《移动机器人实训》报告要求

1. 移动机器人概述

移动机器人是一种能够自主或半自主地在物理空间内移动和执行任务的智能机器设备。它们结合了机械、电子和计算机科学的多个领域，在生活中变得越来越常见，例如扫地机器人、无人驾驶汽车等。移动机器人可以分为轮式、履带式、躯干式、腿式等。轮式移动机器人应用广泛，有两轮、三轮、四轮的机器人。我们所学的移动机器人是四轮式的，由执行器、控制器、传感器和机械机构组成，可以通过编程和反馈控制完成转动、直走、循迹等多种任务。

* 1. 移动机器人的技术与应用场景

移动机器人技术的发展涉及多个关键领域，其应用场景也非常广泛，涵盖了工业、服务、军事、医疗、农业和个人消费等多个领域。下面详细介绍移动机器人的技术和主要应用场景：

技术：

（1）导航与定位技术：

（2）SLAM（Simultaneous Localization and Mapping）：同时定位与地图构建技术，允许机器人在未知环境中自主导航和建立地图。

（3）传感器融合：将多种传感器数据（如视觉、激光雷达、惯性传感器等）融合，提高机器人在各种环境中的感知能力和精确度。

（4）路径规划与决策算法。

（5）使用经典的路径搜索算法（如A\*算法）、动态路径规划和基于机器学习的决策算法，使机器人能够在复杂环境中高效、安全地移动和执行任务。

主要应用场景：

（1）工业自动化。

（2）在工厂和生产线上用于物料运输、装配、包装和检测。移动机器人可以代替人工在危险或重复性高的环境中操作。

（3）服务机器人：

（4）清洁机器人：如自动扫地机器人、自动洗地机器人，用于商业和家庭清洁。

* 1. 基于Arduino的移动机器人设计方案

基于Arduino的移动机器人设计方案通常包括硬件和软件两个主要部分。Arduino是一个开源的电子原型平台，具有丰富的传感器接口和易于编程的特点，非常适合用于学习和制作简单到中等复杂程度的移动机器人。

硬件设计：

（1）模块：使用常见基于H桥的驱动模块，用于控制电机的速度和方向。

（2）编码器：用于测量电机的旋转和速度，实现精确的运动控制。

（3）电源：使用适当的电池供电，考虑电机和控制电路的功耗，确保电池容量足够支持机器人的运行时间。

（4）Arduino主控板：使用常用的Arduino UNO，具有丰富的IO接口和足够的计算能力来控制和管理各种传感器和执行器。

软件设计：

（1）Arduino IDE编程：使用Arduino IDE编写机器人的控制程序。编程语言主要是基于C的Arduino语言。实现基本的运动控制，如前进、后退、左转、右转等。

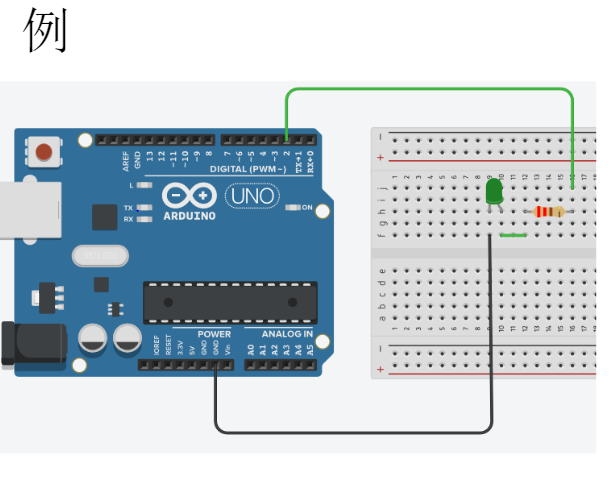
（2）运动控制算法：实现简单的运动控制算法，如闭环控制，或者开环控制下的基本路径规划和避障算法。

2、基于Arduino的移动机器人的控制与实现

2.1 Arduino Uno数字IO口的应用

任务（作业3）：IO控制LED

1. 电路方案



1. 编程实现



简述：若发送字符1则控制板点亮LED灯1秒后熄灭，若发送字符2则控制板电量字符两秒后熄灭，若发送字符3则常量LED，若发送字符4则熄灭LED灯。

3）测量数据

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测量量 | 电阻电压（V） | 电阻（Ω） | LED电压（V） | 电流（mA） | 功耗（mW） |
| 测量值 | 1.16 | 300 | 2 | 3.87 | 7.74 |

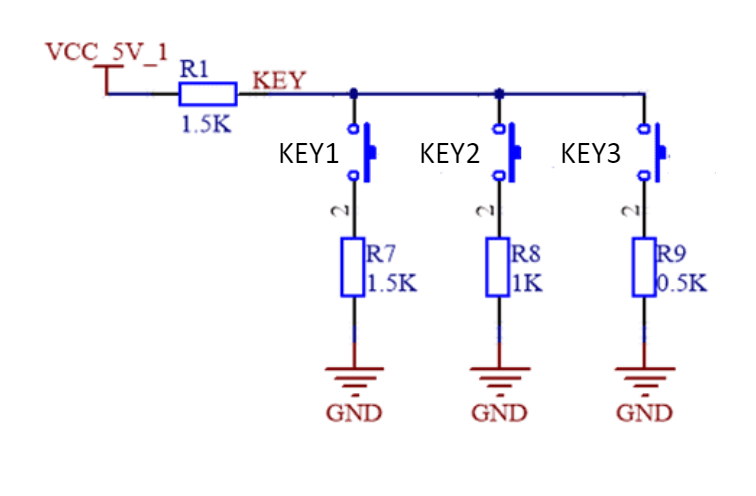
4）结论

经过测量，USB供电不足5V。LED电压为2V，电流为3.87mA，功耗为7.74mW。

2.2 Arduino Uno模拟IO口的应用

任务（作业4）：采集模拟键盘的值，通过串口打印测试。

1. 电路方案



2）编程实现



简述：若AD检测大于400则输出“NO KEY”表示都未闭合。

若AD检测大于300小于400则输出“KEY1”表示KEY1闭合。

若AD检测大于200小于300则输出“KEY2”表示KEY2闭合。

若AD检测大于100小于200则输出“KEY3”表示KEY3闭合。

1. 测量数据

|  |  |  |  |
| --- | --- | --- | --- |
| 情况 | KEY点电压/V 万用表测量 | AD检测  (0~1023) | 计算值/V |
| 都未闭合 | 3.16 | 673 | 3.3 |
| 闭合KEY1 | 1.58 | 339 | 1.65 |
| 闭合KEY2 | 1.26 | 271 | 1.32 |
| 闭合KEY3 | 0.8 | 173 | 0.84 |

结论：KEY1、KEY2、KEY3都未闭合时，AD大于400，KEY点电压为3.16,；

KEY1闭合时，AD小于400大于300，KEY电压为1.58；

KEY2闭合时，AD小于300大于200，KEY电压为1.26；

KEY3闭合时，AD小于200大于100，KEY电压为0.8。

任务（作业5）：PWM控制LED亮度。

测量数据

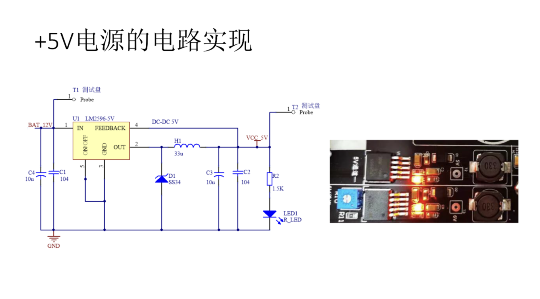
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 给定值 | 二极管两端的电压/V | UR/V | I/mA | 亮度比较 | 理论占空比/% | 测量占空比/% | PWM频率/Hz |
| 100 | 0.81 | 0.98 | 3.26 | 暗 | 39.216 | 39.216 | 490 |
| 150 | 1.22 | 1.47 | 4.90 | 较暗 | 58.824 | 58.826 | 490 |
| 200 | 1.62 | 1.97 | 6.57 | 较亮 | 78.431 | 78.436 | 490 |
| 250 | 2.02 | 2.44 | 8.13 | 亮 | 98.039 | 98.039 | 490 |

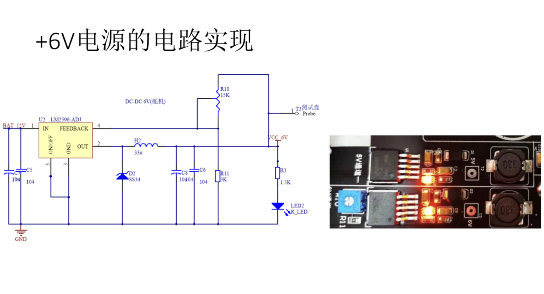
4）结论

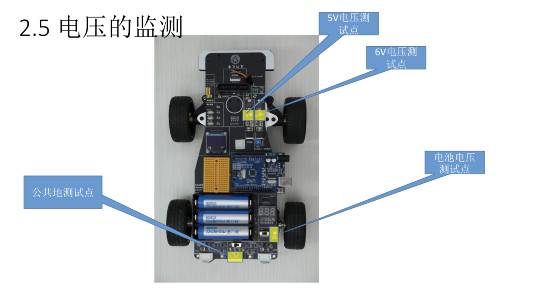
随着PWM增大，二极管两端电压增大，电流增大，亮度增加，占空比增大。PWM频率基本没有改变。

2.3 电源系统

1）系统方案







2）系统参数

用三块3.7V的锂电池，尺寸为18650,81代表直径为18mm，65表示长度为65mm，0表示圆柱形电池。

3）测量数据

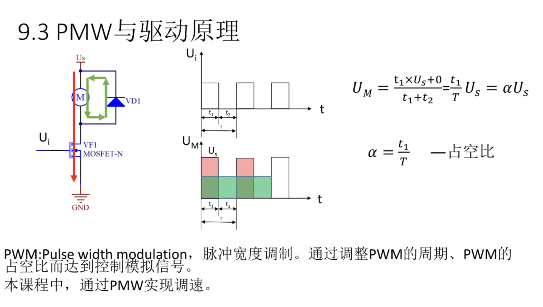
|  |  |
| --- | --- |
| 电源电压/V | 11.04 |
| +5V端电压/V | 5.03 |
| +6V端电压/V | 5.98 |

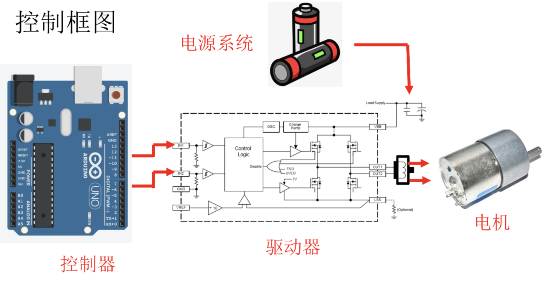
1. 结论

电源电压为三块锂电池电压加和，+5V端电压和+6V端电压若不与之相等则需要用螺丝刀调整旋钮，以此调整端电压。

2.4 电机驱动

1）方案设计





用万用表测量电机电压。

2）编程实现



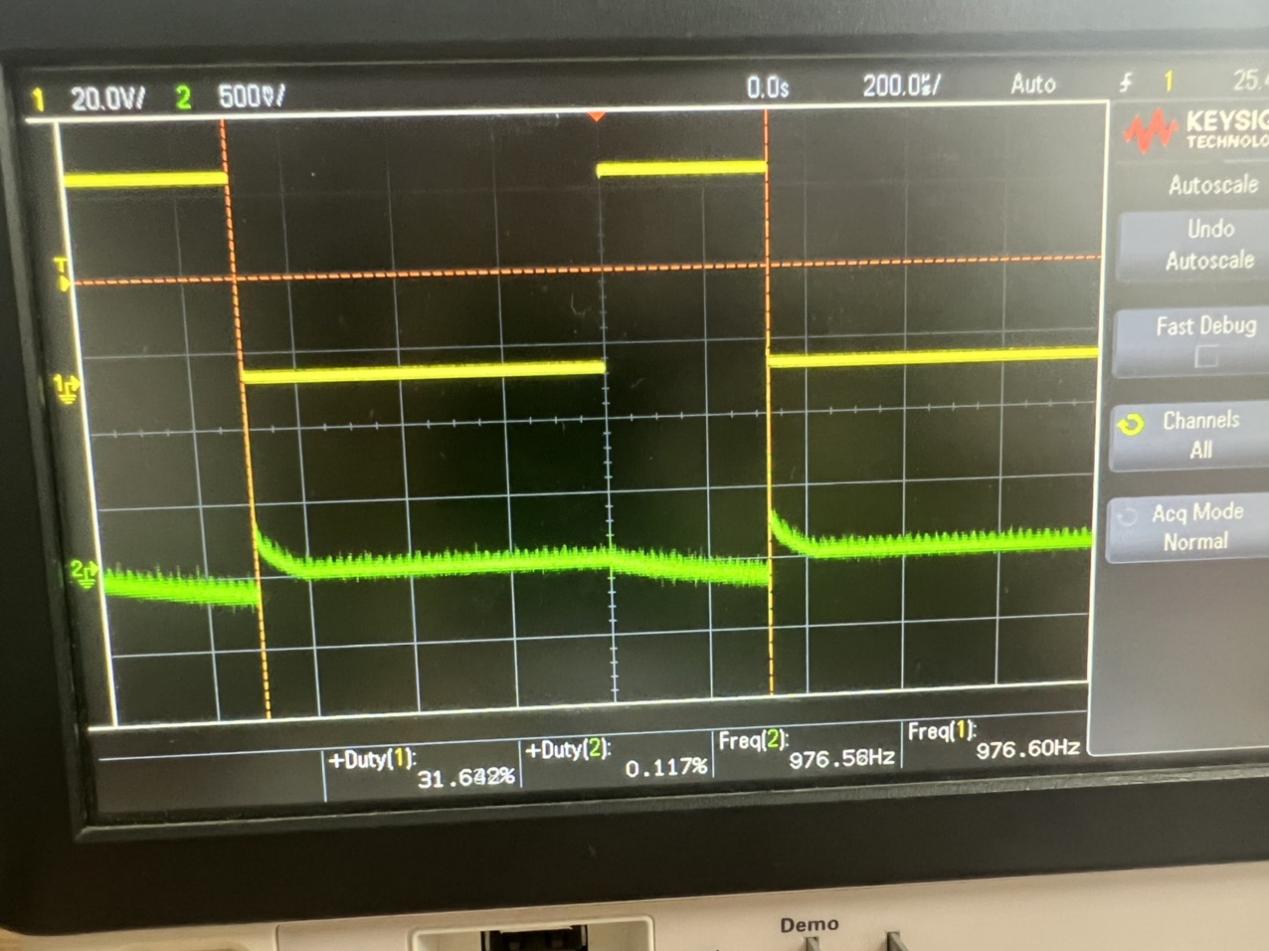
简述：调整PWM，观察占空比变化、频率变化，判断电机是正转或反转。

3）测量数据（含波形）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 给定值 | 理论值 | 测量值 | | | |
| 占空比% | 调速信号的频率/HZ | 占空比% | 正转或反转 | 电机电压/V |
| 50 | 19.6 | 976.6 | 19.922 | 正转 | -1.36 |
| 80 | 31.4 | 976.6 | 31.64 | 正转 | -4.11 |
| 100 | 39.2 | 976.6 | 39.45 | 正转 | -5.42 |
| -50 | 19.6 | 490.2 | 19.6 | 反转 | 2.18 |
| -80 | 31.4 | 490.2 | 31.37 | 反转 | 5.31 |
| -100 | 39.2 | 490.2 | 39.22 | 反转 | 6.86 |

波形：

50：

80：

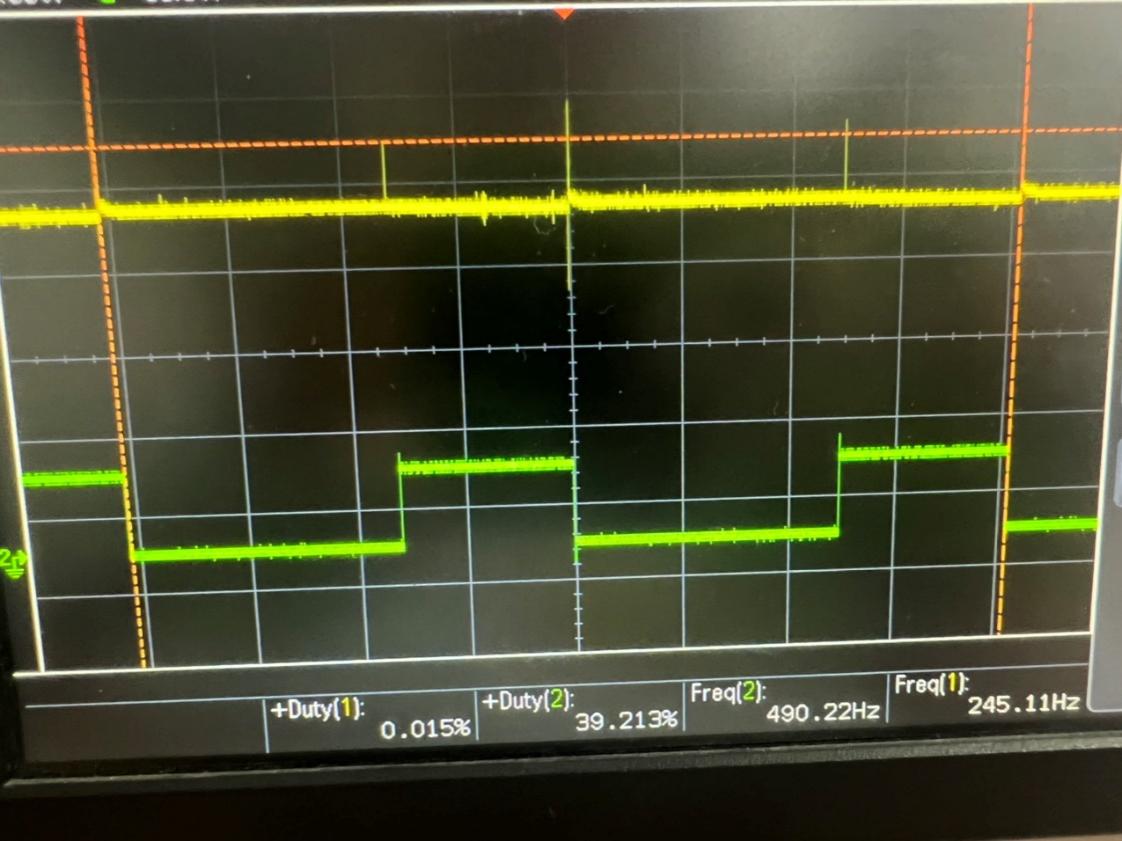
100：

-50：

-80：



-100：



1. 结论

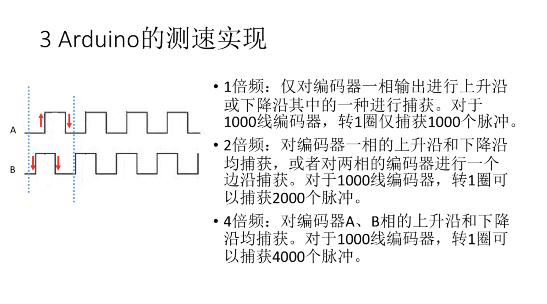
随PWM增大，占空比增大，电机电压绝对值增大。

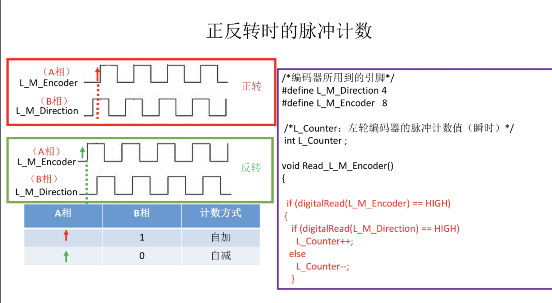
正转时电机电压为负数，频率基本不变，为976.6Hz。

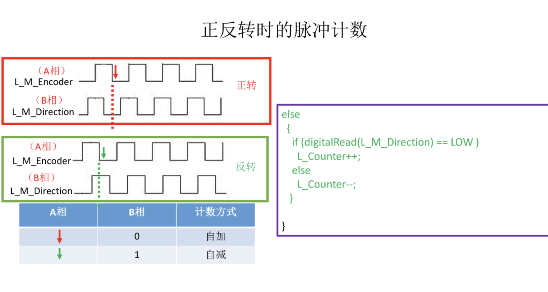
反转时电机电压为正数，频率基本不变，为490.2Hz。

2.5 测速实现

1）方案设计



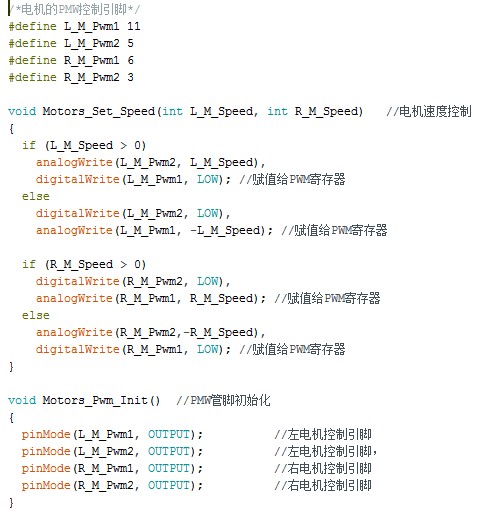


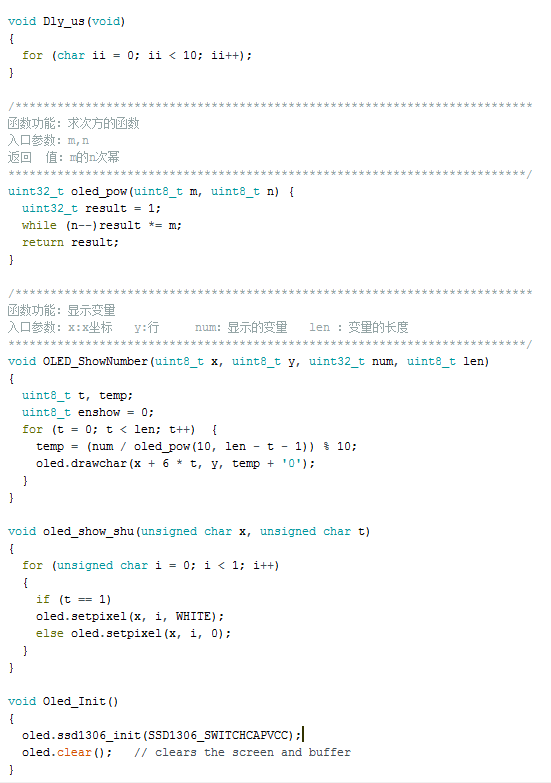


2）编程实现







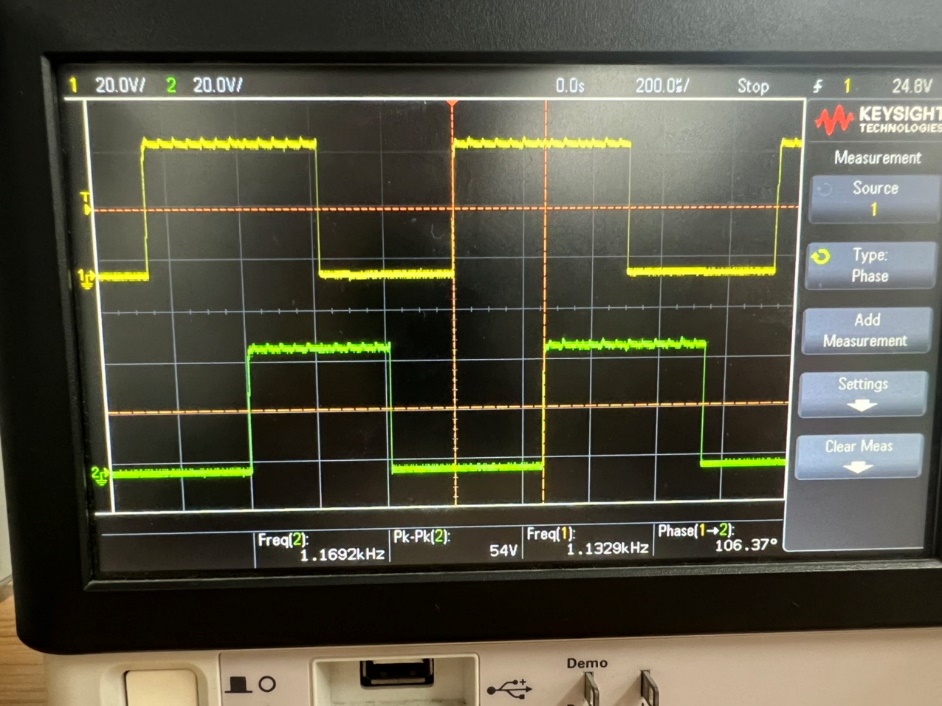


简述：若A为上升沿且B为高电平时为正转，若A为上升沿B为低电平时为反转。若A为下降沿且B为高电平时为反转，若A为下降沿B为低电平时为正转。

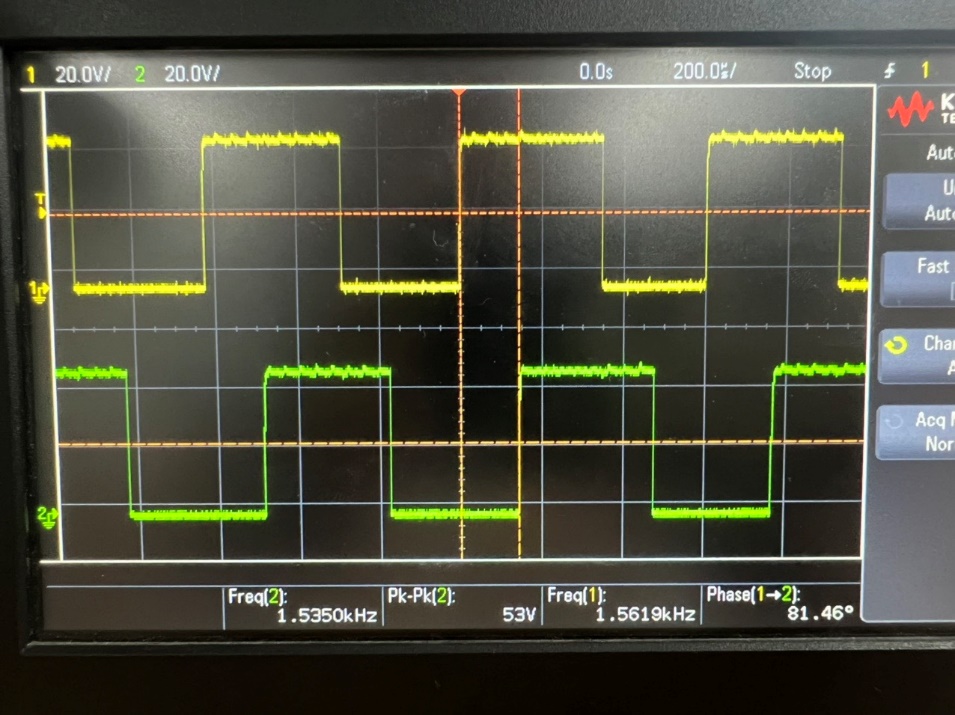
3）测量数据（含波形）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Arduino测速(左轮） | | | | | | |
| 序号 | 给定值 | 示波器测量 | | | 0LED | 计算值 |
| A相频率/HZ | B相频率/HZ | AB相位差/° | 5ms计数 | 计算频率/HZ |
| 1 | 100 | 1169 | 1132 | 106.37 | 12 | 1200 |
| 2 | 150 | 1535 | 1562 | 81.469 | 16 | 1600 |

100:



150:



1. 结论

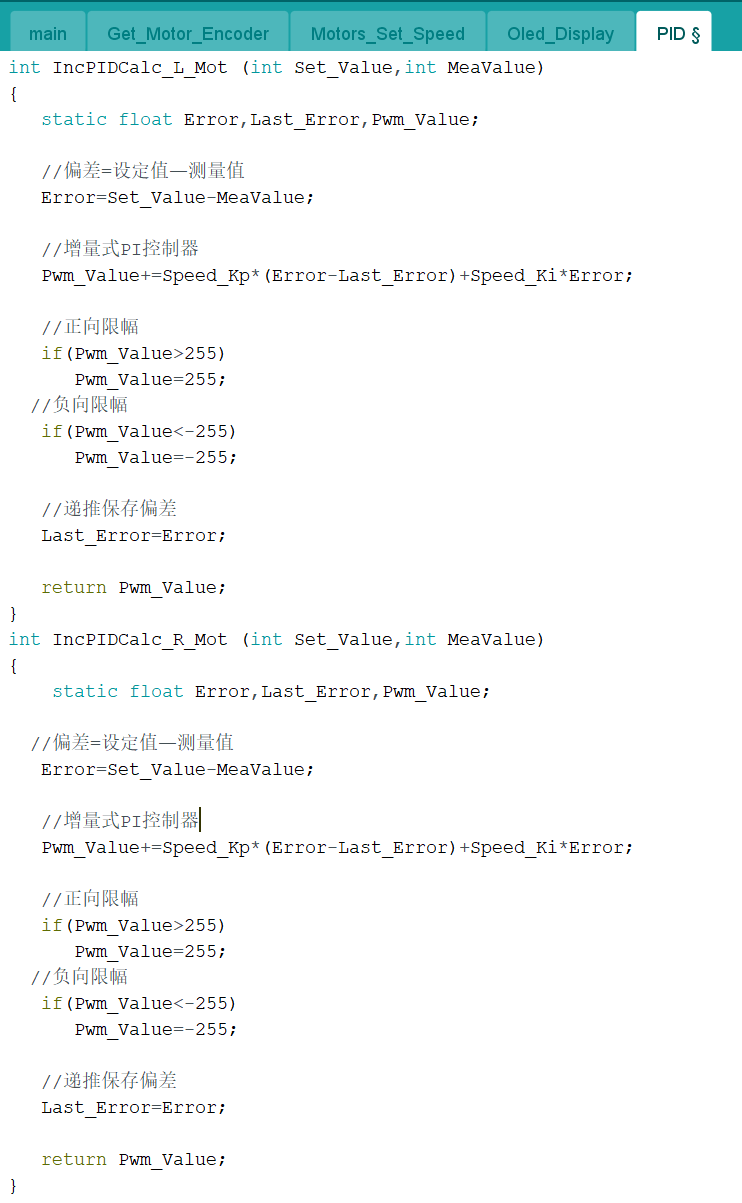
随给定值增大，A、B频率增大，相位差减小，5ms计数增大。

2.6 闭环控制实现

1）方案设计

先根据经验初设置比例系数和积分系数，再通过观察输出波形修改系数值，如可以通过减小比例系数增大积分系数来减小超调量。通过修改比例系数和积分系数以此来减小超调量、减小上升时间和调节时间。

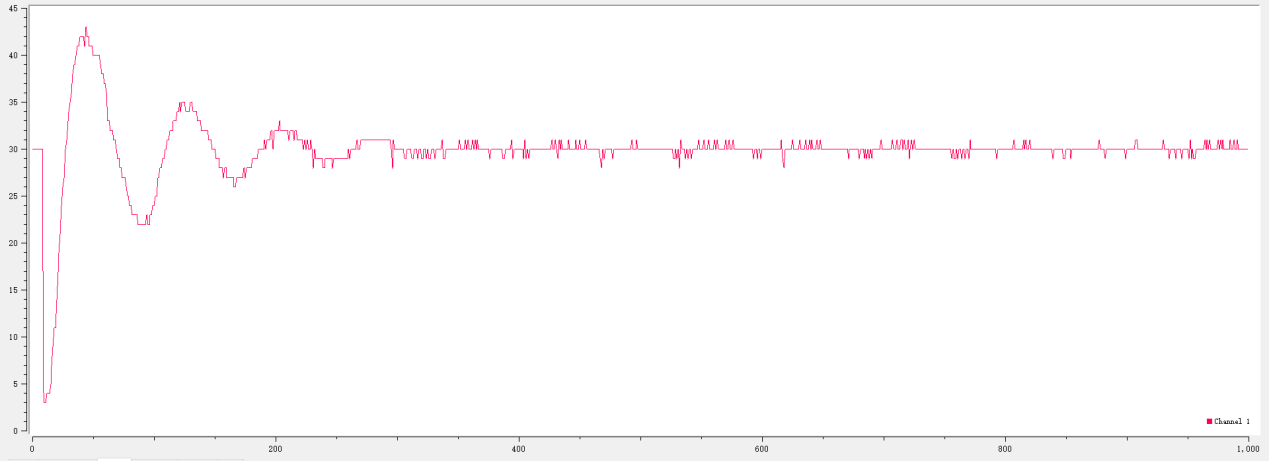
2）编程实现



简述：通过修改Ki和Kp可以调整波形输出。

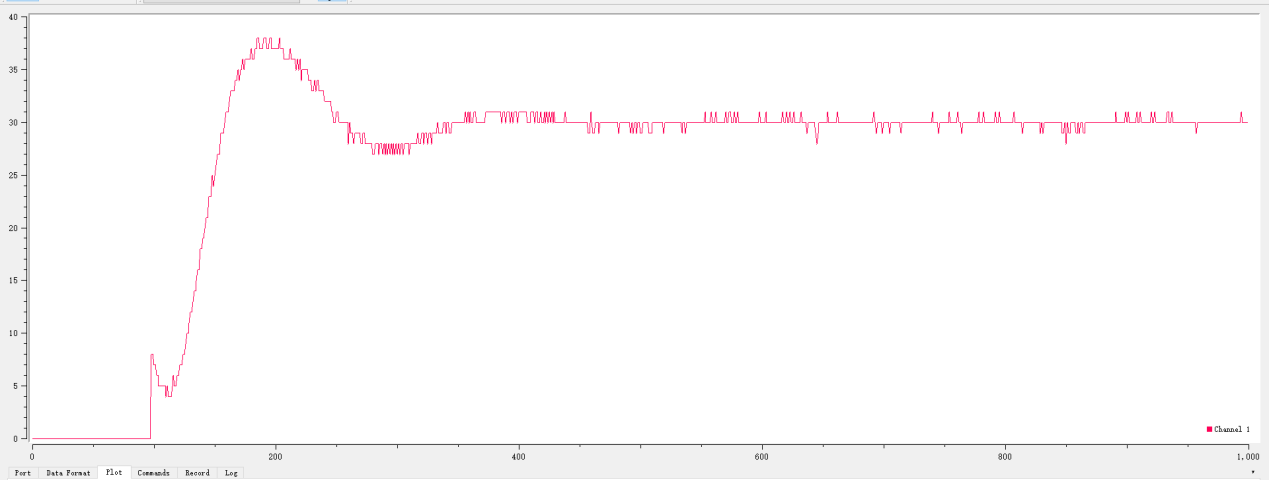
3）测量数据（含波形）

Ki=0.5, Kp=0.5,



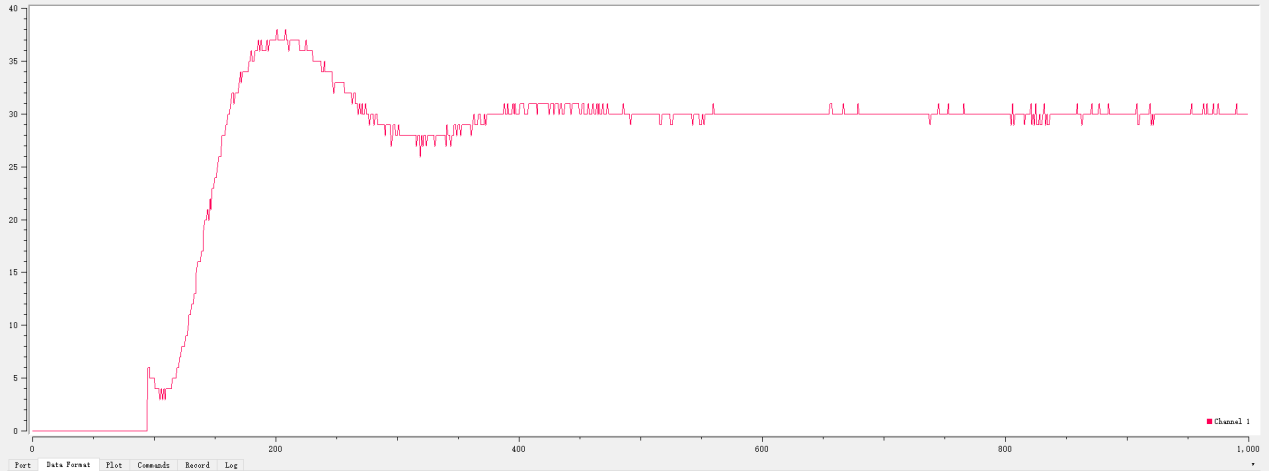
由上图可知：该波形超调量很大，调节时间长，但上升时间较短，为了使调节时间变短，我们考虑减小比例环节系数ki。

Ki=0.1,Kp=0.5



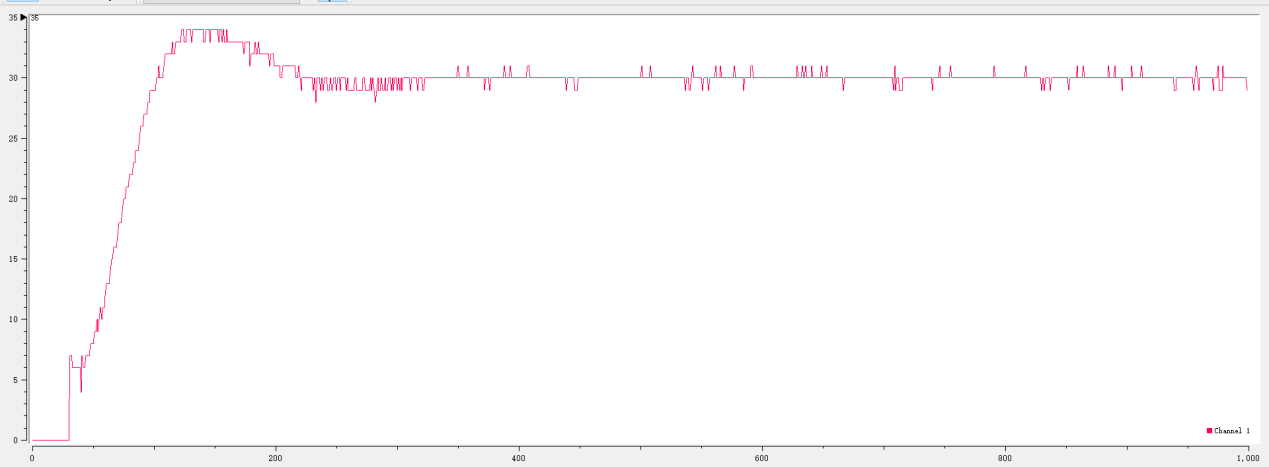
由上图可知：该波形超调量较大，调节时间变短，上升速度很快，为了使超调量减小，我们考虑将比例环节系数Ki减小。

Ki=0.08, Kp=0.5



由上图可知：该波形超调量仍然较大，但调节时间以及上升时间较合适，因此我们考虑增加积分环节系数Kp。

Ki=0.08, Kp=2.5



由上图可知：该波形超调量较小，调节时间较小，上升速度很快，并且在我们给出外界扰动后仍能快速恢复平衡状态，我们认为该控制系统良好。

4）结论

若超调量大则减小比例环节增大积分环节；若上升时间长则增大比例系数；

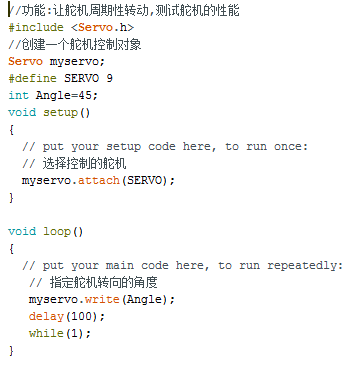
若调节时间长则可以减小比例环节增大积分环节。

2.7 舵机控制实现

1）方案设计

通过调整Angle，将示波器连在舵机线上，观察周期周期变化和脉宽变化。

2）编程实现

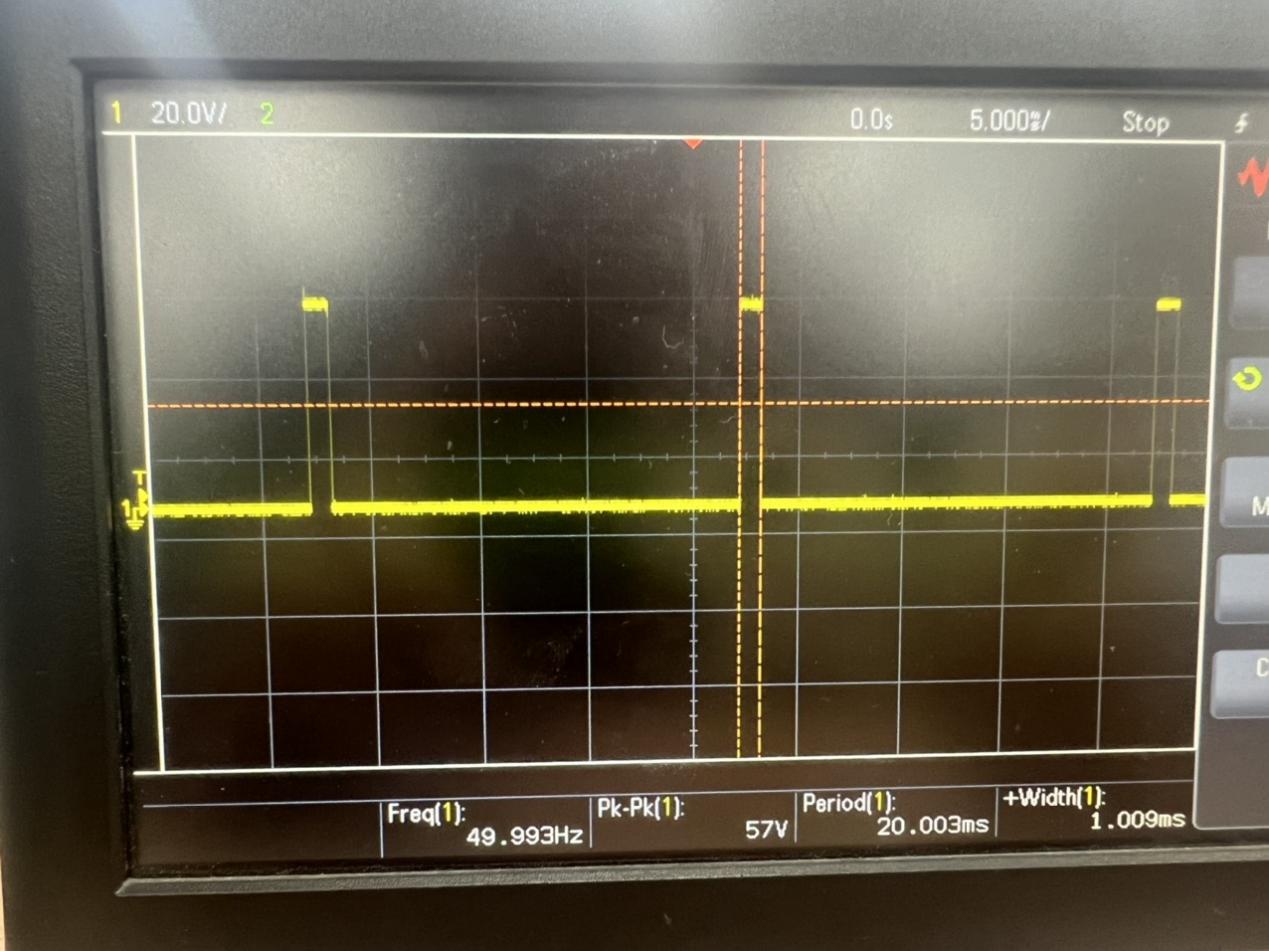


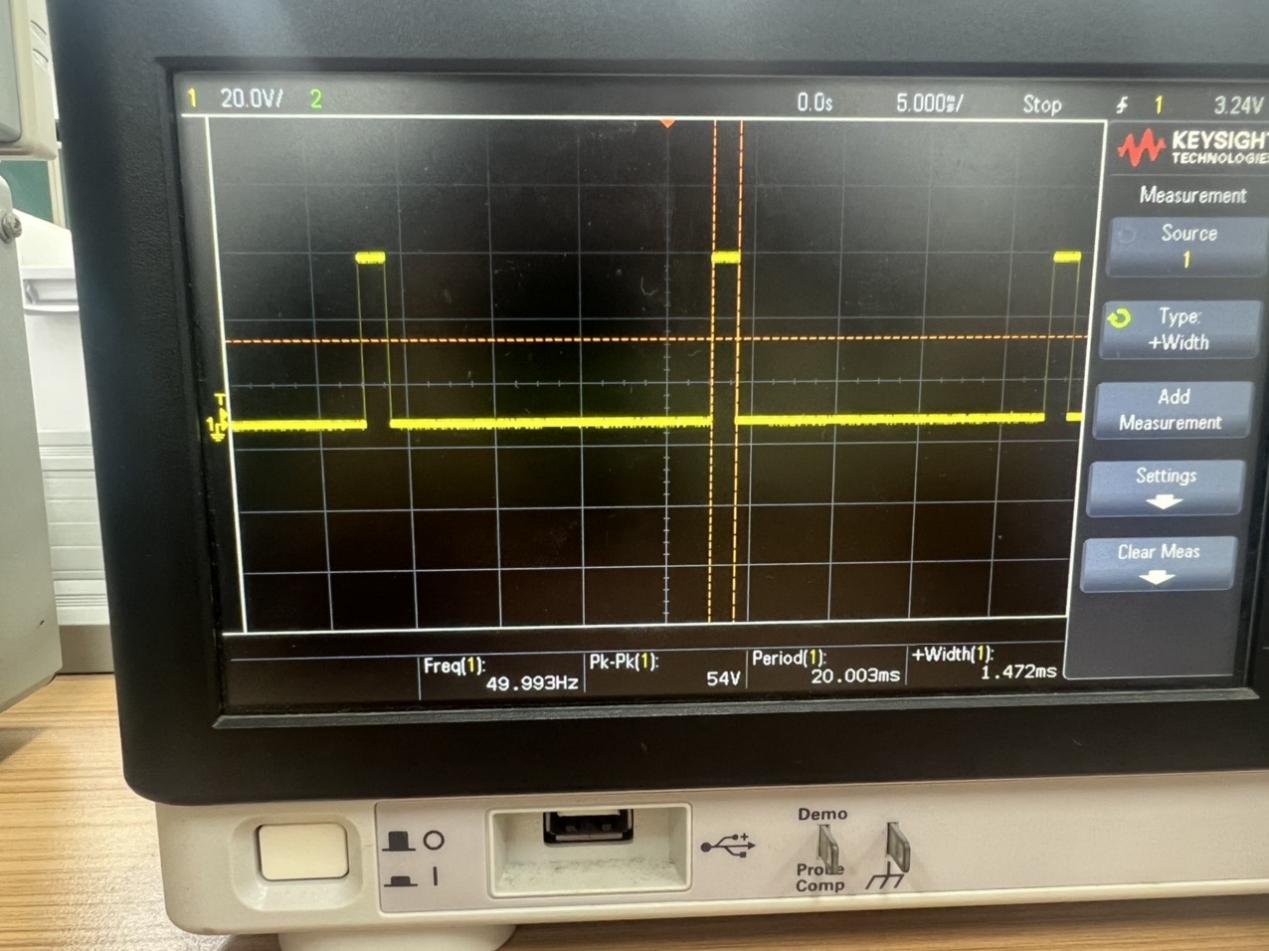
不断改变Angle来观察周期和脉宽变化。

3）测量数据（含波形）

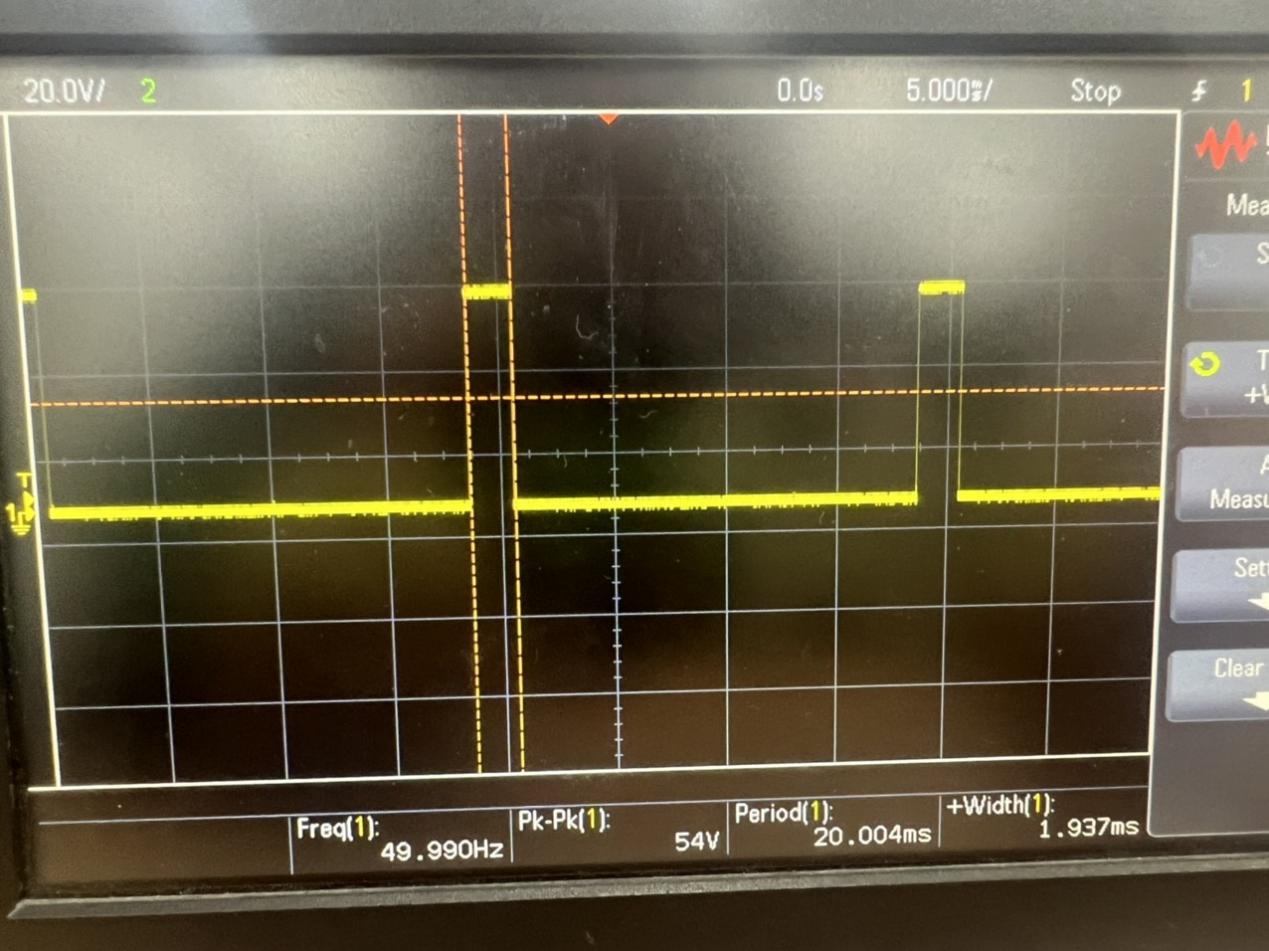
|  |  |  |  |
| --- | --- | --- | --- |
| 舵机控制特性测试 | | | |
| 序号 | 给定值 | 周期/ms | 脉宽/ms |
| 1 | 90 | 20.003 | 1.472 |
| 2 | 135 | 20.004 | 1.937 |
| 3 | 45 | 20.003 | 1.009 |

Angle=45



Angle=90：

Angle=135

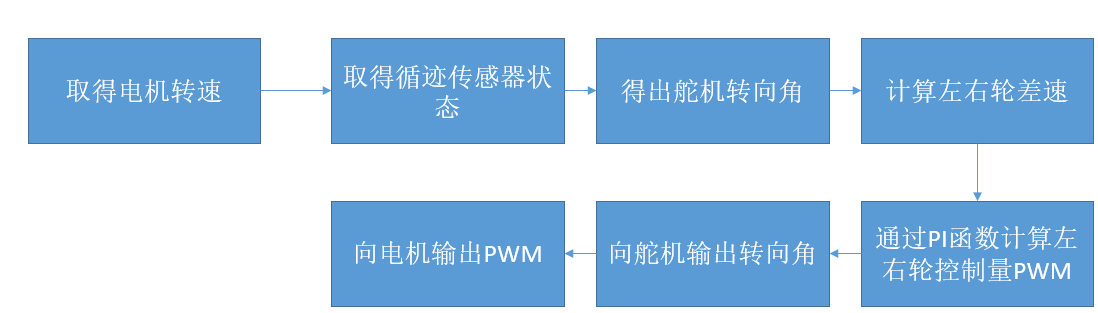


4）结论

随着Angle的增大，周期基本不变，正脉宽逐渐增大。

3、移动机器人的循迹实现

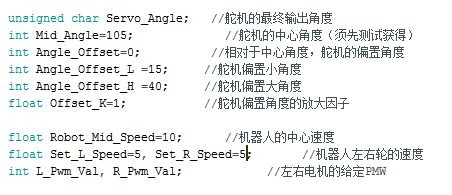
1）方案设计

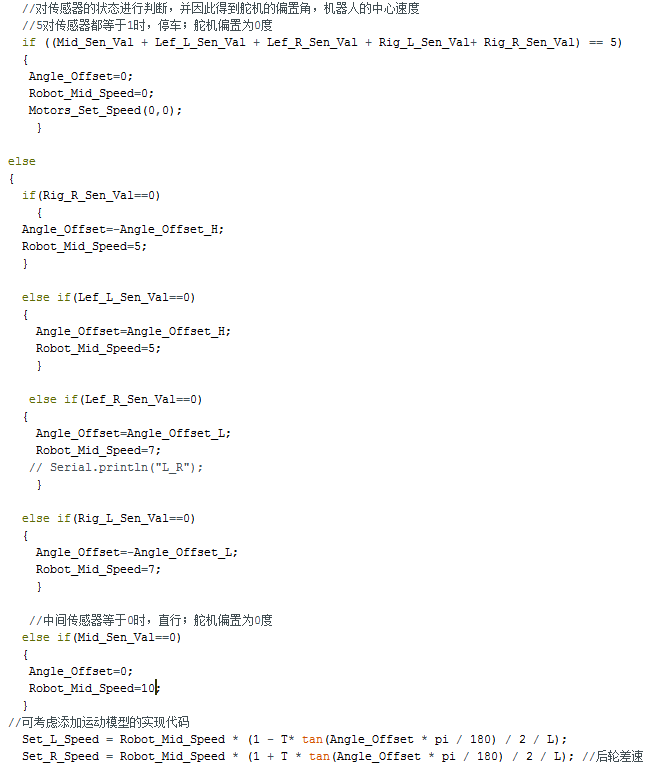


循迹逻辑:

当黑线位于中间传感器下方时，前轮打正车辆直行；当黑线位于左侧传感器下方时，前轮左转，车辆左转弯；当黑线位于右侧传感器下方时，前轮右转车辆右转弯。若外侧传感器感应黑线则转大弯，若内侧传感器感应黑线则转小弯。检测到全为白时停止前进。

2）编程实现





3）问题与解决方法

开始我们所设速度过快，我们发现小车在前进时很容易因为速度过快转弯不够及时无法正确转弯，于是我们调小速度，情况有所好转。

我们开始选择转小角15度，大角30度，我们发现在转读书较大的弯时小车容易卡在转弯处，转弯不够及时，于是我们先选择增大小角转弯度数，但发现小车在走直线时轮子转弯很大，无法平稳前进，于是我们增大大角转弯度数，以此保证小车可以正常转弯。

1. 总结与建议

在调整小车参数时，要通过不断试验，观察小车在跑道上的表现，再结合跑道的情况，修改小车参数。速度不应过大，应先保证低速循迹再追求高速循迹。转弯角不应设置太大，否则容易在直线处方向错乱，小车方向出现问题；转弯角度也不能设置太小，否则无法保证小车在度数较大的弯处正常通过。总的来说，要通过不断的试验、修改、分析、再试验，以此保证小车在赛场上达到最佳状态。