**《模块化机器人综合设计实践》**

**实验报告**

学生A姓名：曹 啸 学号：2017080106018

学生B姓名：文博正 学号：2017080106017

学生C姓名：欧亚非 学号：2017080106010

一、控制系统设计

1.1 机械结构设计

1.1.1 组成模块

所设计的小车模块有底盘，驱动模块，转向模块，控制模块，传感模块和供电装置组成，如图1-1至图1-6所示。

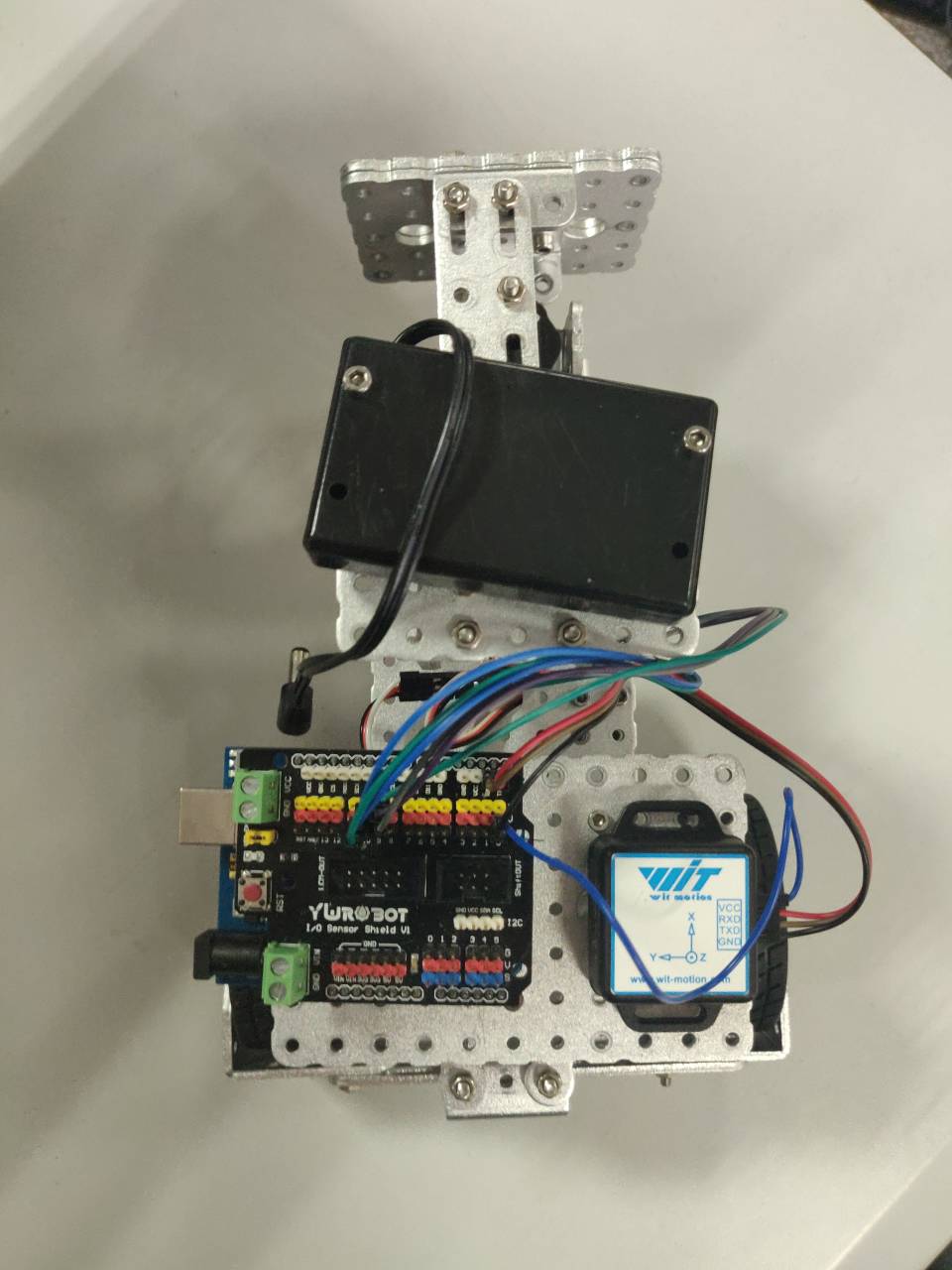


图1-1 顶视图

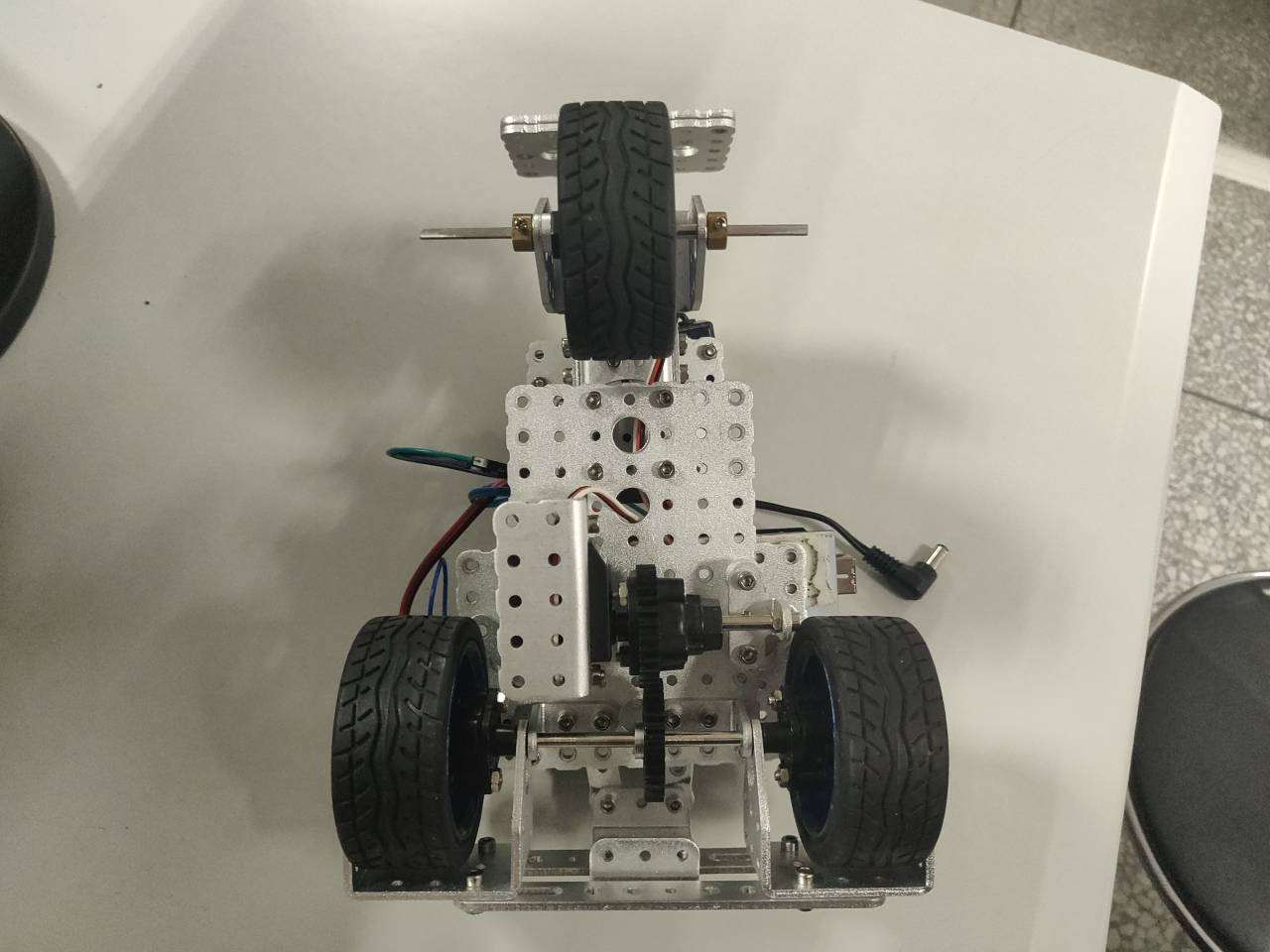


图1-2 底视图

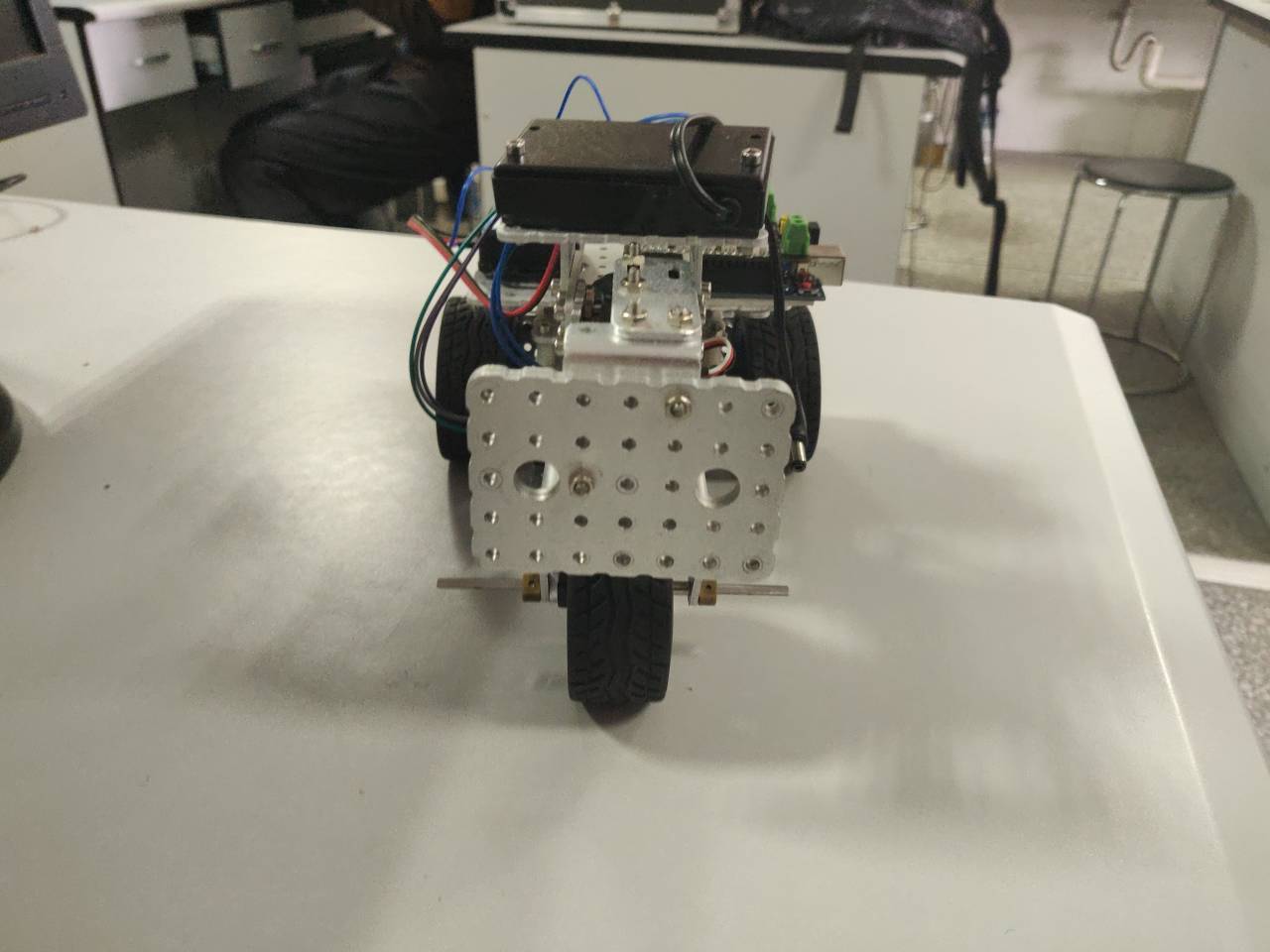


图1-3 前视图

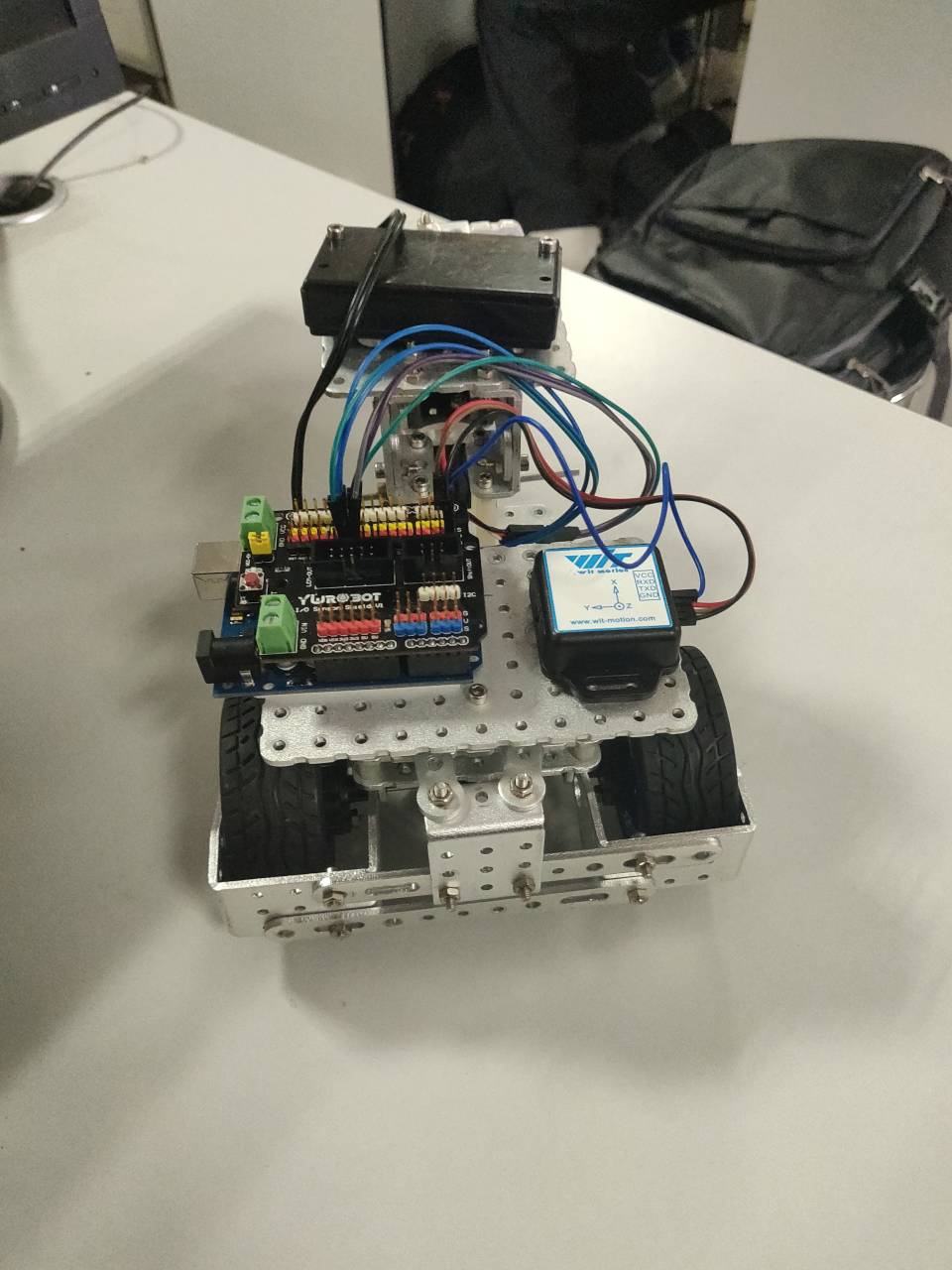


图1-4 后视图

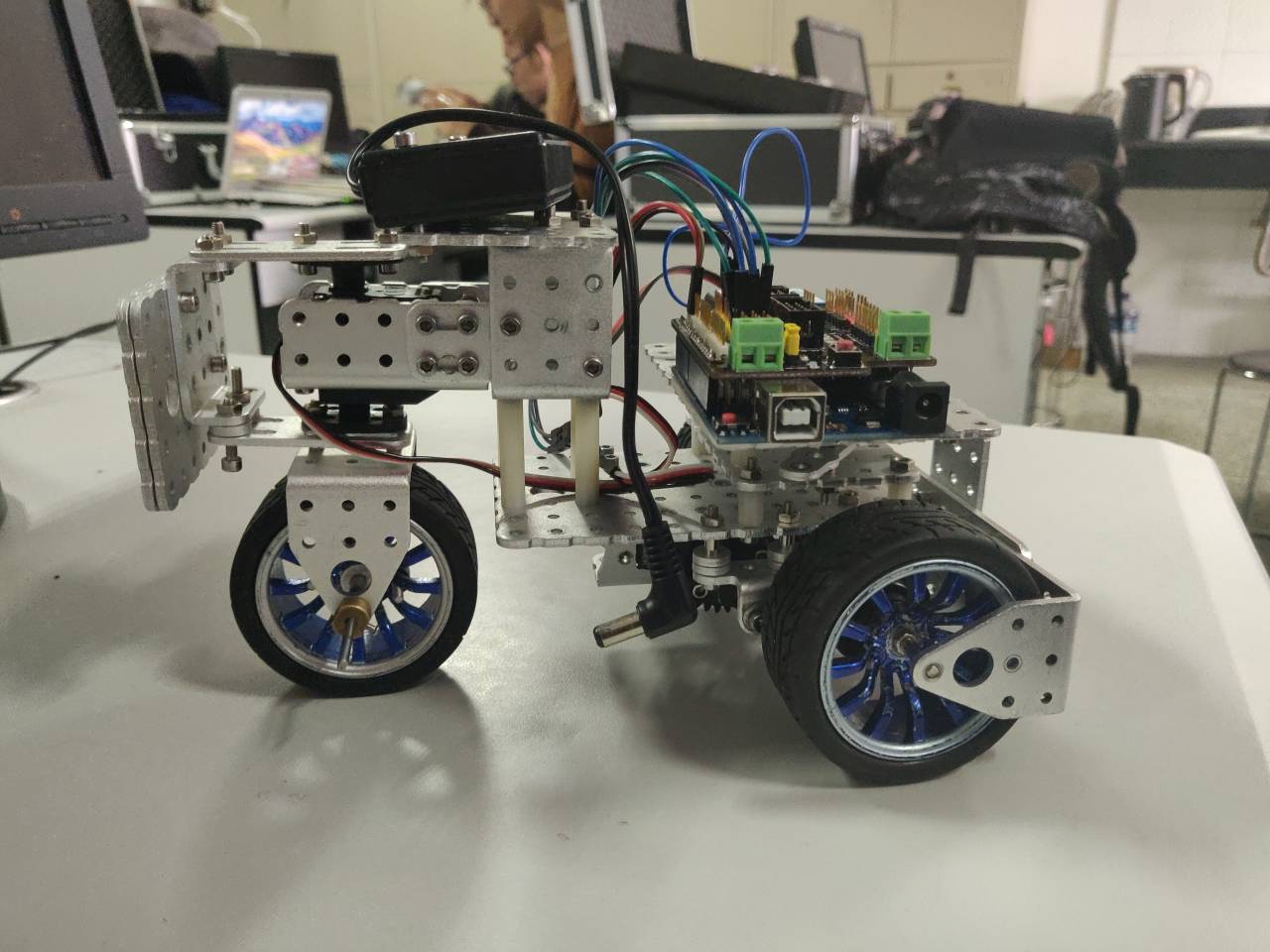


图1-5 左视图

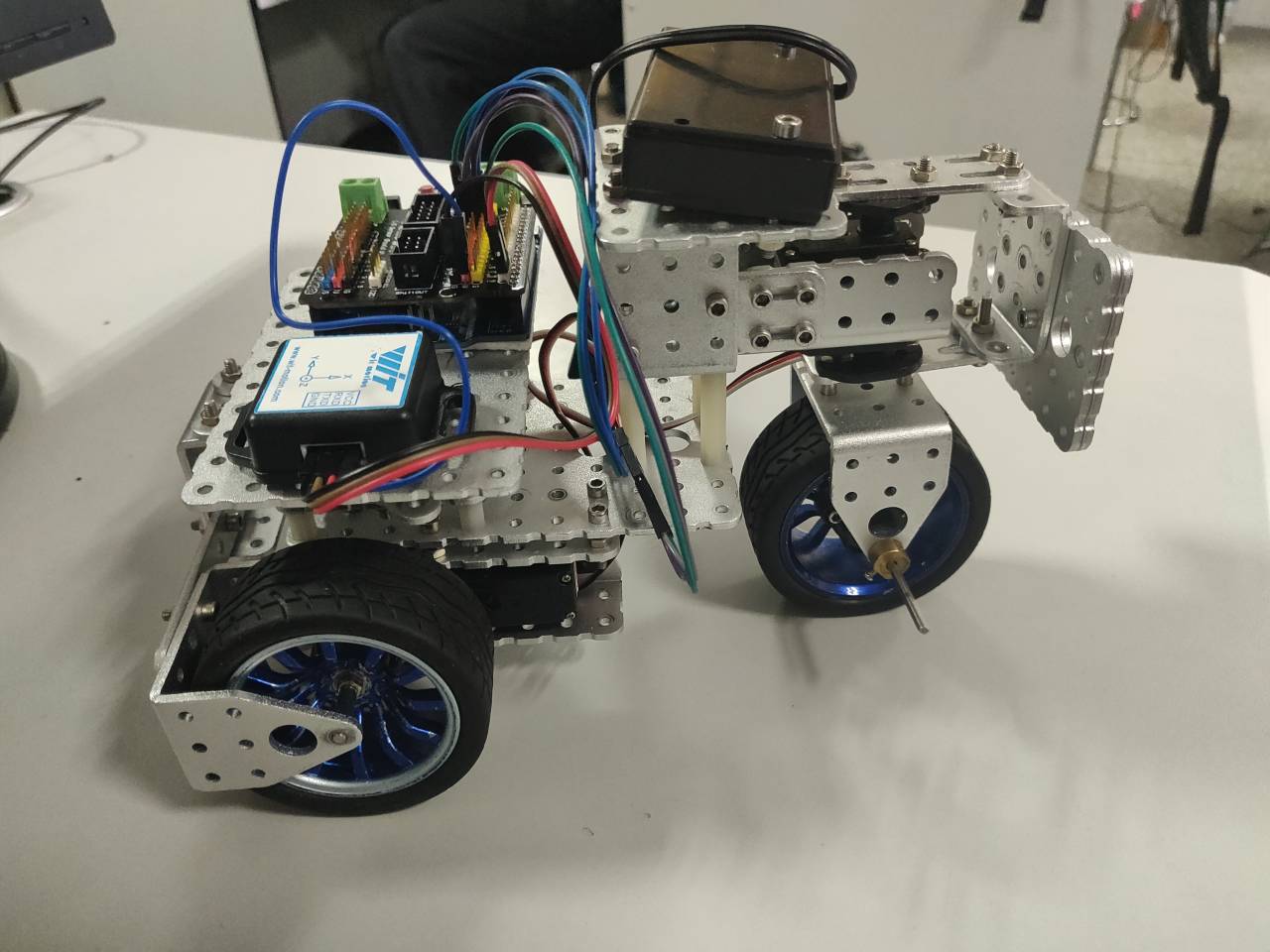


图1-6 右视图

底盘采用了双层结构，底层安装驱动模块，顶层固定控制器和传感器。底盘前端连接转向模块。所有固定均采用螺栓螺母连接，保证了结构的稳定性。

驱动模块以圆周舵机为核心，通过齿轮传动将转矩稳定传递到轴上。两后轮通过轴连到一起，避免了双舵机分别驱动时可能因为差速而导致的方向偏离问题，但没有涉及悬挂机构，因此转向时会存在摩擦阻力，但在此次任务中影响不大，可以忽略。轮外端和舵机外端均为简支梁结构，可以保证轮和传动齿轮不会发生轴向偏移，也解决了应力集中问题。驱动模块的导线可通过底盘上的孔连接到控制器。

转向模块以标准舵机为主，可以实现大范围扇形旋转。舵机两端也是简支梁结构，可稳定提供转矩。前轮固定于一心轴。转向模块仅提供方向的控制，因此不与驱动模块连接。标准舵机的导线可直接与底盘上的控制器相连。

控制器和传感器并排固定于底盘上部，所有的接口和引脚都可以很容易的进行连接。

供电装置采用实验室提供的充电电池，固定在车身前部。初期的方案中，这个模块固定于车竖直的后板，但这会导致上坡时可能出现后翻，因此更换其位置。实际上由于车重心非常低，重量主要集中在驱动模块，因此这一改动并不会导致车的重心过高。

1.1.2方案介绍

在实现小车的设计过程中，我们尝试了若干种方案。最终在四轮（差速控制转向），履带（差速控制转向）和三轮（前轮控制转向）中选择了以前轮控制转向的三轮车方案。

四轮（差速控制转向）的车身稳定，搭建方便，但是受限于材料，无法轻易搭建悬挂机构，因此必须考虑差速控制转向。但两个舵机的速度不能保证完全一致，且转向时会产生较大摩擦阻力，还会增加程序的设计难度，因此我们没有采用这种方案。

履带（差速控制转向）相比于上一种方案，转向时的摩擦阻力较小，且机构稳定。但履带本身的不平整会导致传感器接受到多余的信号，增加程序的设计难度和调试时间，因此我们没有采用这种方案。

三轮（前轮控制转向）可能存在侧翻问题，但设计时将重心压得很低，因此不存在这种情况。同时仅用一个圆周舵机和一个标准舵机，拆装方便，易于控制，结构稳定，转向时后轮的摩擦阻力也可以忽略不计。因此我们最终采用了这种方案。

1.2传感系统设计

为了达到小车能够寻找跷跷板平衡位置、摆放在平台上能自行走上跷跷板这两个要求，传感系统只需使用一个九轴姿态传感器模块JY901。JY901九轴姿态角度传感器如图1-7所示。

因此，传感测量只有一个位姿检测模块，由一个JY901九轴姿态角度传感器组成。JY901传感器的主要性能参数如表1-1所示。



图1-7 JY901九轴姿态角度传感器

表1-1 JY901九轴姿态角度传感器主要性能参数

|  |  |
| --- | --- |
| 电压 | 3.3V-5V |
| 电流 | <25mA |
| 体积 | 15.24mm X 15.24mm X 2mm |
| 焊盘间距 | 上下100mil(2.54mm)左右600mil(15.24mm) |
| 测量维度 | 加速度：3 维，角速度：3 维，磁场：3 维，角度：3 维，气压:1 维（JY-901B）， GPS：3 维（接 GPS 模块） |
| 量程 | 加速度:±2/4/8/16g（可选），角速度:±250/500/1000/2000°/s（可选），角度±180° |
| 稳定性 | 加速度：0.01g，角速度 0.05°/s |
| 姿态测量稳定度 | 0.01° |
| 数据输出内容 | 时间、加速度、角速度、角度、磁场、端口状态、气压等 |
| 数据输出频率 | 0.1Hz-200Hz |
| 数据接口 | 串口/I2C |

JY901姿态角度传感器的输出数据包括时间、加速度、角速度、角度、磁场、端口状态、气压等，但是传感系统所需要的数据主要是加速度、角速度、角度，对于其他不需要的数据包，不进行读取。

在上一节中，已经讨论了传感器安装的位置和方法。不难看出，传感系统模块中的JY901传感器所测量的数据主要是小车当前运动的加速度、角速度、角度数据。具体来说，为了完成小车从平台上自动走上跷跷板的任务，以及保证小车在跷跷板上走直线，需要使用JY901获得Z轴角度数据来判断车头朝向；为了对当前跷跷板的运动状态进行判断、控制小车运动找平衡，需要使用JY901获取Y轴角度、角速度、Z轴加速度等数据。

由于传感系统只有一个模块，而且只需要测量加速度、角速度、角度数据，传感系统的设计要求相对较宽松，主要的设计要求包括以下两点。第一，跷跷板的摆动角度很小，因此传感系统所得数据需要非常精确，且噪声不能太大；第二，跷跷板摆动速度很快，姿态数据变化也很快，所以传感系统数据传输的速度不能太慢。通过设置JY901的相关参数，这些要求很容易实现。

在使用JY901姿态角度传感模块的时候，需要注意以下几点影响因素。第一，虽然JY901的官方说明文档中显示，它的三轴都没有角度漂移，但实际使用时，角度漂移是不可避免的，因此传感系统设计还需要考虑到漂移的因素，需要保证在每次使用前都手动校准角度；第二，由于安装时不能保证传感器在小车上处于绝对水平，因此还需要将小车摆放在水平位置，按当前位置校零；第三，使用九轴算法时，传感器通过磁场解算角度、角速度数据，因此磁场干扰对传感器的影响很大。必要时，可以改用六轴算法解算姿态。

1.3执行器设计

在整体的硬件性能上，我们提出以下要求：方便转动方向的调节；保证足够的爬坡能力，并且不发生打滑；保证小车以当前方向行驶时不发生偏移或偏移量较少。

基于硬件要求，我们提出以下三种方案:

方案一：前轮转向，后轮驱动。执行器选择一个180°舵机和一个360°舵机。其中，180°舵机控制前轮转向，360°舵机通过齿轮与轴控制后轮转动。

方案二：前轮转向，后轮驱动，执行器选择一个180°舵机和两个360°舵机。其中，180°舵机控制前轮转向，两个360°舵机分别单独控制对应后轮的转动。

方案三：后轮驱动，差速转向。执行器选择两个360°舵机，分别单独与左右后轮相连。

经比较，方案一的优点在于调节转动方向的灵敏性和行驶时较小的方向偏移，但爬坡能力相对较弱。方案二弥补了方案一驱动能力不足的缺点，但由于左右舵机在制造时的差异，难以保证两个后轮转速相同，从而存在一定的行驶方向偏移。方案三使用差速转向，小车转向性能受物理环境影响较大，且调整方向的复杂性略高于方案一和方案二。最后，我们决定使用方案一，通过优化小车的机械机构来减轻驱动的负载。

在使用执行器（舵机）时，我们需要对两种执行器进行单独控制。其中，对于180°舵机，需要控制它的转角。对于360°舵机，需要控制它的转速和转向。使用时，需要对舵机的参数进行预校正。180°舵机在安装后需要校准相对于跷跷板的正方向，360°舵机在安装前需要测量在Arduino舵机函数库中的转动停止时的PWM占空比。

1.4控制算法设计

小车寻找跷跷板平衡的控制系统框图如图1-8所示。传感器用来测量当前小车的姿态参数，由控制器进行数据的处理，输出数据控制舵机的转动方向和速度，以此控制小车的前进、后退及其速度，最终由小车在跷跷板上的运动，控制跷跷板摆动，此时小车也随着跷跷板摆动，再将角度数据反馈给传感器。最终实现跷跷板的平衡。



图1-8 控制系统框图

基于控制系统框图，我们对控制系统提出以下性能要求：响应速度较快；超调量较小；震荡次数较少。由于跷跷板触地对整个系统的影响较大，所以控制系统对响应速度和超调量具有较高要求。

在采用1.3中的方案一作为执行器方案后，控制系统将受到以下外界干扰：跷跷板转动时惯性的影响，跷跷板转动轴摩擦力的影响等，同时，传感器在使用过程中会随时间产生较大的漂移量。惯性和摩擦的影响可以通过提高响应速度和减小超调量来削弱，传感器的漂移可以通过滤波算法进行校正，

控制算法分为两部分，传感器信号处理与舵机驱动。由于传感器在测量数据时产生漂移，我们使用滤波算法进行校核，其中，常见的滤波算法有均值滤波、互补滤波、卡尔曼滤波等。

均值滤波的优点在于计算简单，通过计算多次数据的平均值，将平均值作为真实值。相应地，对滤波效果一般，对噪声的抑制能力较弱。

互补滤波是通过基于传感器对角度、角速度、加速度的测量灵敏性不同而设计的滤波算法。传感器对小车的加速度比较敏感，取瞬时值计算倾角误差比较大；而陀螺仪积分得到的角度不受小车加速度的影响，但是随着时间的增加积分漂移和温度漂移带来的误差比较大。由于加速度计的低频特性较好，加速度计的角度可以直接得出，所以没有累计误差，长时间内性能较好，可以通过高通滤波来抑制低频。而陀螺仪的角速度积分由于误差的积累，长时间的性能较差而短时间内性能较好，通过低通来抑制高频。互补滤波则需要选择低通与高通的切换频率。互补滤波的公式为

卡尔曼滤波是一种利用线性系统状态方程，通过系统输入输出观测数据，对系统状态进行最优估计的算法。由于观测数据中包括系统中的噪声和干扰的影响，所以最优估计也可看作是滤波过程。卡尔曼滤波需要对系统进行建模，同时需要干扰噪声的方差等数据。



图1-9 控制算法流程图

比较三种滤波方法，均值滤波计算简单但无法达到理想滤波效果。卡尔曼滤波较为精确，但对建模的精确度较高，受模型准确性的影响较大。同时，噪声方差等数值的要求难以满足，设计难度较大。互补滤波具有较好的滤波效果，同时相对于卡尔曼滤波，设计要求相对简单。综合考虑，我们决定使用互补滤波作为传感器信号处理的滤波算法。

在驱动部分中，我们使用PID算法作为调平衡的核心算法。在PID算法中，定值和实际输出值构成控制偏差，将偏差按比例、积分和微分通过线性组合构成控制量，对被控对象进行控制。

PID控制器各校正环节的作用如下：

比例环节：即时成比例地反应控制系统的偏差信号e(t)，偏差一旦产生，控制器立即产生控制作用以减小误差。当偏差e=0时，控制作用也为0。因此，比例控制是基于偏差进行调节的，即有差调节。

积分环节：能对误差进行记忆，主要用于消除静差，提高系统的无差度，积分作用的强弱取决于积分时间常数T，T越大，积分作用越弱，反之则越强。

微分环节：能反映偏差信号的变化趋势(变化速率)，并能在偏差信号值变得太大之前，在系统中引入一个有效的早期修正信号，从而加快系统的动作速度，减小调节时间。

同时，通过比较PD、PID、PI的使用效果。考虑到系对超调量和响应速度的要求较高，我们最终使用PI，该算法在满足方案要求的同时，能够减少震荡。

控制流程图如图1-9所示。

开机后，系统初始化，定义并初始化工作所需的变量，初次读取JY901数据。标志位在初始化时被定义为False，通过判断标志位来选择小车工作模式。第一种工作模式是从平台转向、爬坡走上跷跷板，第二种工作模式是走上跷跷板以后寻找平衡点。

在爬坡模式中，通过不断读取Z轴角度，和设定的标准值对比来实时修正小车的偏转。同时，不断读取Y轴角度，通过判断Y轴角度是否小于设定值来修改标志位。当标志位为False时，重复执行爬坡模式。

在调平衡模式中，根据9轴传感器的摆放位置，读取Y轴的角速度和角加速度作为调平衡算法的其中两个输入参数，其输出用来调节驱动舵机转速。当达到平衡并持续5秒之后，LED灯亮起，开始执行下坡任务，启动后轮，后轮倒转。

二、控制系统的制作与调试

2.1机械结构的制作与调试

我们搭建小车经历了数个阶段，每次都解决了问题，保证了最终版的稳定性。以下为小车的搭建与改良过程：

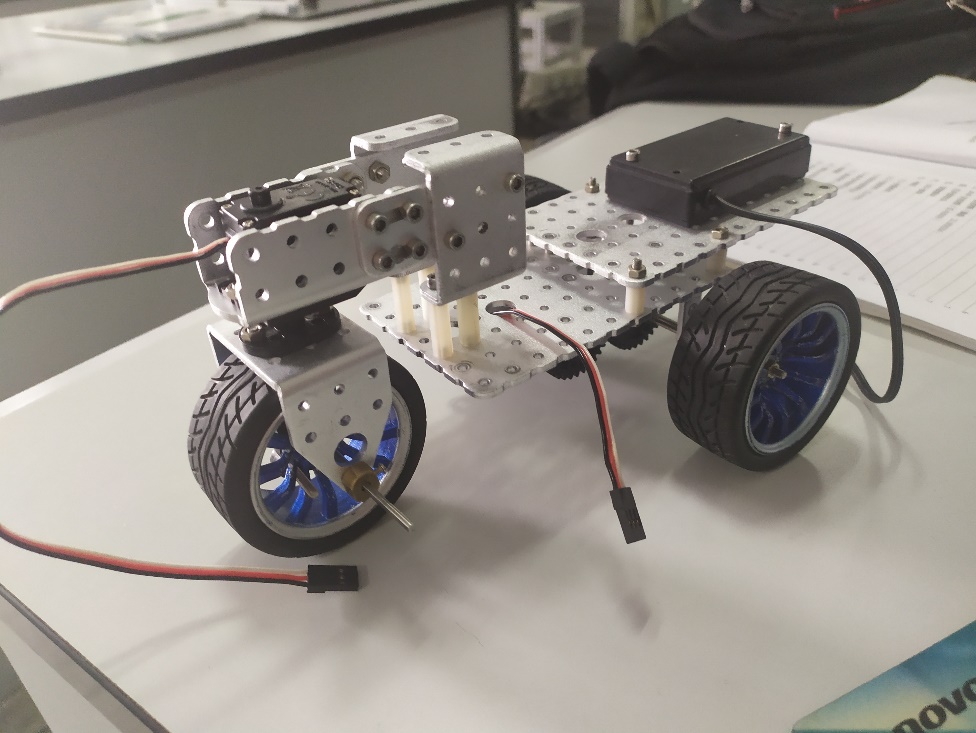


图2-1 初次搭建的版本

这是在方案确定后我们初次搭建的小车框架。但这种设计的缺点非常明显：在需要传递较大转矩的地方存在大量悬臂结构，这会导致应力集中，出现频繁的摇晃，以及各零件的不稳定。实际运行中，也出现了后轮脱落的重大失误。

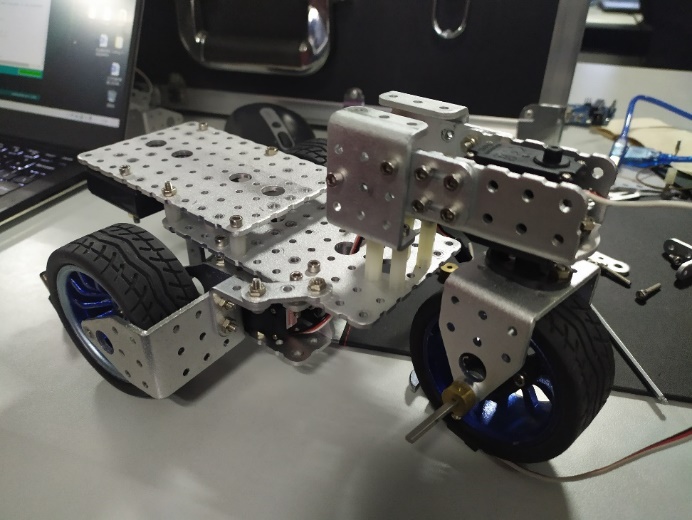


图2-2 一次改进后的版本

很明显地可以看出，我们把后轮的悬臂结构改为了简支梁结构，这样解决了应力集中的问题。但图中所示的固定件与连接件位置存在隐患，仅凭螺母摩擦力无法保证周向固定，后轮还是会出现径向的偏移。而且固定件此位置会导致舵机齿轮出现极大摩擦，严重影响小车的性能。

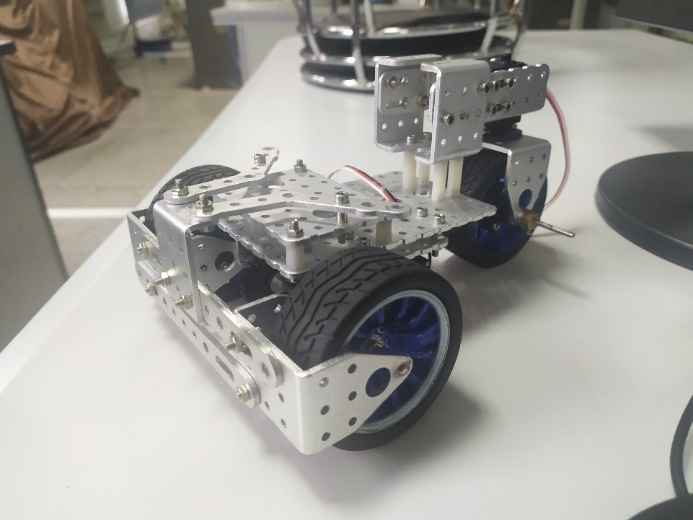


图2-3 二次改进后的版本

彻底解决了后轮固定的问题，将后轮固定件与底板完全固定在一起，同时底盘底部留有足够空间。这种连接方式的拆卸只需要拆除固定件与底盘的连接就可调整，符合设计要求。

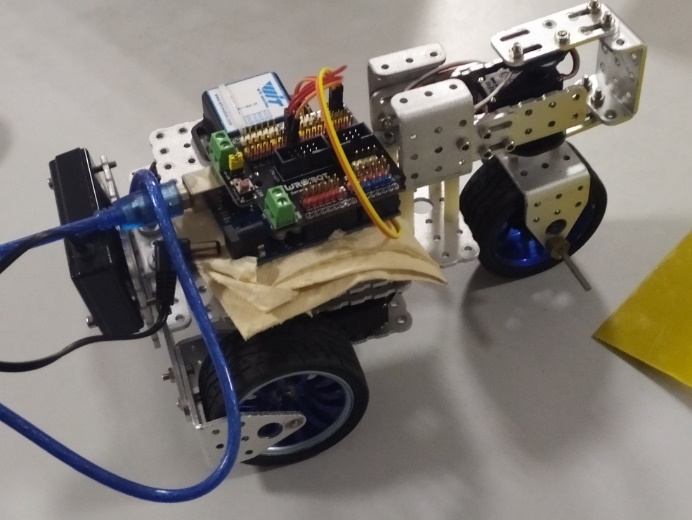


图2-4 三次改进后的版本

这次的改进解决了前轮存在的悬臂机构问题，同时在后轮固定件上加装竖直的固定板，用于安装电池，这样的设计稳定而且美观。在长期的调试中，我们一直采用这个结构，但发现重心偏后，在斜坡上导致了前轮抬起的问题。这严重影响了传感器的判定，因此我们对这个结构稍作修改。

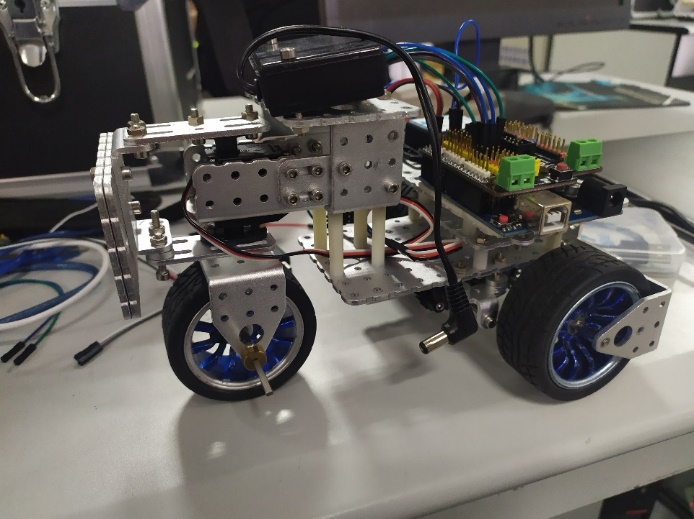


图2-5 最终版本

取消了后竖直板的设计，将电池固定于车重心上部。由于质量集中在驱动模块，所以不用担心重心过高问题。这次的设计满足了我们所有的要求，最后的调试和测评阶段均采用此结构。

2.2电路系统的制作与调试

电路系统中，有1个圆周舵机、1个标准舵机和1个JY901姿态角度传感器需要与Arduino UNO控制器相连，如图2-6所示。



图2-6 电路系统图

在搭建如图2-6所示的电路时，首先遇到的问题是引脚不够的问题。这是由于Arduino UNO开发板的5V引脚只有2个，而设计的电路中JY901传感器、圆周舵机、标准舵机各需要一个5V供电。为了解决这个问题，我们尝试了2中方法。

第一种方案，自己利用洞洞板焊接2组排针，每组排针分别串联，将Arduino UNO开发板的5V引脚通过杜邦线引出至其中一组排针，GND引脚引出至另一组排针。传感器、圆周舵机、标准舵机的对应引脚分别接到两组排针上，达到拓展引脚的目的。这种方法成本低廉、操作较容易，但在使用时由于所使用的洞洞板占空间较大，很难稳定地固定在小车上，同时降低了整个机械电路系统的整体性，结构上也不稳定。在实际测试时，遇到了杜邦线因小车震动而脱落、杜邦线排布混乱的问题。

为了解决第一种方案带来的问题，我们提出了第二种方案，即采用YwRobot生产的IO SHIELD V1拓展版拓展引脚，如图2-7所示。通过将这款拓展版连接至Arduino UNO开发板拓展5V和GND引脚，虽然提高了电路系统的成本，但操作大大简化，系统的整体性、稳定性得到了提高。

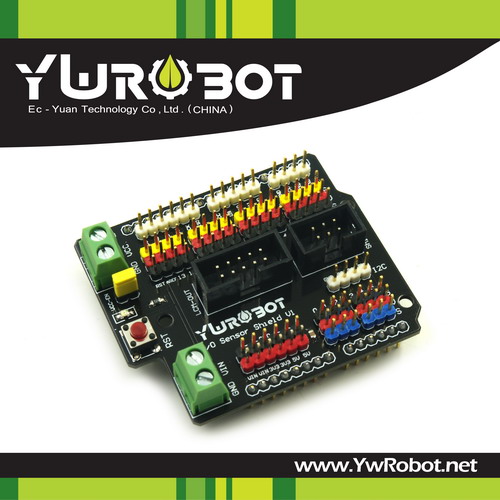


图2-7 YwRobot IO SHIELD V1拓展版

为了检验所制作的电路符合要求，检验分为以下几步。

第一，分别确定各元器件正常工作。例如，一开始所用的Arduino UNO开发板无法连接至电脑下载程序，在尝试了多台电脑后无果后，决定更换开发板；通过官方给的上位机程序，调整JY901传感器的参数并校零，发现无法校零磁场，初始化传感器后解决了问题。

第二，单独连接每一个模块至开发板，下载测试程序，检验是否正常工作。例如，我们所使用的前两个圆周舵机经检验都已经损坏，直到第三个才正常工作。

第三，分别测试每个模块的实际性能参数。例如，对于JY901传感器，使用实例程序获取读取到的参数，确定其变化范围、变化速度、精度等，看是否满足设计要求；对于圆周舵机，测试其转速的上下限、PWM波控制舵机速度所能达到的精度、舵机响应的速度等，看是否满足设计要求。

第四，整合所有模块，并通过编写综合测试程序，综合评价电路系统各模块是否协调、能否达到设计要求。

2.3控制程序的编写与调试

控制程序分为两个模块，传感器信号处理模块与舵机驱动模块。信号处理模块使用驱动库文件，通过设定Arduino与JY901的通信波特率实现读取X、Y、Z轴加速度、角度、角速度。之后通过互补滤波对角度数据进行处理。

驱动模块的核心为PI控制器，将角速度值作为偏差，控制量 。其中， 为角度值互补滤波处理后的结果。根据与测量结果，当 时，舵机正转，小车前进；反之舵机反转，小车后退。

在信号处理模块中，需要调用函数测得单词程序的运行周期，作为互补滤波公式中的 ,并 设置为0.98。互补滤波公式见式（1-1），这里重新列出如下：

在该模块中，需要注意JY901的波特率设置，并注意根据传感器的摆放位置合理选用X、Y、Z轴的数据。通过Arduino接受传感器数据并发送到上位机，可以检测信号读取是否正确。

在驱动模块中，需要注意小车在向前和向后行驶时，驱动能力不同，需要设计两套不同的和参数。同时，在小车于跷跷板直行并即将进入调平衡时，应当使速度尽量慢，减小动静摩擦不同以及惯性带来的影响。基于对任务时间限制的考量，我们使速度随时间降低，降低的幅度需要预先测量。

三、控制系统的测试与分析

3.1测试方法

评价小车性能有以下几个要素：行驶角度调整时间（调整角度为45°）；调平衡的次数（前后移动一次为调平衡一次）；调平衡的时间；调平衡总过程跷跷板平均倾角。其中，行驶角度调整时间代表180°舵机的灵敏性；调平衡的次数反映系统的震荡特性；调平衡的时间反映系统的综合性能；平均倾角反映系统的超调量。

同时，为测试滤波效果，测试工作将在滤波和不滤波两种条件下进行。

为测量以上要素，我们使用秒表来计算时间，并通过量角器测量倾角，人工记录角度大小和调平衡的次数。

3.2测试数据与现象

在测试时，我们发现小车在调整过程中会使跷跷板触地，从而严重影响系统的平衡。成功的测试均呈现震荡幅度由大到小变化并最终不变的现象。经过性能比较，我们选择测试中表现最理想的一组参数作为使用参数。参数性能测试表如下。

使用互补滤波的数据见表3-1至表3-4；

不使用滤波的数据见表3-5至表3-8。

表3-1 使用互补滤波的行驶角度调整时间

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 行驶角度调整时间 | 2.7s | 2.2s | 3s | 2.3s | 2.2s | 2.48s |

表3-2 使用互补滤波的调平衡次数

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 调平衡次数 | 7 | 5 | 2 | 8 | 6 | 5.6 |

表3-3 使用互补滤波的调平衡时间

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 调平衡时间 | 20s | 15s | 31s | 17s | 22s | 21s |

表3-4 使用互补滤波的平均角度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 跷跷板倾角 | 4.2° | 3.3° | 3.7° | 2.8° | 2.6° | 3.32° |

表3-5 不使用滤波的行驶角度调整时间

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 行驶角度调整时间 | 3.2s | 3.3s | 4s | 3.6s | 3.7s | 3.56 |

表3-6 不使用滤波的调平衡次数

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 调平衡次数 | 10 | 7 | 9 | 13 | 11 | 10 |

表3-7 不使用滤波的调平衡时间

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 调平衡时间 | 22s | 30s | 27s | 39s | 34s | 30.4s |

表3-8 不使用滤波的平均角度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试次数 | 1 | 2 | 3 | 4 | 5 | 平均值 |
| 跷跷板倾角 | 3.9° | 4.7° | 4.1° | 3.4° | 3.7° | 3.96° |

在多次调平衡测试中，测试量围绕在某个值在某个范围内上下波动。并且，使用滤波的效果明显强于不使用滤波。

3.3结果分析

使用滤波后，行驶角度调整平均时间T=2.48s，调平衡平均次数N=5.6，调平衡平均时间t=21s，调平衡总过程跷跷板平均倾角=3.32°。

相对于其他数据，调平衡总过程的平均倾角较大，说明系统存在较大超调，即驱动模块中，KI参数较大。需要适当减小来抑制系统超调。

将使用滤波前与使用滤波后对比，行驶角度调整平均时间T=3.56s，调平衡平均次数N=10，调平衡平均时间t=30.4s，调平衡总过程跷跷板平均倾角=3.96°。

可以看出，使用滤波将提高系统整体性能，包括响应速度、超调、震荡特性等。

四、总结与心得

文博正：

这次的课程是我比较系统的接触机器人制作一线操作的体验，之前并没有参加什么竞赛，偶尔参加了一个，还设计出了轻易就会被烧毁的电路。这次负责机械结构的设计，以为自己对这方面的知识掌握的娴熟，实际上各种有毛病的结构让我认识到了我的不足之处。而且硬件组和软件组要通力配合，才能设计出结构稳定，功能可靠的装置。

机械设计不是零件的堆积，而是深思熟虑后模块的组合，这既是为了功能的直观化，也是为了加工和修改的方便。作为一名机械设计者，更要有整体和部分相结合的思想，还要将所掌握的理论知识运用到生产实际当中，合理建模，不断修改完善自己的机械装置。在材料受限的情况下，如何最大化使用手头的资源进行设计，以结构来弥补材料的不足，尽量止损，也是重要的素质。课程中困难重重，但不失败个十几次不好意思说自己做过机械设计，我在此次经历中意识到了灵活运用知识和积累经验的重要性。希望自己接下来能积累更多经验，成为制造业合格的人才。

欧亚非：

这次实践课，我最大的收获是对硬件的设计制作有了一定的认识。以前我所接触的软件、算法设计，基本上都与硬件相分离。而这次跷跷板平衡小车的制作，让我认识到软件与硬件相结合的难度之大。

第一，机械、电路、程序各部分的设计不能简单堆积，想一出是一出，必须先系统地设计其结构思路，在着手搭建。我们一开始不做设计、直接搭建，浪费了很多时间。

第二，各部分不能完全脱离，分工协调很重要。组内合理分工能大大节约时间，提高效率。虽然有组内分工，但大家必须都清楚其他部分是怎么实现的。一开始，我们搭建机械部分没有考虑到算法实现的难易程度，设计了使用差速控制转向的结构，是不合理的。

第三，需要将理论知识应用到实践中。在老师的引导下，我们将《控制工程基础》、《信号与系统》中的理论知识应用到了小车跷跷板系统里，对这个系统的各个部分进行了分析，对小车的设计搭建帮助很大。

第四，需要有耐心，做好不断失败、不断修改的准备。随着后期调试期间不断发现问题，硬件的设计也需要不断调整，设计不可能一遍就达到最好版本，只能不断改进。

在这次课程中，我们遇到了很多困难，最终的运行结果也不是很理想。这更让我认识到做硬件的困难，认识到我的学习还存在很多不足。今后还需多加实践，才能在毕业的时候具备一些真才实学。

曹啸：

本次集中实践教学课，给了我很大的启示。马克思主义告诉我们，认识真理的过程是实践——认识——实践。而之前的教学都属于直接的认识教育，导致很多人对事情的了解只是浮于书本，缺少具体的认识。众所周知，马克思主义的目的并非是解释世界、认识世界的运作原理，而是通过实践把握真理，从而改造世界。所以，一切学习的目的都是为了实践，通过实践检验真理并改造世界。所以，从这个角度来看，这类实践课应当提早开设。

在本次课程中，我认识到自己对工程控制的理解尚浅，仅仅是能够做题是远远不够的，现代控制理论还有很多要学习的地方，我也会在研究生阶段继续在控制的学术道路上前进。