Angle_Arm.angle(deg) > 22 and Angle_Arm.angle(deg) <
14            48) ArmControl(-10);
15        else if( !L1 and ! L2 and Angle_Arm.angle(deg) > 190 and Angle_Arm.angle(deg)
15            < 350) ArmControl(-100);
16        else if( !L1 and ! L2 and Angle_Arm.angle(deg) > 158 and Angle_Arm.angle(deg)
16            < 190) ArmControl(-20);
17    }
18 }
```

```
17     else Arm.stop(brake);  
18 }  
19 return 0;  
20 }
```

5.4 半自动设计 (4.x 版本)

4.x 版本我们对手动进行了大改。在以往的手动中，R1 和 R2 两个按键主要负责吸环和退环，而在 4.x 版本上我们将 R2 改成了摇臂准备模式。在持续按下 R2 时，摇臂会尝试移动到准备位置，当其到达准备位置时传送带开始运动将环送上摇臂。此外我们在右侧摇杆上、下两个方向加入了新功能——摇臂的控制。由其上下拨动控制摇臂的高度，既方便了套边桩，也方便取边桩。此外，我们还加入了档位的设计并融入摇杆操作中，使得摇臂可以快速达到取 6 环/套边桩位置，操作更为便捷。



6 自动程序思路

相比于上个赛季，这个赛季的自动程序需要更多套来与队友不同的自动程序完成配合。在下文我将分别描述对于不同机型的自动程序设计思路。下文中的图片中的示例比例可能与现实不符，重点希望突出的是我们每一步设计自动程序的思路。下文中本队伍的自动为两边是完全镜像相同的思路，故我们只会显示并说明「红方」的自动，蓝方自动即为此自动的完全镜像。(3.0 及以后版本版本因重新绘图，显示并说明「蓝方」的自动。)

* 注 1: 由于 0.1 版本的机器仅仅作为实验机型，主要负责相关功能的测试，因此并没有为 0.1 代机型制作任何自动程序，因此不作为本部分内容。

* 注 2: 由于有世锦赛名额的大赛中「AWP」的判定发生改变，且为了考虑队友自动调节，我们并未写出单车的“Solo AWP”，以下的「AWP」也就变成了尽可能满足「AWP」条件的半「AWP」。

6.1 1.0&1.1 版机器

6.1.1 红方右场/蓝方左场两环

第一套自动是作为红方在右场的自动路线思路，这一套自动为基础的两环一底座思路，配合队友很容易拿下 AWP。首先先从出发区出发，用夹子拾取己方的底座，随后将自己的预装套环打到 Mobile Goal 上。随后转弯向前进走失去一摞套环中下面的那个己方颜色的套环并把这个套环也打到 Mobile Goal 上。最后，将底座放下，并且撞高挂杆完成 AWP 的部分任务。以此，完成一个底座两个套环的最基础自动程序。

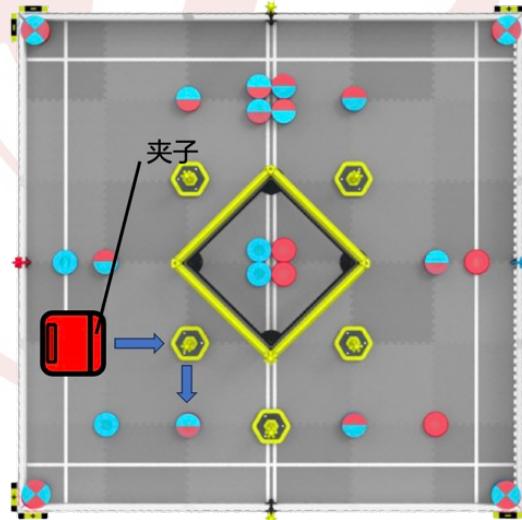


图 28: 1.0 车红方右侧两环自动

6.1.2 红方右场/蓝方左场双底座

第二套自动也是红方右场，夹取两个底座。整体的流程是，从发车区出发，向前移动并夹住己方的底座，打自己的预装套环。打完第一个环后，放下底座转向朝向中立底座。向后退并且再次转向用夹子再次朝向中立底座，随后夹住中立底座转向摞起来的套环，向前走吸入下方的那个套环打入第二个底座。最后，将底座放下，并且撞高挂杆完成 AWP 的部分任务。这一套自动的结果是两个底座两个套环两个占领，一共 6 分，相比描述的上一套自动多了 2 分。

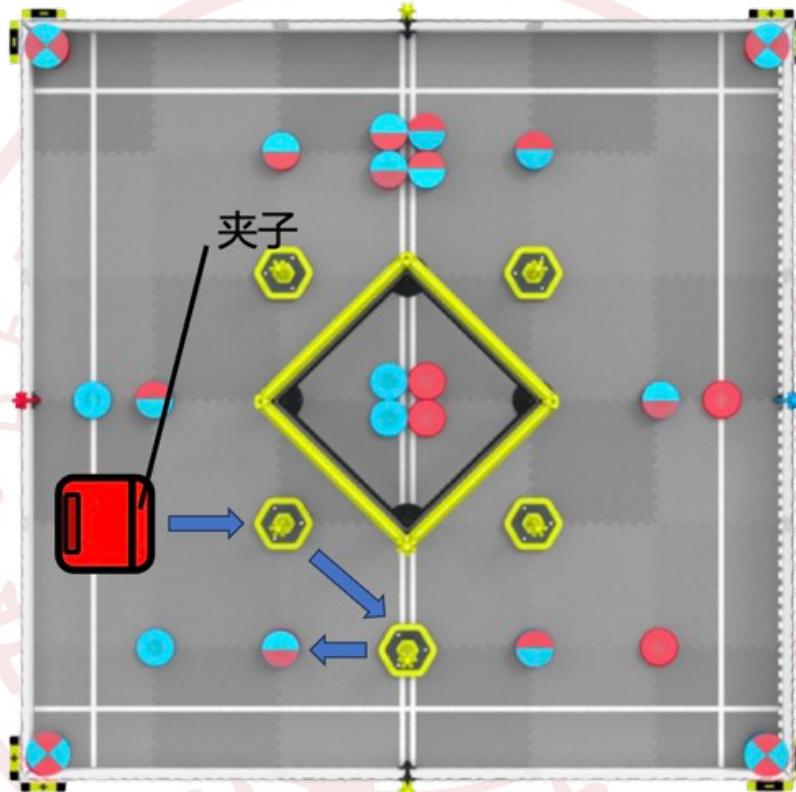


图 29: 1.0 车红方右侧双底座

但是，这一套自动也有很多存在的问题，比如路线的设计对中立底座的抢夺速度比较慢，而且一开始我们认为（假定）底座的摆放方向没有要求。所以，最后这一套自动程序并没有在赛场上真正的实施。

在北京延庆赛赛后，我们通过研究别的队伍的自动路线并且自己仔细回想了下自己的程序和路线，我们认定：若想抢到中间中立的 Mobile Goal，需要设计一个路线可以直线去抢夺中立的 Mobile Goal。鉴于这一套自动的思路是在赛季初期备战北京延庆赛的期间完成的，于是在延庆赛中放弃了这套自动，并不急于将这一套程序立刻调试出来。

6.1.3 红方左场/蓝方右场三环

这一套自动的程序是从红方的左场出发的，首先开向己方的底座夹住打一个预装的套环，随后转向擦着两个的堆套环，用吸子正常向前行驶吸入下方的己方同色套环打进底座，接着转向一个角度非正面向靠里的堆套环的下己方套环行驶，吸入底套环打入 Mobile Goal，完成一个底座上三个套环。最后，将底座放下，并且撞高挂杆完成 AWP 的部分任务。

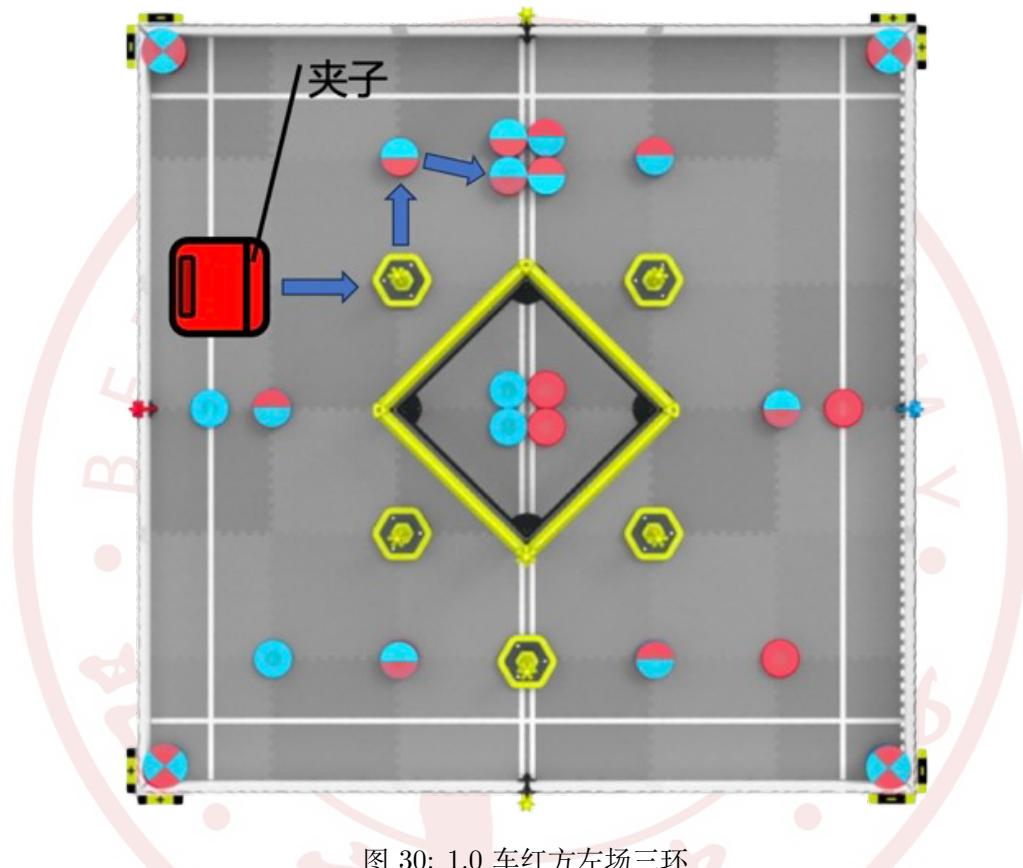


图 30: 1.0 车红方左场三环

然而，这一套自动程序也有一个潜在的问题或者是隐患。这个隐患就是如果对方的机器也要吸这四摞套环之一或之二，这四摞套环的摆放和排列必然为改变，因此如果己方完成到这一个步骤的速度比对方慢，将会在这一部分受到很大的影响，轻则吸子被卡住或者无法吸收到本方的套环，重则有可能把对方颜色的套环打成顶套环或者被拉到对方的场地里自动阶段 DQ。

6.1.4 红方右场/蓝方左场单车 AWP

由于在常规赛的排名中 AWP 的取得至关重要，在赛季初期我们便开始尝试研发单车 AWP 的程序。

这个单车 AWP 的思路是针对 1.0 和 1.1 车型的，这次的发车位置是需要摆车架共同协作完成摆车的。思路是首先将抬臂升起，吸入角落的一摞套环的最高的己方的套环，随后去夹中间的 Mobile Goal 打入预装套环和吸入的套环，接着放下底座后争夺中立的底座，回来吸两层套环的下面一个己方颜色的套环，随后放下底座升起抬臂用抬臂上的扎带触碰高挂杆，实现单车 AWP，两个底座三个套环。

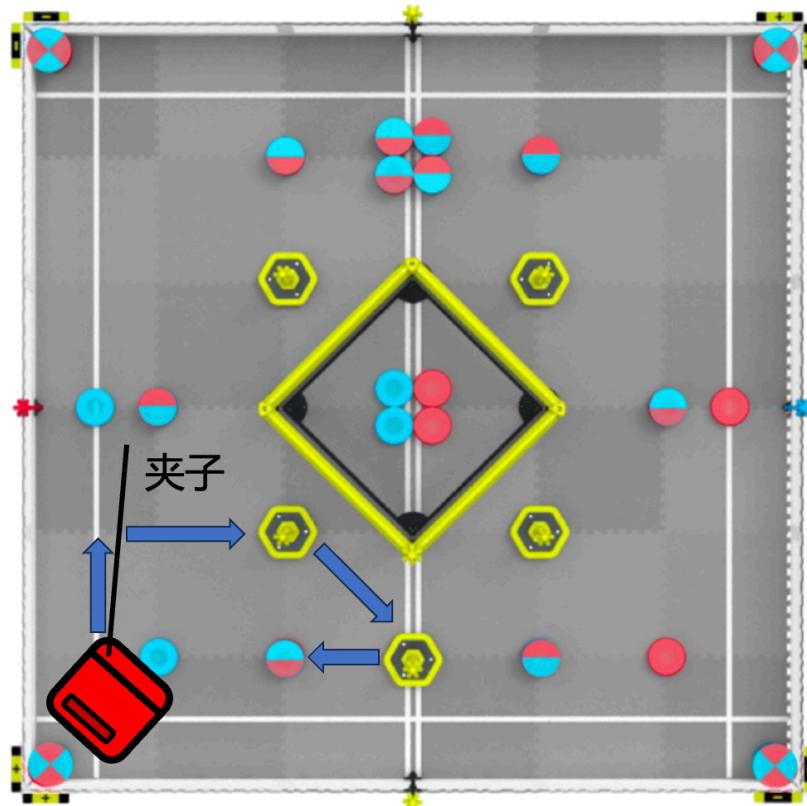


图 31: 1.0 车红方右场单车 AWP

然而这一个自动也有潜在的问题。由于一开始我们认为底座的摆放不需要考虑方向，所以在延庆赛发现底座的摆放也有方向要求后自动会出现一些问题。同时，在 2.0 的规则下，四摞套环的顶套环将全部换成对方颜色的套环，这使得这一套自动程序在使用 2.0 规则的比赛中将无法施行。

6.2 2.0 版机器

进行了机型的调整，但大部分的程序思路并没有进行本质的改变，增加的部分在吸子上方加了一个气阀可以吸摆着的环，而且吸环的效率和速度也都得到了提高。增加的自动思路主要是以下两条路线。(在 2.0 版的车的自动程序中，发车位置都由正发车改为了侧发车，即如下图所示)

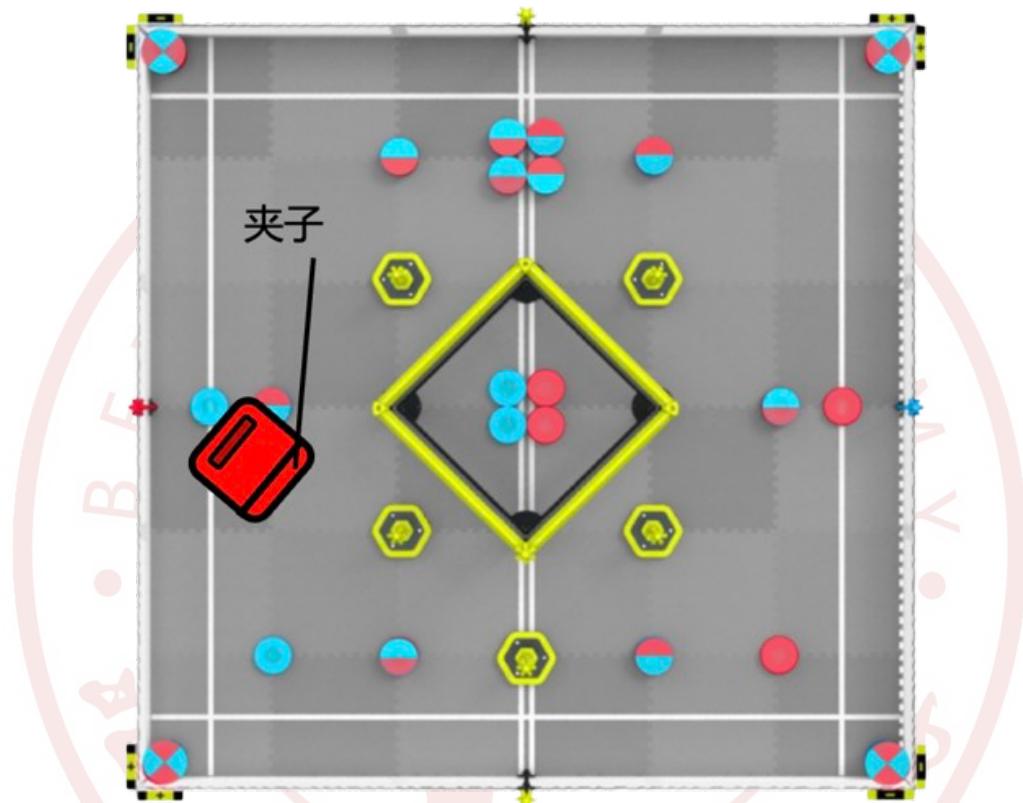


图 32: 2.0 车的发车位置示意图

图上的机器角度和位置描述并不准确，在现实中的机器大小可以正好在初始位置用夹子对准底座，这样节省了更多的时间的同时也能更好的夹取底座，只是摆车的难度加大了一些。

接下来的部分将会主要介绍两条新设计的自动程序路线。

6.2.1 红方左场/蓝方右场四环

这一套自动就采用了侧方向的发车，首先向前行驶夹住底座，掉头并将预装的套环打进底座中。然后一步需要很多微调，最终可以以横着行驶的方式将四摞套环中的两摞的下方套环吸入并打到底座中。这一步的难点是要确保车轮很小心不能过界。在打了3个套环后，下一步就是转向到上下摞的两个套环的方向，向套环行驶用吸子将下方的套环吸入，完成4个套环的打入。最后一步就是将底座放下（在2.0规则下选择不放底座），升起抬臂向高挂杆移动触碰高挂杆完成AWP的部分任务。

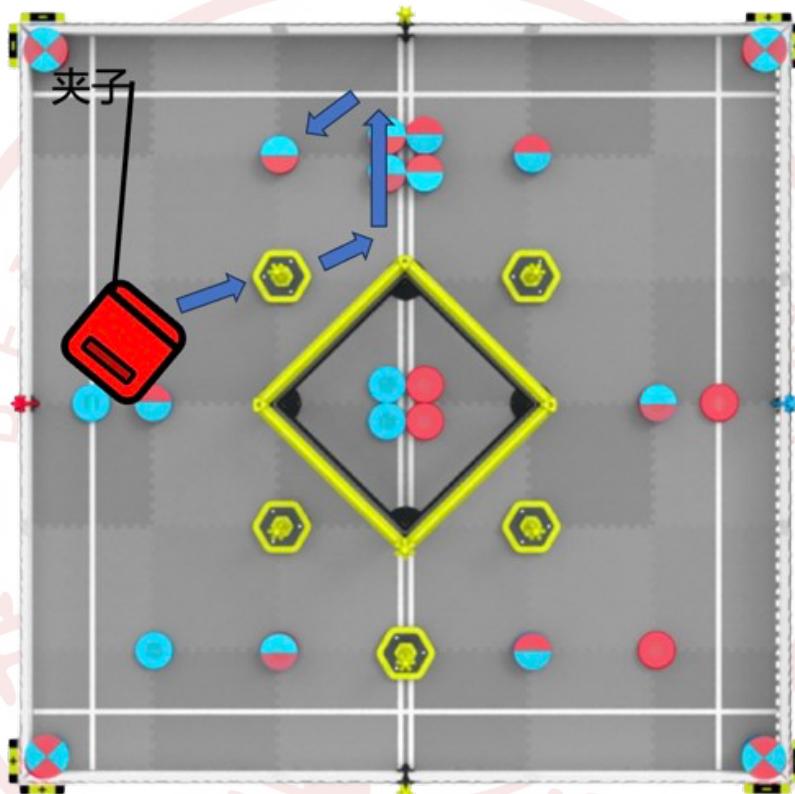


图 33: 红方左场四环

这一套自动程序的最大难点就是前文提到的容易过界的事情。实际上，相比于2.0版的车，1.0和1.1版车的吸子其实更容易完成这个四环的路线，因为1.0和1.1的机型的吸子是突出的，更加宽，而2.0的是有点凹进车身内的。因此，这一套自动需要反复的进行微小调整与修改才可以在赛场上用到。

```

1 extern int auton;
2 void runAuto()
3 {
4     GyroMove(90, -400*0.85, 0);
5     GyroMove(30, -200*0.85, 0);
6     Stop();
7     Lift.set(1);

```

```
8   Catch.set(1);
9   sleep(200); //夹底座
10
11  GyroTurn(150, 5, 30);
12  Stop();sleep(200);
13  吸子Control(100);
14  吸子Angle.set(1);
15  GyroMove(40, 200*0.85, 150);
16  吸子Control(-100);
17  GyroMove(60, 220*0.85, 150);
18  Stop();
19  吸子Angle.set(0);
20  sleep(300);
21  吸子Control(100);
22  Stop();
23  sleep(200);
24  GyroMove(40, -100*0.85, 110);
25  GyroMove(20, 200*0.85, 113);
26  Move(50, 50);
27  sleep(800);
28  Stop();
29  sleep(300);
30  GyroMove(50, -350*0.85, 113);
31  Stop();sleep(200); //吸环
32
33  GyroTurn(65, 5, 30);
34  Stop();sleep(200);
35  GyroMove(90, 150*0.85, 65);
36  GyroMove(80, 200*0.85, 65);
37  GyroTurn(-75, 5, 30);
38
39  GyroMove(50, 450, -75);
40  ArmControl(90);
41  Stop();sleep(200);waitUntil(Angle_Arm.angle() > 95 and Angle_Arm.angle() < 350)
;
42  ArmControl(0);
43  GyroMove(50, 300, -75); //原700
44  Stop();
45 }
```

6.2.2 红方右场/蓝方左场双底座单车 AWP

这一套的自动从侧边发车，以最快的速度争夺中间的中立底座，在直线距离走完后转向对准底座夹住。随后将预装的套环打入，并且对准摞两层的套环，行驶吸入下方的套环，在第一个底座上打入两个套环。然后放下底座，向第二个底座行驶夹住。重要的一步来了，就是用气阀升高吸子的夹角，驶向中间两个摞起来的套环的上方的那个，中间换挡为慢速，吸入后夹着底座撞高挂杆完成 AWP 的任务。

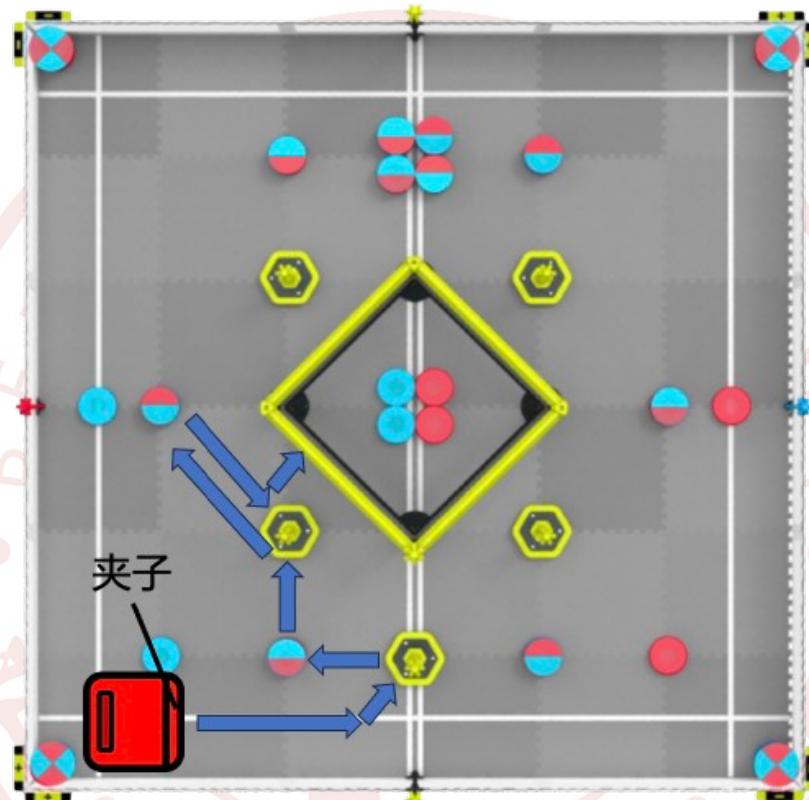


图 34: 红方右场双底座三环单车 AWP

在测试的过程中，这一套自动程序的容错率极高，而且得益于 2.0 车的强大功能，一切步骤的进展速度也十分迅速。因此这一套程序将会是我们队在场上的常用程序，变为了近一两场的比赛中的“神自动”。

```

1 extern int auton;
2 void runAuto()
3 {
4     GyroMove(-90, 500*0.85, 0); //往前
5     GyroMove(-40, 140*0.85, 0); //往前
6     Stop();
7     sleep(200);
8     GyroTurn(-35, 5, 30); //转弯面向第一个MG
9     Stop();sleep(200); //STOP
10

```

```
11  GyroMove(-70, 80*0.85, -35); //准备去
12  GyroMove(-30, 180*0.85, -35); //放慢速度ORI-170
13  Stop();
14  sleep(300);
15  Catch.set(1); //夹子
16  sleep(100);
17  Lift.set(1); //托盘
18  sleep(300);
19  GyroMove(70, 30*0.85, -35); //离开自动分界线
20  Stop();
21  sleep(200);
22  GyroTurn(27, 5, 30);
23  Stop();
24  sleep(200);
25
26  吸子Control(100);
27  GyroMove(80, 200*0.85, 27); //吸第一个
28  Stop();
29  sleep(1400); //往前端红
30  Lift.set(0);
31  Catch.set(0); //放
32
33
34  GyroTurn(-85, 5, 30); //背对第二个MG
35  Stop();
36  sleep(200);
37  GyroMove(90, -150*0.85, -85); //变速拿
38  吸子Control(0);
39  GyroMove(35, -250*0.85, -85);
40  Stop();
41  Lift.set(1);
42  Catch.set(1); //拿第二个MG
43  sleep(200);
44  吸子Control(100);
45  sleep(500);
46
47  GyroTurn(0, 5, 30);
48  GyroMove(90, 500*0.85, 0);
49  Stop();
50  sleep(200);
51  GyroTurn(-80, 5, 30);
52  Stop();
53  sleep(200);
54  吸子Control(-100);
55  GyroMove(90, 550*0.85, -85);
56  Stop();
57  sleep(200);
58  GyroMove(40, -100*0.85, -85);
59  Stop();
60  sleep(200);
61  GyroMove(90, 100*0.85, -85);
62  GyroTurn(-100, 5, 30);
63  Stop();
64  sleep(200);
65  GyroTurn(-80, 5, 30);
66  Stop();
67  sleep(200);
68  GyroMove(40, -100*0.85, -85);
69  Stop();
70  sleep(200);
71  GyroMove(90, 100*0.85, -85);
72  GyroTurn(-100, 5, 30);
73  Stop();
74  sleep(200);
75  GyroTurn(-80, 5, 30);
76 }
```

6.3 3.0&3.5 版机器

在 3.0、3.5 版中，车辆的速度增加了许多。得益于其速度优势，我们便可以做出更为强大的自动程序，为手动阶段打好基础。

6.3.1 新的筛环设计

考虑到写出的自动会较为充分利用自动时间，我们为此设计了一套新的筛环程序，其通过多线程的方法以定时器中自动开始至现在的时间控制筛环的强制关闭，最大限度地减少自动结束时由于筛环没有关闭而对手动的影响。此外也重新安排了筛环多线程开启、关闭的时机，达到省气的目的。

以下是三种不同的筛环，分别为定时结束筛蓝、定时结束筛红、遇到套环停带（将环暂时存储到车中）。

```
1 int SelectColor_RedTeam() //红队筛蓝环
2 {
3     while(1)
4     {
5         if(吸子Color.hue() > 180 and 吸子Color.hue() < 240)
6         {
7             Time1.reset();
8             Select_Bool = 1;
9         }
10
11        if(Time1.time(msec) < 800 and Select_Bool == 1)
12        {
13            SelectRing.set(1);
14        }
15        if(Time1.time(msec) > 800)
16        {
17            SelectRing.set(0);
18        }
19    }
20 }
21
22 int SelectColor_BlueTeam() //蓝队筛红环
23 {
24     while(1)
25     {
26         if(吸子Color.hue() < 20 or 吸子Color.hue() > 350)
27         {
28             Time1.reset();
29             Select_Bool = 1;
30         }
31
32         if(Time1.time(msec) < 800 and Select_Bool == 1)
33         {
34             SelectRing.set(1);
35         }
36         else
37         {
38             SelectRing.set(0);
39         }
40     }
41 }
42
43 int AWP2Ring() //遇到套环停带
44 {
45     while(1)
46     {
```

```
47     if(吸子Color.hue() < 10 or 吸子Color.hue() > 350 or (吸子Color.hue() > 160 and  
48         吸子Color.hue() < 240))  
49     {  
50         吸子Control(0);  
51     }  
52 }
```

6.3.2 蓝方右场/红方左场六环

这是一套采取正发车（即夹子正对 Mobile Goal）的全新 6 环自动方案，将在本次比赛首次投入使用。由于机械结构的改进致使原来 2.0 版车辆的四环路线易导致过线，我们便使用了一种新方案，重写了环区的代码。此路线为先夹 Mobile Goal，再“迂回”拿到环区的三个环。测试后，我们发现其有着巨大的优点：若先于对方抢到第一个环，则车辆会将上层四个环全部撞出，后面会较为稳定。3.1 版本的车辆吸子进行了大改，使 3.1 具有了稳定快速的角直吸（即可快速吸取角落环堆中最下面的一个），我们便在四环的基础上加入角下方的一个环和本侧中央的环，达成 6 环。

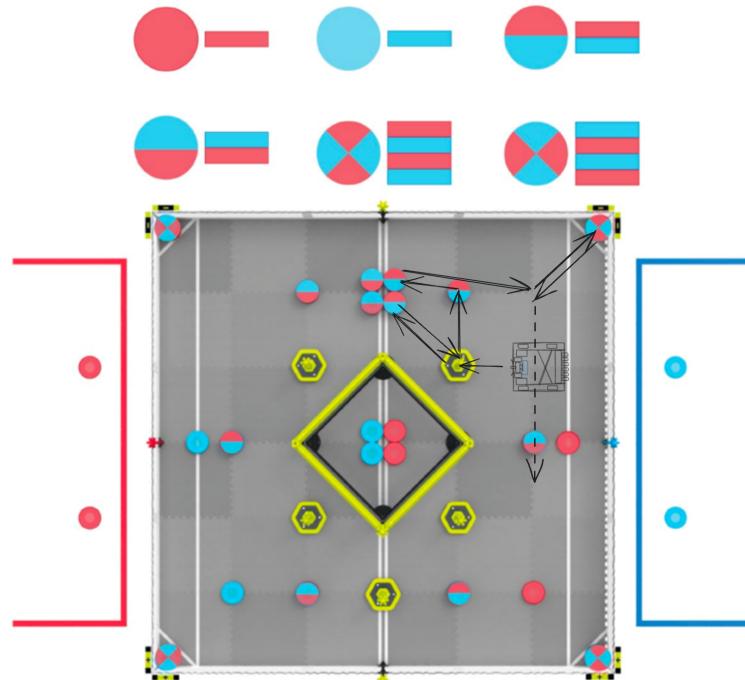


图 35: 蓝方右场 6 环

想要达成 6 环的自动实际上并非一件简单的事情。不同于三环、四环，六环更需要对时间的精准把控与对对方的预测。什么时候走到哪一步都需要经过缜密的考量。就比如此自动是我们综合时间、空间、车辆结构因素、队友的影响、对手的影响以及最终自

动结束时所处的位置价值共同考量的，要力争为手动阶段创造最好条件。——程序员的亲身感受

```
1 void BlueRight6rings() {
2     vex::task Timer(Timer);
3     Catch.set(0);
4     driveStraight(-16 - 1.15, 1); //1.15偏移量
5     Catch.set(1);
6     GyroTurn(-142, 5, 30);
7     //TURNTOHEADING(-142, 1);
8     吸子Control(100); //#1被套上
9     driveStraight(21.5, 0.75); //吸#2环
10    driveStraight(-11.5, 1); //退出拖着#2去#3
11    GyroTurn(-90, 5, 30);
12    //TURNTOHEADING(-90, 0.75); //转向#3环
13    //GYROMOVE(7.5 * 50, 50, -90);
14    //DRIVESTRAIGHT(7.5, 0.5); //到达位置吸环
15    //GYROMOVE(7.5 * 50, 25, -90);
16    driveStraight(15, 0.5); //#不减速
17    //DRIVESTRAIGHT(7.5, 0.25); //减速促使吸环
18    //SLEEP(25); //等待#2、#3吸上
19    GyroTurn(-165, 5, 30);
20    //TURNTOHEADING(-165, 1); //转向#4环
21    driveStraight(11, 0.5); //吸#4环
22    driveStraight(-26, 1); //退出去角
23    Stop1(hold);
24    //GYROTURN(-42.5, 5, 30);
25    turnToHeading(-42, 0.75);
26    //TURNTOHEADING(-40, 1); //转向角
27    GyroTurn(-42, 5, 30);
28    吸子Control(-100);
29    Move(80, 80);
30    sleep(800);
31    Stop1(hold);
32    吸子Control(100);
33    Move(50, 50);
34    sleep(600);
35    Stop1(hold);
36    GyroMove(50,-225, -42);
37    Stop1(hold);
38    GyroTurn(90, 5, 30);
39    吸子Angle.set(1);
40    GyroMove(100,2500,90);
41    吸子Angle.set(0);
42 }
```

6.3.3 蓝方左场/红方右场 3 环第三桩

根据对比赛的先前分析，抢到第三桩可以大大提高胜率、尽力拉大分数差距；再结合 3.0/3.1 版的速度优势，我们决定重写 2.0 版的第三桩程序，保留边发车的特点，再在其基础上加入对第三桩的保护（考虑位置因素）与角直吸，构成了充分利用时间但较为复杂的 3 环第三桩自动。

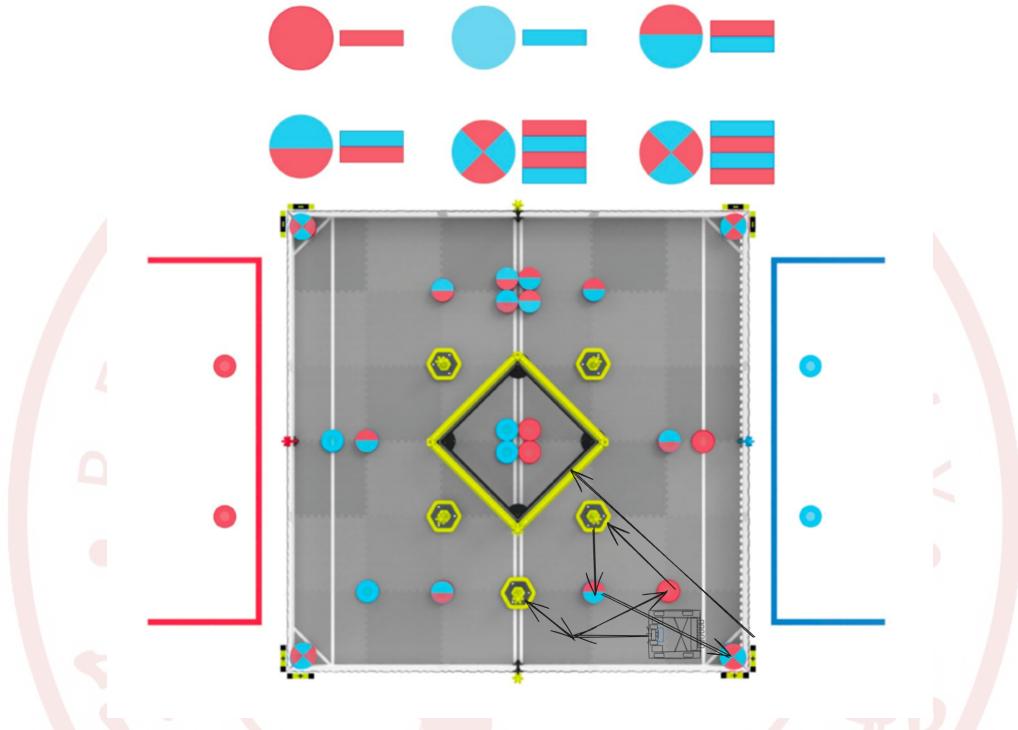


图 36: 蓝方左场第三桩

```

1 void BlueLeftCatch(){
2     horizontalTrackingWheel.setPosition(0, rotationUnits::deg);
3     verticalTrackingWheel.setPosition(0, rotationUnits::deg);
4     x = 0;
5     y = 0;
6     theta = 0;
7
8     vex::task SELECTBLUE>SelectColor_BlueTeam>;
9     Catch.set(0);
10    driveStraight(-30,1);
11    GyroTurn(34,6,30);
12    // TURNTOHEADING(30.5,1,700,25,5000);
13    driveStraight(-13.2,1);
14    Stop1(hold);
15    Catch.set(1);
16    sleep(200); // 抢底座
17
18    driveStraight(12,1);
19    GyroTurn(0,5,30);
20    // TURNTOHEADING(0,1,520,25,3300); // 460,25,3300

```

```
21     吸子Control(100);
22     driveStraight(26,1);
23     GyroTurn(135,5,30);
24     // TURNTOHEADING( 135,1,580,25,6000); //580,25,6000
25     driveStraight(-17,1);
26     Catch.set(0); //加分区附近放底座
27
28     GyroTurn(39,5,30);
29     // TURNTOHEADING( 41,1,569,25,4500); //569,25,4500
30     driveStraight(-28,1);
31     driveStraight(-6,0.5);
32     Stop1(hold);
33     Catch.set(1);
34     sleep(200); //拿自己半场底座
35
36     GyroTurn(100,5,30);
37     // TURNTOHEADING(108,1,569,25,4500); //569,25,4500
38     吸子Control(100);
39     driveStraight(25,1);
40     GyroTurn(19,5,30);
41     // TURNTOHEADING(20,1,569,25,4500); //569,25,4500
42     driveStraight(31,1);
43     吸子Control(-100);
44
45     Move(100,100);
46     sleep(600);
47     Stop1();
48     吸子Control(100);
49     sleep(500);
50     GyroMove(100,-300,52);
51     GyroMove(60,100,52);
52     Stop1(hold);
53     sleep(200);
54     SELECTBLUE.stop();
55     GyroMove(100,-2450,52);
56 }
```

6.3.4 蓝方左场/红方右场 awp

在有世锦赛的名额的比赛当中，众所周知 AWP 的要求变得更加难获得，因此即使不是很适应我们的机型也必须设计出套联队边桩的自动程序。路线由下图中黑线所示。由于 AWP 自动相当于一个比较保守求稳的程序，主要在资格赛中打出，我们便放弃了对于中立 Mobile Goal 的争抢，改为先使用车上的摇臂套联队边桩再在己方 Mobile Goal 上打入三个套环，完成我们的这一套程序。

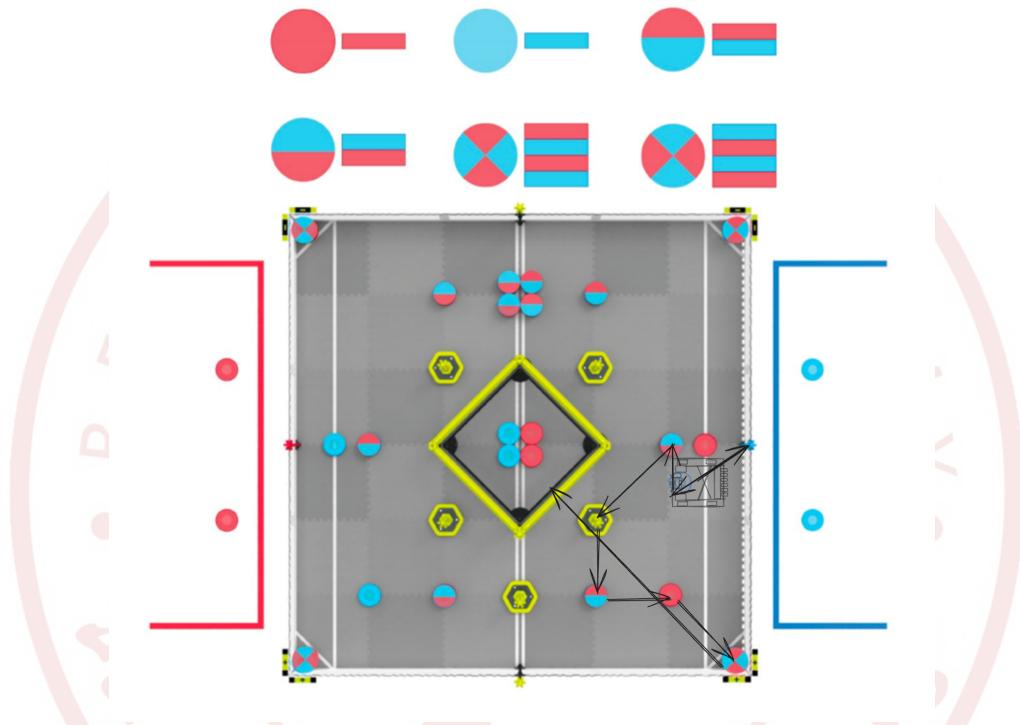


图 37: 蓝方左场/红方右场 awp

```

1 void LeftAWP(){
2     horizontalTrackingWheel.setPosition(0, rotationUnits::deg);
3     verticalTrackingWheel.setPosition(0, rotationUnits::deg);
4     x = 0;
5     y = 0;
6     theta = 0;
7
8     Catch.set(0);
9     // VEX::TASK_SELECTED(SELECTCOLOR_REDTEAM);
10    // GYROTURN(-40,5,30);
11    turnToHeading(-40,1,700,25,5000);
12    ArmControl(100);
13    driveStraight(2.5,1);
14    Stop1(hold);
15    waitUntil(Angle_Arm.angle(deg) > 195 );
16    ArmControl(0); //转身套环

```

```
17     driveStraight(-14.5,1);
18     // GYROMOVE(60,-300,-40);
19     turnToHeading(-90,1,700,25,5000);
20     吸子Angle.set(1);
21     吸子Control(100);
22     driveStraight(10.5,0.8);
23     吸子Angle.set(0);
24     ArmControl(-100);
25     sleep(200);
26     vex::task AWP2RING(AWP2Ring);
27     AWP2RING.setPriority(1);
28     driveStraight(-12.5,1);
29     //吸联队杆下环
30
31     // GYROTURNS(0,5,30);
32     turnToHeading(0,1,569,25,4500);
33     ArmControl(0);
34     driveStraight(-14.9,0.5);
35     AWP2RING.stop();
36     Catch.set(1);
37     sleep(300); //夹底座
38
39     // GYROTURNS(105,5,30);
40     turnToHeading(110,1,557,25,6000);
41     吸子Control(100);
42     driveStraight(26,0.5); //吸2环
43
44     // GYROTURNS(180,5,30);
45     turnToHeading(180,1,557,25,6000);
46     driveStraight(-25.5,1);
47     GyroTurn(42,5,30);
48     // TURNTOHEADINGS(42,1,580,25,6000);
49     吸子Control(-100);
50     driveStraight(16,1);
51     Move(100,100);
52     sleep(400);
53     Stop1(hold);
54     吸子Control(100);
55     sleep(500);
56     driveStraight(-10,0.3);
57     // DRIVESTRAIGHT(-57.2,1);
58     // SELECTRED_STOP();
59     GyroMove(100,-2150,45);
60
61 }
```

6.4 4.0&4.5 版机器

6.4.1 蓝方右场/红方左场抢环 6 环 &5+1 复合自动

考虑到自动阶段环区的争抢将会带来巨大的优势以及 4.x 车辆因速度过快放弃筛环的特点，我们设计了蓝方右场/红方左场抢环 6 环 &5+1 复合自动。该自动车辆将直冲以先获得环区的第 1 环并撞击顶环使本方环跌落至对方的必经之路上，给予对方打入本方环的机会。随后倒退夹住移动桩吸取剩下的 4 个环（含角中 2 个环），最后将摇臂中的预装环打入联队边桩，漂移撞杆此外地，若预装环贴车摆放至车辆一侧，则该环可以被打入移动桩充当顶环。这样最大的优点就是保证了无论如何移动桩的顶端环是本方的，避免造成麻烦。

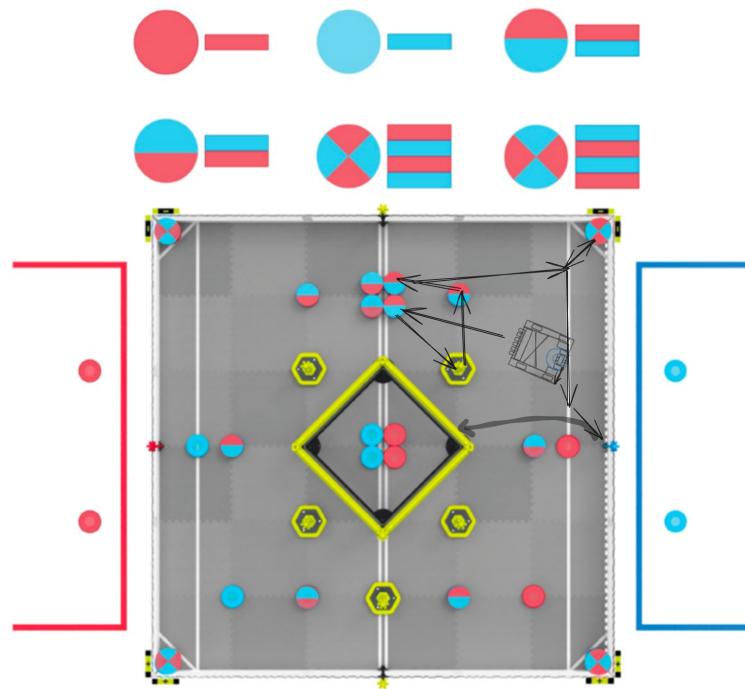


图 38: 蓝方右场/红方左场抢环 6 环 &5+1 复合自动

```
1 void BR5plus1()
2 {
3     Arm.setPosition(85,deg);
4     Catch.set(0);
5     IntakeControl(100);
6     vex::task ARMUP(ArmUp);
7     ARMUP.setPriority(2);
8     GyroMove(50,150,0);
9     driveStraight(40,1);
10    ARMUP.stop();
11    Arm.stop(hold);
12    IntakeControl(100); //吸1环
13
14    turnToHeading(40,1);
15    IntakeControl(0);
16    driveStraight(-12,1);
17    driveStraight(-3,1);
18    Catch.set(1);
19    sleep(100); //夹底座
20
21    IntakeControl(100);
22    turnToHeading(48);
23    // GYROTURN(-48,5,30);
24    driveStraight(26.5,1); //吸2环
25
26    driveStraight(-19.5,1);
27    turnToHeading(87,0.7);
28    // GYROTURN(-82,5,30);
29    driveStraight(17,1); //吸3环
```

```
30     turnToHeading(165,0.7);
31     // GYROTURN(-165,5,30);
32     driveStraight(21,1);
33     turnToHeading(123,0.7);
34     // GYROTURN(-121,5,30);
35     driveStraight(17 ,1);
36     IntakeControl(-100);
37     Move(40,40);
38     sleep(500);
39     Intake.stop(hold);
40     sleep(50);
41     Stop1(hold);
42     IntakeControl(100);
43     sleep(400); //进角吸4
44
45
46     GyroMove(30,-200,123);
47     // DRIVESTRAIGHT(-3,0.2);
48     Stop1(hold);
49     // SLEEP(500);
50     IntakeAngle.set(1);
51     sleep(300);
52     IntakeControl(-100);
53     Move(40,40);
54     sleep(400);
55     IntakeControl(80);
56     IntakeAngle.set(0);
57
58     GyroMove(100,-400,123);
59     turnToHeading(250,0.6);
60     driveStraight(30,1);
61     turnToHeading(205,1);
62     ArmControl(70);
63     driveStraight(7,1);
64     Stop1(hold);
65     waitUntil(Arm.position(deg) > 615);
66     ArmControl(0);
67     GyroMove(100,-400,180);
68     ArmControl(-100);
69     IntakeControl(0);
70     GyroMove(100,-1800,125);
71 }
```

6.4.2 蓝方左场/红方右场半 AWP 自动

这一套自动与 3.5 版本类似，只不过微调了一下角落的吸取，将吸取最底下的环改为吸取 1、3 环形成 1+4 便于对队友减分区侧的补充与加分区的优先争夺。

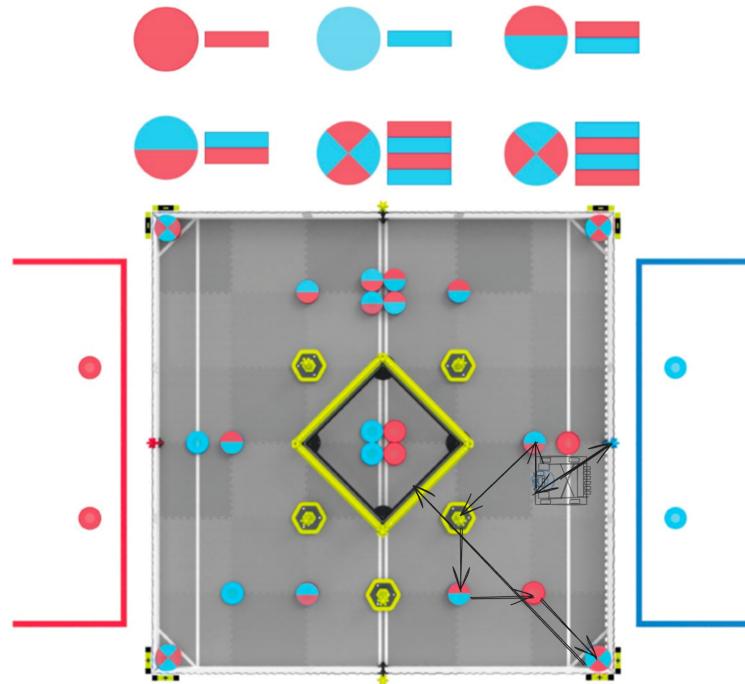


图 39: 蓝方左场/红方右场半 AWP 自动

```
1 void LeftAWP()
2 {
3     Stop1(hold);
4     Catch.set(0);
5     Arm.setPosition(85,deg);
6     ArmControl(80);
7     waitUntil(Arm.position(deg) >615);
8     ArmControl(0);
9     driveStraight(-8,0.5); //挂边桩
10
11    ArmControl(-100);
12    IntakeAngle.set(1);
13    turnToHeading(-55,0.8);
14    IntakeControl(70);
15    driveStraight(14,1);
16    IntakeAngle.set(0);
17    sleep(50);
18    driveStraight(-6,1);
19    Arm.stop(coast);
20    IntakeControl(0); //吸联盟杆下环
21
22    turnToHeading(45,1);
23    driveStraight(-25,1);
24    driveStraight(-5,1);
25    Catch.set(1);
26    sleep(200); //夹底座
27
28    turnToHeading(145,1);
29    IntakeControl(100);
```

```
30     driveStraight(25,1); //吸2环
31
32
33     turnToHeading(78,1);
34     IntakeControl(-100);
35     driveStraight(31,1);
36     // TURNTOHEADING(235,1);
37     // DRIVESTRAIGHT(-20,1);
38     // TURNTOHEADING(95,0.8); //面向角落
39
40     // INTAKECONTROL(-100);
41     // DRIVESTRAIGHT(16 ,0.5);
42
43     Move(40,40);
44     sleep(800);
45     Intake.stop(hold);
46     sleep(50);
47     Stop1(hold);
48     IntakeControl(100);
49     sleep(400); //进角吸3
50
51     GyroMove(30,-200,100);
52     Stop1(hold);
53     IntakeAngle.set(1);
54     sleep(300);
55     IntakeControl(-100);
56     Move(40,40);
57     sleep(400);
58     IntakeControl(80);
59     IntakeAngle.set(0); //吸4
60
61     GyroMove(100,-2700,100);
62     IntakeControl(0);
63     GyroMove(30,-700,100); //后退碰杆
64
65 }
```

7 历次比赛

此部分主要为本队伍对于经历过的比赛的总结与收获，从每次比赛获得经验，尽可能在下一次比赛做得更好。比赛的统计为从 24-25 High Stakes 赛季开始的本支队伍参加的所有场次比赛。

7.1 北京延庆城市赛

这一次的比赛是本支队伍首次参赛，是 7 月 16、17 日在北京延庆举办的 2024-2025 赛季 VEX 机器人城市精英赛。在 2023-2024 赛季的世锦赛结束后，发布了这个赛季的 High Stake 赛题。很明显今年在场上的打法要比去年的赛题难度会大很多，很考验操作手的直觉。对于 7 月 15-17 日的北京城市赛，作为人生的第二场比赛（第一场个人以学校的队号参加的比赛），由于其他事情的安排，实际上也没有什么准备的时间，最终定下的目标就是体验这个赛季的场上情况，在比赛的前两周进行了集训，快速地熟悉规则、搭建机器并熟练操作手法。在这次比赛中用到的机器就是在上文中提到的 1.0 版机器，经过一到两周的训练，对于初代机器的操作已经相对较为熟练了。开赛季的第一场比赛，大部分队伍也和我们队一样没有做好充分的准备，听说甚至有些队伍是比赛前一天才把机器拼装完成的。对于这次比赛，我们对于无论自己的机器还是操作表现都没有很强的信心，最初期待是只要拿到前 30% 左右的晋级大区赛的名额即是满足的结果。可事实上，到场上之后发现对于自己的期待还是有一些可以突破的空间的。



图 40: 延庆城市赛中调试自动

幸运的是第一场比赛十分顺利，与前些日子在基地里的训练相比，这场比赛的强度简直不值一提，这也打消了我的部分紧张情绪。但有着之前的各种比赛的经历，我很清楚自己内心不能放下戒备，而且第二场的两个对手都是实力强悍的机构下的强队，而我的队友好像机器人的功能还没有使用熟练（由于是赛季的第一场比赛，此类情况还是比

较常见的)。在第二场比赛的自动阶段，我的自动程序跑“炸”了，导致我的最高得分环拿成了对方的，这一失误让对方拿下了自动阶段的胜利。(ps: 在这个赛季，自动阶段的6分是至关重要的，基本上直接引导了比赛的走向。) 确认完得分后，我对于这个失误哭笑不得，在场边观众们的注视下充满了尴尬，带着这个情绪并简单调整，进入了手动阶段。这一次的手动阶段我发挥出色，但奈何在1分45秒的时间内队友什么也没干(没有责怪的意思，因为赛季初没有人会完全适应掌握赛场并不犯下失误。有一个对于场上打法的完备的思路是需要经验的积累的)，导致最终还是输掉了第二轮的比赛。



图 41: 比赛结束后留念照片

前6场比赛最终以5胜1负收场，因为AWP的过多丢失，最终排名第5。在经过联队选择后在第二天的复赛稳扎稳打。在过程中甚至“掀翻”了一联。奈何，在最终的决赛中，在赢得了第一场后第二场出现失误，最后时刻由于0.1的规则有1个Mobile Goal上所有的套环都碰到了队友的机器，因此以1分的差距输掉了。第三场，自动程序出现了问题，士气大减，并且在场上因为自己经验的缺少加分区被别的队伍占领，最后实在是乱了阵脚在大优势下输掉了比赛。

经验教训：这次的比赛其实给了我很大的启发。事实上，在赛后对场地的检查中，我发现场上的Mobile Goal的高度出现了差错，以至于实际上可以放7个套环。不能说这是自动失败的主要原因，但是这一点肯定也是造成了影响的。第二，对于加分区的占领极度重要，在场上不能因为“贪心”想多打一个套环就放弃对于加分区的占领，这个其实会导致出现更多的损失。最后一点就是操作熟练度的重要性，因为比赛中一紧张如果操作不熟练就容易乱了阵脚，从而导致出现更多的操作失误等。于是，这次比赛的经验总结主要是以下的几点：

- 赛前细心检查场地
- 及时果断占领加分区
- 巩固操作熟练度

7.2 北京海嘉城市赛

今年秋来的格外的早——八月份就能隐约闻到秋天的气息，九月初空气就已渐凉了——导致早早在中秋节假期期间便可以说是完全入秋了。北京的秋呐……可谓是诡计多端。人在这一片荒凉下变得娇柔做作却又无法尽兴，处在一个进退两难的地步。怎么说进退两难呢？硬要说的话就是人们的思绪大多被秋天的怀伤/孤寂/思念占据了，没有足够的「决心」去静下心办好事。这么想，幸好北京的秋是如此短暂，否则全人类都得栽在这儿了。在深陷泥潭的季节下，却总有人会在夜晚仰望星空。这些人可能是不甘于无所事事/可能是不甘于止步不前/可能是不甘于眼睁睁地看着秋天的叶子飘落。于是乎，经历了许久的沉寂的耐不住寂寞的摩拳擦掌的同学决定开始了前进的步伐，一想到埋藏心底已久的机器人校队愿望这时终于有了实现的机会，「决心」便重新夺回了大脑的所有权……既然有了静下心去办好事的「决心」，一些看似不可能的事就变成了顺水推舟……原本代表着怀伤/孤寂/思念的秋天被赋予了新的意义：蒋 XX 意识到，秋天不仅仅是枯叶的飘落，秋天的变化也象征着人生阶段的过渡，让人对成长/流逝/未来进行反思，激发对人生进一步的思考。所以，我们开始奔跑了起来！

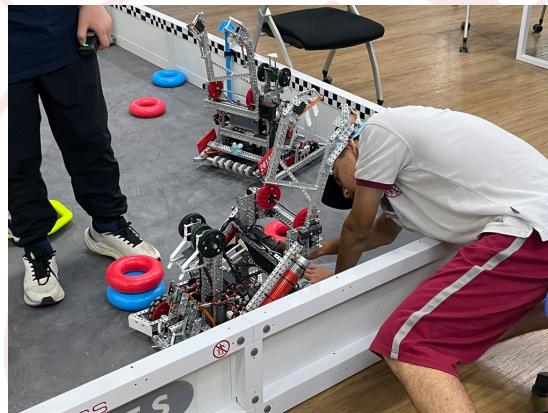


图 42: 北京城市赛准备阶段

于是，在解决了一些看不见摸不着的问题之后，这样就算是正式踏上了旅途。俗话说得好，万事开头难，很快，蒋 XX 就遇到了旅途中的第一个关卡……机器人选修课在得到学校批准后，项目需要依靠许多同学的参与才能顺利展开。然而，一开始蒋 XX 对同学们的兴趣程度并没有太大的把握，尤其是这种在北中比较“新颖”的课程形式。

尽管有些担心，但宣讲会后的报名情况却超出了预期，第一志愿的报名人数比我们想象得要多，这让人颇为惊喜。由于已经与其他校区沟通过，也在他们那边做了宣讲，接下来不得不开始认真考虑人员筛选的事宜，涉及简历筛选、面试安排等多项任务。同时，别的校区内又有新同学希望加入，为了确保所有人都有平等的机会，不得不安排了另一轮面试。

在整个过程中，刘老师给予了机器人团队很多支持，帮助我们一步一步地完成了筛选人员的工作。尽管任务紧迫，但在大家的努力下，这个阶段还是顺利结束了。不过，人选结束后，新的问题又随之而来：如何解决人员的运送问题。不同校区共有 18 名同学需要跨校区参与活动。如果没有校车接送，打车方案虽然是个选择，但出于安全考虑并不能通过审核。当时蒋 XX 人在纽约，这一个个的问题甚至让他夜不能寐。回国后，便希望能线下与周校长沟通这个难题，但出乎意料的是，开学时校长和学校已经从更高层面协调好了校车接送的安排。这最艰难的一关在天时/地利/人和下解决了。于是，机器人团队开始了进入预定轨道。不得不说，创立一件事就是经历风暴和经历风暴后的平静二者不断的交替，在千辛万苦创立了机器人社团之后，也算是进入了目前的第一个风平浪静：至今为止 VEX 机器人选修课可以说是一直在有条不紊地进行着……



图 43: 选修课授课



图 44: 选修课

这就是为什么说我们是那种时时刻刻都仰望着星空的人，就像是常年在海面上驰骋，乘风破浪的海盗一般耐不住风平浪静，即使到了无风无雨的平坦海面，也想要可以地去创造一些海浪来克服。从悲观的角度来说，这样完完全全是被那「不满足」熏醉了；而从乐观的角度来看，这也是一种积极面对生活态度。这不嘛，你瞧，创建了机器人选修课完了的蒋 XX 又开始了新的一波翻越海浪：VEX 机器人北京城市赛——在刚刚结束的北京城市赛中，北京中学的 VEX 机器人队伍在所有参赛队员的共同努力下，表现出色，最终荣获第三名。此次比赛吸引了全市多支强队参与竞争，赛场上气氛紧张而激烈，队伍在小组赛中表现出了极高的竞技水平。

比赛一开始，6353B 车队便士气高涨，迎来了小组赛的五连胜，并且每场都在不断刷新着本次比赛的场上最高得分：42…44…46。这五场比赛展现了队伍出色的团队协作能力和蒋 XX 对机器人操作的娴熟掌控。无论是在场地控制、任务执行，还是应对临场变化，队伍都表现得十分稳健。然而，小组赛的最后一场，当时在场上操作的蒋 XX 显然被前



图 45: 比赛的机器人

五场的连胜“冲昏了头脑”，临时改变了往常的打法，直插场地中间的高挂杆下方抢下方的套环。这是一个鲁莽又贪心的决策，这一个冒险的举动使得吸环装置联动套环装置的链条崩坏，实际上整台车便丧失了功能，即使后来蒋 XX 冷静了下来，和一旁的高 XX 等观察手快速思考对策改变打法，可还是无法改变这一场的结果，以 5 分的劣势中断了资格赛不败的成绩。尽管如此，凭借之前的出色表现，北京中学队伍成功晋级复赛，并且成为了第一联队。



图 46: 比赛照片

复赛一开始，似乎已经明示或暗示地写下了本次比赛的终章。首轮，即使身为第一联队，即使是整场比赛资格赛表现最好的两支队伍联合在了一起，还是在场上出现了致命的失误。自动阶段来自某校的队友出现了失误，手动阶段蒋 XX 又一次鲁莽地（当然也是一次失误）把在加分区的本方的得分底座撂倒了。蒋 XX 整个人都顿时紧张了起来，全身的肌肉都绷紧了，身体微微地颤抖，有些不知所措。好在，某校的队友成为了首轮的战神，在最后时刻将被撂倒的底座重新扶了起来，最终以 6 分的优势赢得了首轮，成功杀入半决赛。然而，在半决赛中，由于自动程序环节的失误，未能充分发挥出机器人的全部潜力，导致关键分失利。随后的手动阶段中，身为第一轮的战神的队友却因为运气也好命运也好出现了长达 30 秒的断联（手柄和机器之间的连接中断），重连后即便两支队伍尝试力挽狂澜，那 30 秒的时间空缺依旧留下了不可弥补的分数分割，这个场上最有

可能获得冠军的联队输掉了半决赛。

虽然未能挺进决赛，但队伍仍然以顽强的拼搏精神，最终取得了第三名的好成绩。乐观来讲，这次比赛的经验无疑为北京中学 VEX 机器人队伍的未来发展打下了坚实的基础。蒋 XX 带领的团队展现了强大的竞争力和潜力，自动程序的失误暴露了一些需要改进的地方，但也为后续的提升指明了方向。未来，队伍将进一步优化机器人设计，完善自动化编程，提升比赛中的稳定性与应变能力。同时，通过更多的训练和比赛经验积累，力争在下一次的赛事中突破瓶颈，冲击更高的荣誉。



图 47: 颁奖环节

7.3 秦皇岛华北区赛

这是 6353 车队创建以来的第二场比赛，也是 6353A 创建以来的第一场比赛。由于北京中学机器人选修课的日渐成熟，有越来越多的成员有足够的资格随队参加比赛。人数的增加既有好处也有坏处，好处是可以更好的分担任务，坏处则是会有一部分人会变成无效职责，甚至产生副作用。这也是为什么分裂出了 6353A 这一支由新生代成员组成的分队。在这次秦皇岛的华北比赛中，6353B 队使用了 2.0 版本的机器，而 6353A 队则使用了较老的 1.1 版本的机器。



图 48: 奖杯和工程笔记

10月20号中午，一行人乘大巴经历半天车程抵达了秦皇岛的检录地点，检录完成后回到酒店开始布置场地与车辆自动程序的测试。自动程序的测试环节异常艰难，明明在原来的基地里测试的时候一帆风顺，也十分稳定，但来到秦皇岛后却总是越界或者总是吸不起来环。队员们都百思不得其解，只好从头开始调整参数，这也导致我们熬到了很晚。而最后，我们才意识到了真正的问题：酒店的地毯太软了，这才使自动的结果又了巨大的出入。于是，我们又删掉了最新的自动参数，打算明早在比赛场地尝试老参数的可行性。

10月21号清晨，一行人乘大巴从酒店返回了比赛场地。比赛正式开始，积分赛阶段的感



图 49: 比赛场地

受并不如北京城市赛那般畅通无阻，也不免是因为所有参赛队伍都整体又了加强，也有一大部分的决定因素是在于队友的感人发挥。运气也是实力的一部分，积分赛阶段我们总是能与一些发挥失误的队伍组队，导致整个上午都处在失败和胜利的反复横跳中。中午，6353B 队排名暂时考后，低于 6353A 队的排名。吃过午饭，可以说是重振旗鼓，主操打算以自己的实力来弥补队友实力上的欠缺。于是，6353B 开始高歌前进，凭借超常的发挥多次拯救队伍于水深火热之中。这一路高歌凯旋，经历连胜，排名上一路上升多名。相反，6353A 队由于经验上的欠缺，有时无法做出快速而正确的抉择，导致下午经历了连败，排名倒退。积分赛结束，6353B 队顺利进入复赛，于 7258k 队联队。而 6353A 队遗憾没有晋级。希望 A 队可以快速吸取这次难得的比赛机会，加强实战的战术演练，在上海全国赛上再接再厉。



图 50: 黄昏时刻

复赛前夜，酒店的地毡问题无法得到解决，只好与 7258k 过完战术后，早早入睡。
10月22号，复赛正式开始。与 7258K 队的配合十分不错，第一场取得了胜利。不幸的是，1/4 决赛就遇到了实力强劲的 957Y 联队，见识到了技术上的实力差距，遗憾落败。不幸中的万幸，下一场比赛险胜，拿下了第三名。



图 51: 赛后合影

这一次比赛让我们学到了很多，意识到了一些场上急需改变的问题比如说自动程序、打发思路等。在这次比赛结束到下一次的国赛还有一段时间，我们需要抓住这段时间继续磨练我们的操作和配合，熟练和机器的联结，争取在未来的国赛走出更大的一步。

上海交通大学 VEX 国赛经历与总结

2024 年，6353B 队带着满腔热情与精心准备，来到了上海交通大学，参加了 VEX 机器人全国赛。这是一场全国顶尖车队汇聚一堂的竞技盛会，也是我们团队磨砺自身、提升能力的重要机会。然而，这次比赛的过程却充满了波折与挑战。

7.3.1 赛程回顾

第一天：起步艰难

比赛的第一天是资格赛的开始。所有人都满怀期待，希望以一个良好的开局为接下来的比赛打下坚实的基础。然而，我们很快发现，赛场上的 wifi 配置存在问题，场控的信号异常不稳定。这直接导致我们的机器人在比赛中移动变得异常缓慢，甚至在关键时刻完全失去了对车辆的控制。这种技术问题使得我们的战术执行效率大打折扣，无论是得分桩的抢夺还是高效的环打击，都远远达不到我们训练时的水平。

尽管如此，我们仍然尽力调整心态，希望在每一场比赛中发挥出团队的最佳实力。经过一天的激烈比拼，我们仅侥幸赢下了一场比赛。这一结果让整个团队感到沮丧，但我们也清楚，首日的表现并不足以决定整个比赛的成败。

第二天：配队与自身的挑战

比赛的第二天，wifi 问题虽然有所恢复，但新的困难接踵而至：配队的运气不佳。我们的联盟队伍实力参差不齐，很多时候无法与我们的战术配合默契，导致得分效率低下。此外，我们自身也面临着许多问题：自动阶段的程序稳定性不足，关键环节的执行存在偏差，车辆的调度和防守策略也显得有些生疏。

尽管我们尽力在每一场比赛中快速调整，但当天的战绩依然不尽如人意，仅仅赢下一局。这种情况让我们感到非常焦虑。即便如此，比赛结束后，我们还是幸运地被选入了 16 强联队，获得了第三天复赛的资格。这也给了我们团队一次重新调整、全力以赴的机会。

当晚：总结与反思

当晚回到酒店，团队进行了深刻的反思与总结。我们逐一分析了两天比赛中的不足之处：自动部分的稳定性问题、车辆在复杂局势下的应对能力，以及队伍内部的沟通与配合效率。我们也针对复赛可能遇到的强劲对手制定了多套应对策略，并与联盟队伍进行了详细的技术磨合。虽然时间紧迫，但每个人都全力以赴，希望能够在接下来的比赛中有所突破。

第三天：顽强的复赛之战

第三天的复赛，对手是实力强劲的 1 联，他们拥有更先进的车辆和更稳定的战术执行能力。然而，这场比赛也是我们最渴望突破自我的一战。

比赛一开始，我们就陷入了困境。车辆的链条在初期阶段意外断裂，导致我们瞬间失去了最重要的攻防工具。面对这样的局势，我们没有选择放弃，而是迅速调整战术，最大限度地利用剩余车辆的功能进行防守和得分。驾驶员冷静地操控车辆，试图用障碍和干扰来拖慢对方的进攻节奏，争取每一分的机会。

比赛进行到最后，双方的比分异常接近。尽管我们最终以微弱的分差落败，但整个团队的表现赢得了观众和裁判的一致认可。比赛结束后，所有队员都深感遗憾，但也为自己的坚持与努力感到骄傲。

总结与收获

这次国赛的经历让我们意识到，比赛不仅仅是对技术水平的比拼，更是对团队默契、心

理素质和应变能力的全面考验。我们在困难面前展现出的顽强拼搏精神，是团队最宝贵的财富。

1. 技术方面的反思

- 自动程序的改进：自动阶段的稳定性直接决定了比赛开局的优势，未来需要投入更多精力优化代码逻辑和调试环境。
- 车辆硬件的可靠性：链条断裂等硬件问题暴露了我们在设备维护方面的不足，下一步需要加强对车辆关键部件的检查与测试。

2. 战术方面的总结

- 配队应变能力：在配队运气不佳时，应快速制定简洁高效的战略计划，以最大化己方车辆的作用。
- 逆境中的调整：比赛中车辆失灵的情况需要提前设计应急策略，例如利用防守牵制对手、争夺关键桩位等。

3. 团队合作的提升

- 赛前沟通与分工：与联盟队伍的磨合非常重要，未来可以尝试更多赛前交流，确保比赛策略能够快速达成一致。
- 内部责任的明确：团队成员需要在比赛中分工明确，尤其是在应对突发情况时，各自的职责要有清晰的规划。

未来展望

尽管这次比赛的成绩不尽如人意，但我们从中学到了许多宝贵的经验。这些经验将激励我们在未来的训练与比赛中更加努力，不断提升自己的技术水平与团队协作能力。6353B 队的每一位成员都相信，只要坚持不懈、不断反思与进步，我们一定能在未来的赛场上取得更辉煌的成绩。

8 VEX 机器人选修课周记

九月九日星期一，机器人选修课第一次授课。

筹备已久的 VEX 机器人选修课终于迎来了本学期的第一次授课，主题是关于 VEX 机器人比赛的整体介绍。为选修课的第一节课，授课的主要目的就是为了让同学们对 VEX 机器人有初步的认识，并激发同学们对机器人比赛的浓厚兴趣。

首先，老师介绍了 VEX 机器人的基本概念。VEX 机器人是一项国际性机器人竞赛，旨在通过搭建、编程和控制机器人来解决实际问题。这项比赛不仅仅考验我们的动手能力，还要求我们具备创新思维、团队合作以及解决问题的能力。通过这次介绍，我了解到，VEX 机器人比赛分为不同的项目和年龄段，包括初中、高中甚至大学生的比赛，每个项目的任务和挑战都不尽相同。

接下来，老师向我们展示了往年的比赛场景和任务，让我对比赛规则有了直观的了解。VEX 机器人比赛的场地通常由障碍物、得分点以及任务目标组成，比赛要求参赛队伍在规定时间内完成尽可能多的任务得分。这不仅要求参赛者设计出功能强大的机器人，还需要在比赛过程中不断调整策略和操作方法。通过老师的讲解，我了解到比赛不仅有操作机器人的人工部分，还有完全由程序控制的自动部分，这让我对编程的重要性有了深刻的认识。这节课让同学们对 VEX 机器人比赛有了全方位的了解，明白了它不仅仅是一场比赛，更是一个锻炼我们综合能力的平台。



图 52: 选修课授课 (1)

九月十二日星期四，机器人选修课第二次授课。

同学们经过第一次的机器人选修课后，都或多或少地对机器人已经产生了或多或少的兴趣。因此不难看出，这第二次的机器人选修课同学们也是怀揣着十足的干劲进入的教室。选修课的人数很多，包括来自国际部的高中生们，也包括来自初中部的同学们：为了保证全体同学的参与度，以及为了使同学们逐渐意识到团队合作，默契配合的重要性。以及利于日后整体授课的便捷，老师们决定在第二次授课这个关节环节上培养同学们以上的品质，这有利于日后学习配合，比赛配合。那么话说回来，这个任务便是组件比赛场地。

在开始之前，老师在大屏幕上放映了搭建的图纸，告诉了同学们这也是具有一定的挑战性的，不出所料，同学们在看到建造材料后才意识到场地比想象中要大的很多。分工是在团队配合中不可或缺的一环，这不仅考验决策者的整体意识水平，也决定了这个团队在未来工作的效率：在完整的场地中，包括了很多部分，地垫，侧墙，高架等等，其中铺地垫和分侧墙的工作较为简单，便交付给了能力/力气较弱的初中部同学，而较为困难的高架拼接工作便交给了能力/力气较强的国际部高中同学，而多余的剩下来的同学于是被安排到了套环桩的拼接工作，至此，每一位同学便有了自己的工作。以上便是人员分工的分配工作。

人员分配完成之后，随着老师的一声令下，同学们便纷纷投入了工作。随着一群初中生蜂拥而至，地垫便有了雏形，随着力气大的高中同学帮助调整之后，地垫便完成了拼接，与此同时另一边的高中同学们，一部分呈地基状，俯卧在地板上支撑着高架的竖梁，另外两个人举起方形横截面使其插入横梁，又另起人握住两侧进行更加精确的对接……尽管人员的分配已经十分合理，但仍然有提前做完的人员无所事事——这便是人多的弊端，也是之后授课需要注意的事情；相反地，人多也有人多的好处，同学们赶在下课前便完成了偌大场地的拼接。



图 53: 选修课授课 (2)

九月十九日星期四，机器人选修课第三次授课。

选修课不能仅仅停止于“参与”，更重要的还是让同学们能够通过选修课学到自己真正感兴趣的东西。然而，通过前几次的课程看来，想到做到这一点并非像普通的课堂授课一样简单，只要有单纯的知识输出就可以了；对于我们的选修课来说，同学们之间的机器人基础十分的不一致，其中包括已经有多年机器人学习/参赛经验的同学，同时也包括完全零基础的同学。至于说为什么依然通过这些零基础的同学来上课，我们认为培养兴趣/激励兴趣是最重要的，就算是零基础的同学也完全可以，但这并不代表我们的课程很佛系，其中的一部分高水平同学也是朝着世锦赛为最终目标而学习奋斗的。言归正传，同学们之间的参差不齐还源于之前说到的年级不一致，随之而然的也是认知水平的不一致，仍然，我们的选修课秉持着有教无类，尽可能为所有同学提供学习的机会。

因此，既然不能从同学那里挑毛病，就只好改改授课了。于是，经过深思熟虑，从第三次授课开始了分层授课：一部分有过机器人基础的被分在 A 班，剩下的零基础的则分在了 B 班。其中 A 班的授课内容紧密连接着 VEX 机器人的比赛，内容包括规则分析，编程，图纸设计等等；而 B 班的授课内容相比较为基础，老师们打算从较为低级别的机器人比赛入手，授课内容也更多的为机器人实操搭建，一些较高难度，高精尖的内容并没有教授。值得注意的是，课程的分班并没有取决于年级的差异，A 班中也包含了很多具有丰富比赛经验的初中生，反之亦然。既然课程的分层取决于学生的学习水平，那么当然，如果 B 班的同学表现优秀的话，也可以进入 A 班进行学习，并且有机会代表北京中学参赛。

于是，在第三次的分层授课中，AB 两班分别学习了两个比赛级别的规则。根据这次分层授课观察，这种方案似乎可行性很高。

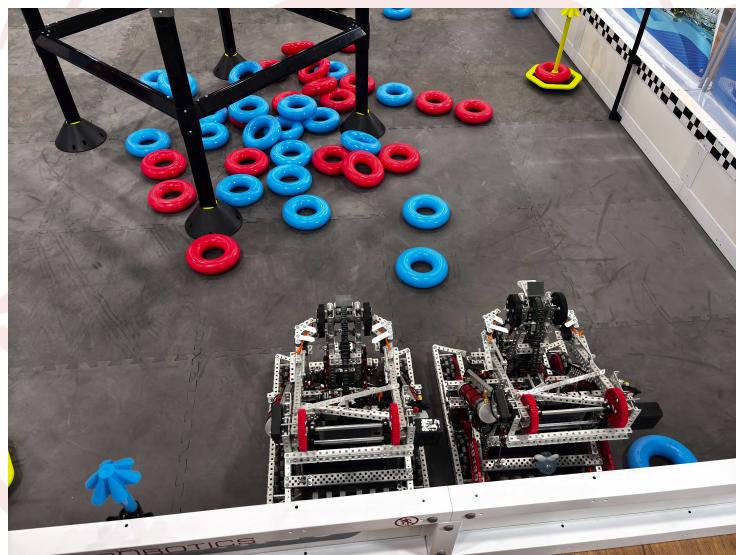


图 54: 选修课授课 (3)

九月二十三日星期一，机器人选修课第四次授课。

本周是 VEX 机器人的第四次授课，也是我们参加北京城市赛后的第一次回到教室的时间。比赛刚结束，我们都还沉浸在比赛的回忆中，虽然我们在城市赛中获得了第三名的好成绩，但我们深知还有很多需要改进和反思的地方。因此，这节课的重点内容是比赛录像的回放与总结反思。老师一开始播放了我们在比赛中的几场关键对决，尤其是小组赛的五连胜和半决赛中的失利。通过录像的回放，我们不仅再次感受到了比赛时的压力，也更清楚地看到了比赛中我们操作和策略上的优点和不足。

通过小组赛的五连胜回放，我们明白了自身车队的强势所在，而最值得反思的还是半决赛的失利。在这场比赛中，自动部分的失误导致我们失去了早期的优势。这一环节暴露出我们的程序调试和场地适应能力还有待加强。

经过反思讨论后，我们也为未来的比赛制定了几个目标：第一，要进一步提升自动程序的准确性，确保在比赛中的每一个环节都能发挥作用；第二，在临场应变和团队沟通上加强练习，确保每个队员都能在关键时刻迅速调整策略；第三，要继续优化机器人的设计，使其在面对复杂情况时更具灵活性。这次回顾不仅帮助我们深入了解了自己的优势和不足，也为我们的未来的训练指明了方向。老师最后鼓励我们说：“比赛的胜负固然重要，但更重要的是在每次比赛后，我们都能够从中学到东西，并不断进步。”



图 55: 选修课授课 (4)

九月二十六日星期四，机器人选修课第五次授课。

不知不觉，VEX 选修课已经进行了快一个月了，同学们在这一个月里学习的机器人知识，看了那么多次选手驾驶机器人的录像，如今早已手痒痒了，急需“过一次驾驶机器人瘾”，这不吗，机会就来了：

这次课程开始前，老师特意为我们详细讲解了机器人的控制原理以及驾驶技巧。尽管此前我们已经在搭建和编程的过程中了解了机器人的工作机制，但这一次的重点是如何灵活地驾驶机器人，并在短时间内作出反应，及时调整操作。为此，老师先将我们分成几组，每组轮流体验机器人操作。同学们上手时，最初都在适应控制器的灵敏度，了解机器人如何在场地上进行精准的移动。尽管操作看似简单，但实际上，每一步的控制都要求我们保持高度的专注力，特别是在面对狭小空间和需要快速转向的时候。

经过一段时间的适应，令人期待的模拟对抗赛正式拉开帷幕。两两对抗的比赛让同学们紧张而兴奋，虽然大部分同学都是第一次亲自动驾驶机器人，但比赛的激烈程度却丝毫不亚于以往的正式赛事。俗话说得好，初生牛犊不怕虎，正是因为是第一次上手操作，同学们反而展现出了极大的勇气和创造力，毫不犹豫地尝试各种复杂的动作，有时甚至是高难度的突破和抢分。这种无畏的精神不仅使得比赛更具挑战性，也为观众带来了许多出乎意料的精彩时刻。

围观的同学们全神贯注，目不转睛地注视着赛场上那迅速移动的机器人。一时间，欢呼声、惊叹声此起彼伏，每一轮对抗都让人屏息凝神。机器人在赛场上高速穿梭，时而精准吸起环，时而灵活避开对手的阻挠，争分夺秒地进行得分操作。虽然这是一次模拟对抗

赛，但场上的竞争气氛丝毫不逊色于真正的比赛。每位选手都在不断尝试各种策略，有时快速进攻，有时巧妙防守，不断调整自己的操作方式以应对对手的攻势。这一过程中，队友之间的配合与默契也显得尤为重要，同学们通过不断的沟通和协调，学会如何在瞬息万变的场上迅速作出判断。

本次授课不仅让我们熟悉了机器人操作，也让同学们意识到团队合作和临场反应的重要性。通过这次体验，同学们更加理解了机器人驾驶的难度和乐趣，也对未来的比赛充满了期待。希望在接下来的课程中，我们能够不断提升操作技巧，进一步优化我们的比赛策略。在比赛过程中，同学们通过沟通协调，迅速调整策略，不仅提高了同学们的兴趣，也让老师们对同学们基本的驾驶水平有了了解，以便为参赛队伍源源不断地提供新的人才。未来的挑战还很多，但我相信，通过不断的练习和总结，我们会在接下来的比赛中取得更好的成绩。

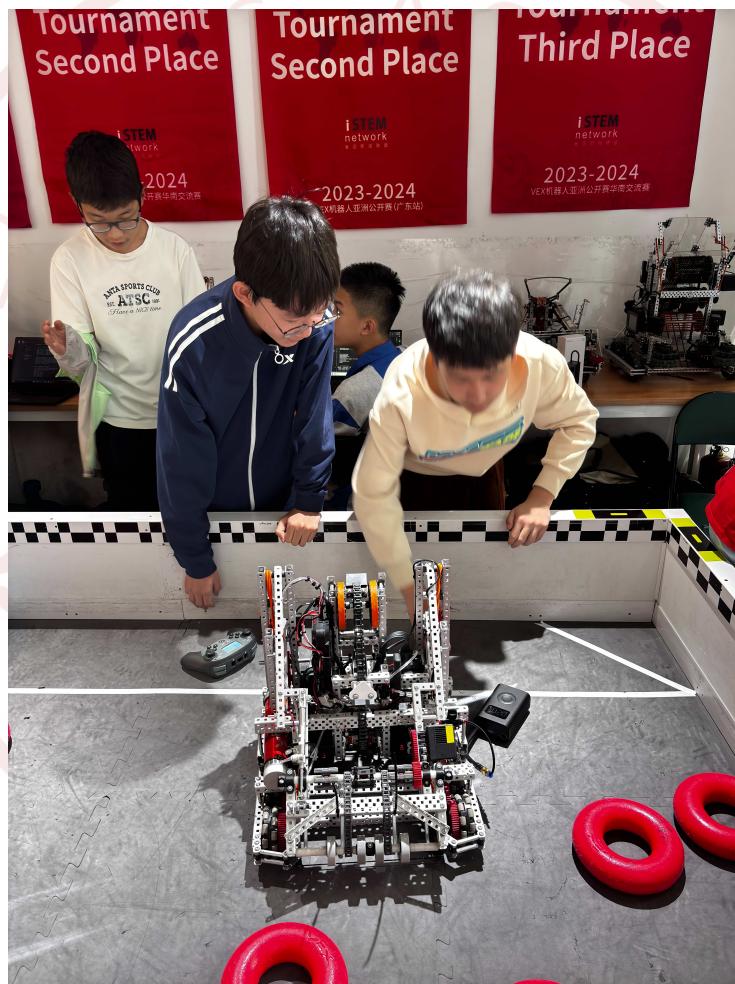


图 56: 选修课授课 (5)

十月十日星期四，机器人选修课第六次授课。

国庆假期结束后的第一次上课，标志着我们重新回归了更加结构化的授课方式，也就是模拟对抗赛结束后，终于回到了分班上课的授课制度。这一变化让同学们既感到熟悉又有些期待，因为分班上课意味着教学内容会更加集中，针对性更强，大家能够按照各自的进度和需求进行深入的学习。

与此同时，授课内容也发生了重要的转变。我们正式进入了课程的第二阶段，从此前侧重的机器人操控与基本编程，转向更加复杂的程序设计和机器人结构的设计学习。相较于第一阶段的动手实践，第二阶段要求我们具备更多的理论知识，同时也需要更加精密的思考和规划。此次上课的核心内容是 Onshape 的学习——这是一个专业的机器人设计网站，能够帮助我们用数字化的方式进行机器人模型的搭建和优化。

由于这是我们第一次接触 Onshape，老师在课堂一开始便向我们详细讲解了这个平台的重要性。Onshape 不仅是一个设计工具，更是我们未来设计和优化机器人结构的必备技能平台。老师先从基本界面开始讲解，包括如何创建新项目、如何使用平台上的工具进行设计等。课堂上，老师还特别强调了文件共享功能的重要性，因为在未来的我和团队合作中，能够有效地进行在线协作将会大大提升我们的工作效率。这节课是我们第一次要求同学们自带电脑上课，因此不可避免地出现了许多问题。有些同学的设备配置不够，运行 Onshape 时出现了卡顿；也有的同学因为网络问题无法顺利进入平台。此外，由于每个同学对计算机的熟练程度不同，很多人对于注册账号、操作软件的步骤不太熟悉，导致课堂的进展比预期缓慢了许多。尽管老师尽力引导我们，但由于问题较多，最终这一节课主要集中在帮助大家完成账号注册、了解基本功能以及进行文件共享上。虽然教学进度较慢，但这也是为接下来的深入学习打下了基础。

总体而言，这次课程让我们对未来的学习方向有了一个清晰的认识。Onshape 的学习虽然起步缓慢，但只要我们克服了最初的技术难题，相信在未来的课程中，我们会逐渐掌握这个工具，并能用它来设计出更为复杂和精确的机器人模型。希望在接下来的课程中，大家能够更加熟练地使用设备，避免技术问题的干扰，这样我们就可以更高效地进行学习，并尽快进入实际的设计环节。



图 57: 选修课授课 (6)