

摘要

独轮机器人具有非线性、强耦合的复杂动力学系统。是控制理论和动力学研究的一大挑战，是一个非常好的研究模型。本文将设计一款独轮机器人，侧重分析独轮机器人的运动，并建立动力学模型以及编写控制程序。

本文通过对独轮机器人的工作原理进行分析，为独轮机器人挑选合适的电子元件。本文选择 mpu9250 电子陀螺仪捕获独轮机器人运动姿态；GA12-N20 电机为独轮机器人提供驱动力；电机驱动使用 L298N 电机驱动模块；两节 18650 电池供电；arduino uno 板作为独轮机器人的核心控制器。

为解决电子陀螺仪的数据带有偏差和噪声的问题，本文对数据进行修正处理，受电容的滤波特性的启发，设计滤波算法。本文从电容滤波电路切入，进行分析，对电容进行数学建模。针对计算机处理数据的特点，本文对模型进行离散化处理，并使用递推公式取代求和公式。本文使用 MATLAB 软件对电容滤波算法进行了初步的检验分析，并使用 arduino 实物进行实践验证。由实验结果可知该算法能有效滤除信号的噪声。为使数据更有可读性，更便于数据分析。本文对数据进行相应的处理，从而获得了角速度、角加速度、角度三个物理量。

独轮机器人的动力学系统具有多变量强耦合的特点，为使设计简单。本文对独轮机器人的倾倒过程拆解成前后倾倒与左右倾倒的组合。对每个轴向的倾倒运动独立分析、建立动力学模型，并对控制系统进行设计与仿真。在左右倾倒控制的设计中，通过分析倾倒过程中受到的力的变化、角度的变化、角速度的变化，得到了 PD 控制系统。但在左右倾倒仿真中发现，该控制系统不能区分出独轮机器人的倾倒和回摆的区别。

为解决控制系统无法识别回摆与倾倒的区别的问题，本文对倾倒过程再度分析。在原控制系统上加入了积分项，也就是角度参数。角度参数可以使得控制系统清楚独轮机器人需要恢复到何位置。本文再次对控制系统进行了检验，发现此时的控制系统已经能够区分回摆和倾倒，并能够使得独轮机器人经过几次抖动后便恢复到直立状态。

有了左右倾倒的控制经验，前后倾倒的控制系统直接采用了角度、角速度、角加速度三组数据同时控制。由于前后倾倒会导致有位移发生，文中使用的仿真程序，加入了动画，可实时查看独轮机器人的姿态。从仿真的数据曲线可以看出，所设计的 PID 控制系统完全可以满足独轮机器人站立的控制需求。

本文为使独轮机器人更具特色，所以对独轮机器人进行了润色。本文为独轮机器人设计了无线充电装置。初期制作了特斯拉线圈进行无线充电，实践中发现特斯拉线圈能放电的特点存在很大危险性，故改使用线圈式的无线充电装置。文中对线圈式无线充电进行了实际检测，由测试的数据得知，本文制作的线圈式无线充电装置功率足够独轮机器人使用。

关键词：独轮机器人，滤波算法，无线充电，特斯拉线圈，PID 控制算法，动力学建模

Abstract

The one-wheeled robot has a nonlinear and strongly coupled complex dynamic system. It is a great challenge to control theory and dynamics research, and it is a very good research model. This paper will design a one-wheeled robot, focusing on the analysis of the movement of the one-wheeled robot, and establish a dynamic model and write a control program.

This paper analyzes the working principle of the one-wheel robot and selects the appropriate electronic components for the one-wheel robot. In this paper, the mpu9250 electronic gyroscope is selected to capture the motion posture of the one-wheel robot. The GA12-N20 motor provides the driving force for the one-wheeled robot. L298N motor drive module is used for motor drive. Two 18650 batteries powered; Arduino uno board is used as the controller of one-wheel robot.

In order to solve the problem of deviation and noise in the data of the electronic gyroscope, this paper corrected the data and designed a filtering algorithm inspired by the filtering characteristics of the capacitance. In this paper, from the capacitor filter circuit, analysis, capacitance mathematical modeling. According to the characteristics of computer data processing, this paper discretizes the model and uses the recursive formula to replace the summation formula. In this paper, the capacitance filtering algorithm was preliminarily tested and analyzed with MATLAB software, and the actual arduino was used for practical verification. The experimental results show that the algorithm can effectively filter the noise of the signal. In order to make the data more readable, more convenient for data analysis. In this paper, the corresponding data processing, so as to obtain the angular velocity, angular acceleration, Angle three physical quantities.

The dynamic system of one-wheeled robot is characterized by strong coupling of multiple variables, which makes the design simple. In this paper, the dumping process of one-wheel robot is disassembled into a combination of front and back dumping and left and right dumping. The dumping motion of each axis is analyzed independently, the dynamic model is established, and the control system is designed and simulated. In the design of left-right dumping control, the PD control system is obtained by analyzing the change of force, Angle and angular velocity in the process of dumping. However, in the simulation of left and right dumping, the control system could not distinguish the dumping and backswing of the one-wheeled robot.

In order to solve the problem that the control system cannot recognize the difference between backswing and dumping, this paper analyzes the dumping process again. The integral term, which is the Angle parameter, is added to the original control system. The Angle parameter allows the control system to know where the one-wheeled robot needs to be restored. In this paper, the control system has been tested again, and it is found that the control system has been able to distinguish back swing from toppling, and can make the one-wheeled robot return to the upright state after several wobbles.

With the control experience of right and left toppling, the control system of front and back toppling directly adopts the simultaneous control of three sets of data including Angle, angular velocity and angular acceleration. As the forward and backward dumping will lead to displacement, the simulation program used in this paper is animated to view the posture of the one-wheeled robot in real time. It can be seen from the simulation data curve that the designed

PID control system can completely meet the control requirements of standing one-wheeled robot.

In order to make the one-wheel robot more characteristic, this paper has embellished the one-wheel robot. This paper designs a wireless charging device for a one-wheel robot. At the initial stage, tesla coil was made for wireless charging. In practice, it was found that tesla coil could discharge in a very dangerous way, so a wireless charging device of coil type was used instead. In this paper, the coil wireless charging device is tested in practice. According to the test data, the coil wireless charging device made in this paper is powerful enough for the one-wheel robot to use.

Keywords: one-wheel robot, filtering algorithm, wireless charging, tesla coil, PID control algorithm, dynamics modeling

目 录

1 引言.....	1
1.1 研究意义.....	1
1.2 国内外研究现状.....	1
2 原理分析.....	4
3 电路元器件选型.....	5
3.1 传感器.....	5
3.2 电机.....	6
3.3 电机驱动.....	7
3.4 电池.....	8
3.5 控制器.....	9
4 控制系统设计.....	12
4.1 mpu9250 数据修正处理.....	12
4.2 mpu9250 数据滤波处理.....	12
4.2.1 建立滤波模型.....	13
4.2.2 算法效果测试.....	14
4.2.3 实际的滤波分析.....	17
4.3 数据物理意义化.....	18
4.4 独轮机器人左右倾倒PID控制.....	19
4.4.1 左右倾倒动力学建模.....	20
4.4.2 左右倾倒的控制系统建立与仿真分析.....	21
4.5 独轮机器人前后倾倒PID控制.....	23
4.5.1 前后倾倒动力学建模.....	23
4.5.2 前后倾倒的控制系统建立与仿真分析.....	24
5 润色功能.....	26
5.1 无线充电介绍与原理分析.....	26
5.2 一代无线充电实物制作.....	26
5.3 二代无线充电实物制作.....	28
5.4 无线充电实测.....	29
6 总结.....	31
参考文献.....	32
谢 辞.....	33
附 录.....	34

1 引言

在科技发达的今天，机器人已经不再是什么陌生的名词。简单的有我们家中常用的扫地机器人，复杂的有工业生产中的机械臂。机器人已经走进了人们的生活，服务的领域包括餐饮、教育、军事、娱乐、农业、工业制造等等。自 2013 年以来，我国便连续五年工业机器人需求市场全球最大，机器人销量和市场增速稳居世界前列。可见我国机器人市场大，发展迅猛。

回顾机器人的发展史，机器人从低级到高级，可分为三个阶段。第一阶段为程序控制机器人，机器人只能按照预先编好的程序一步一步地进行工作；第二阶段为自适应机器人，其特点是配备有相应的传感器，能够从周围环境获取信息，并做出相应的动作；第三阶段的机器人相比较第二阶段的机器人，第三阶段机器人具有思维能力，表现更为智能。

在机器人当中可移动的机器人具有很强的适应能力，可以搭载上其他的功能设备。可移动机器人中有很多很多种，陆地机器人中有足式、轮式。海上机器人有螺旋桨式推进、有仿生的尾鳍推进。天上的有固定翼和多旋翼。

独轮机器人是轮式机器人中特殊的一种。独轮机器人的特点是仅有一个轮子着地，所以其自由度较大，稳定性较差。但独轮机器人有其他轮式机器人不具备的优势，独轮机器人占地面积小、体积小、运动灵活、可零半径旋转。

1.1 研究意义

独轮机器人具有很大的自由度，他具有一个多变量、强耦合、非线性的复杂动力学系统。其站立时与运动时都具有很强的不稳定性，是控制工程一大挑战，是一个很好的研究模型。对独轮机器人的研究可以促进控制理论与动力学的发展^[1]。其次，独轮机器人能更好地服务于人员较多的地方。因独轮机器人的动作容易被人所预测，我们能较轻松地预测出独轮机器人将要运动的方向以及运动速度。独轮机器人的运动相比较其他的机器人有些许特殊。因为一个轮着地的原因，在其运动时需要先发生小角度的倾斜。这个动作能很容易被观测到，在独轮机器人附近的人便可通过此变化预测出独轮机器人的运动。其他轮式机器人运动显得像突然发生的，没有征兆，难以预知。对人而言，像独轮机器人这样的机器人在发生运动时，显得更平缓，更舒适。

1.2 国内外研究现状

2008 年日本村田制作所推出了一款博人眼球的独轮机器人——村田女孩（Murata Girl）^[2]。村田女孩结构由轮子踏板、机器人腿部结构以及封闭在内的惯性飞轮组成。通过陀螺仪进行检测倾斜。该独轮机器人高五十厘米，重五千克，移动速度约五厘米每秒。图 1.1 是村田女孩的实物图。



图 1.1 村田女孩

村田女孩加入了用于测距的传感器，可以用来躲避障碍物。除此还配有蓝牙模块，可以用于接发数据和控制指令。村田女孩的缺点是不可以改变偏航角，也就是不可以像人一样原地左右转。但这其实并不影响村田女孩的运动，村田女孩依旧可以通过惯性飞轮和触地轮的配合进行改变运动方向^[3]。

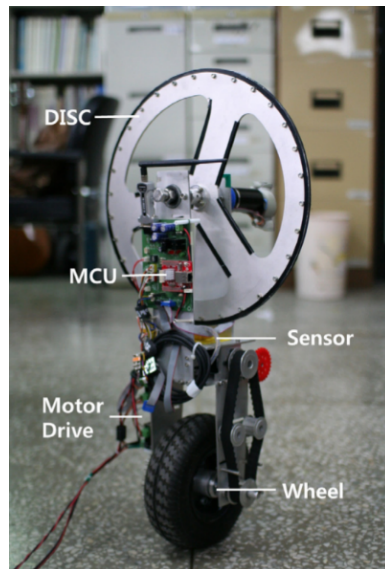


图 1.2 釜山国立大学独轮机器人

韩国的釜山国立大学对独轮机器人也进行了相应的研究。他们设计的独轮机器人如图 1.2 所示，并设计了一种独轮机器人的解耦控制算法。俯仰轴和滚动轴分开控制，假设两者在动力学上是分离的。他们所设计的独轮机器人同样不具备偏航能力^[4]。

相比国外，我国对独轮机器人的研究起步较晚。2009 年，北京工业大学研究团队对独轮机器人进行了较为深入的分析研究。他们对独轮机器人进行了动力学研究、建立模型、对独轮机器人的结构进行设计、控制器设计以及速度跟踪。图 1.3 是北京工业大学研究团队设计的独轮机器人。

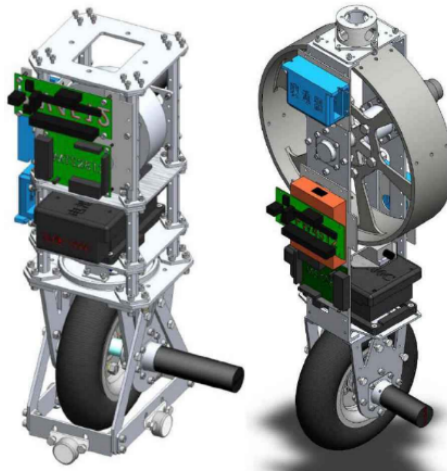


图 1.3 北京工业大学研究团队的独轮机器人

北京工业大学的研究团队对独轮机器人的研究成果有：独轮机器人动力学建模与分析，非线性 PD 控制方法对独轮机器人运动时平衡进行控制，基于非线性动态逆控制方法的独轮自平衡机器人平衡控制，基于迭代学习控制方法的独轮自平衡机器人运动平衡控制^[5]。

2010 年，哈尔滨工业大学的熊梅对独轮机器人的姿态控制算法上进行了研究。熊梅分析了独轮机器人的运动规律建立了动力学模型。简化了拉格朗日动力学方程，运用于独轮机器人的实时控制上。设计了变增益平衡控制算法^[6]。

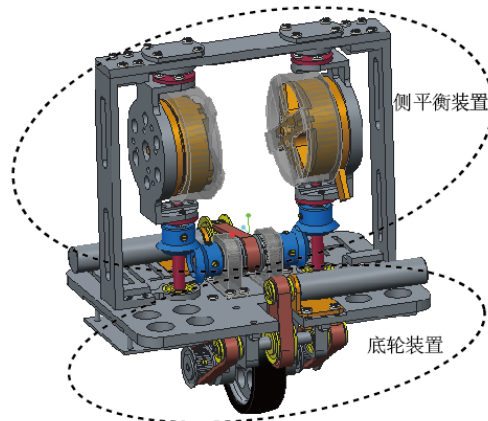


图 1.4 哈尔滨工业大学于法传设计的独轮机器人

2015 年，哈尔滨工业大学的于法传为了解决使用惯性飞轮的独轮机器人输出扭矩小等问题，设计了一款利用陀螺进动效应的独轮机器人。图 1.4 是于法传的独轮机器人设计。控制方面使用解耦控制方法，并使用一种基于离心力原理的补偿控制算法解决系统抗干扰能力欠缺的问题^[7]。

2 原理分析

为了给后续的元器件选择以及控制程序的编写做准备，我们需要对独轮机器人的运动进行初步的分析与研究，本文将先简单分析独轮机器人的工作原理。

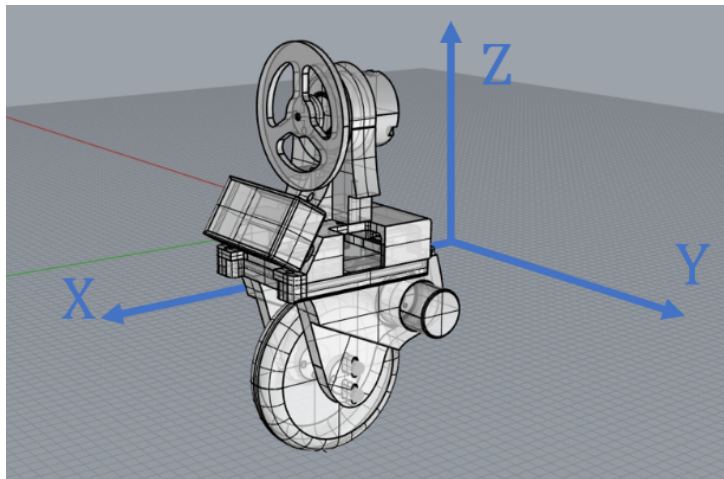


图 2.1 独轮机器人

由于独轮机器人只有一个轮子着地，故其自由度比较大。在空间直角坐标系上独轮机器人可以绕三个轴自由旋转，朝两个轴向自由平移。独轮机器人保持直立状态，需要有控制两个轴向的旋转运动。如图 2.1 所示，控制 X 轴保持独轮机器人不左右倾倒，控制 Y 轴保持独轮机器人不前后倾倒。

独轮机器人的车轮与地面间可以产生一个作用于独轮机器人的摩擦力。由于独轮机器人的重心不在此力的作用线所在的直线上，故可向独轮机器人提供扭矩。当独轮机器人发生前后方向的倾斜，并有倾倒的倾向时，车轮迅速运动，向独轮机器人提供一个反向的扭矩，使得独轮机器人恢复直立状态。

独轮机器人的胸前有一飞轮，用于保持 X 轴方向的平稳。平常飞轮处于静止状态，将飞轮与独轮机器人视为一个整体，则该整体初始的角动量为零。当独轮机器人发生左右倾斜，并有倾倒的倾向时，背部飞轮由静止开始加速旋转。飞轮受到扭矩的作用而获得某一方向的角动量，由于力的作用是相互的，故独轮机器人的主体部分将受到相反的扭矩而发生小的转动，从而抵消机器人的左右倾斜，独轮机器人将恢复直立状态。

由上分析可知，独轮机器人的平衡控制问题实际上是根据独轮机器人的实时运动姿态控制车轮和飞轮运动的问题。

独轮机器人在倾倒时实则是绕某一轴发生的加速转动，故需要读取独轮机器人的角速度并进行分析即可得知其运动姿态。本文设计的独轮机器人将使用电子陀螺仪读取角速度信息。

3 电路元器件选型

3.1 传感器

对独轮机器人的工作原理进行分析可知，独轮机器人需要保持直立状态，则控制系统的核心输入参数是其角速度。电子陀螺仪是最常用的读取角速度的传感器。电子陀螺仪又叫加速度计、角速度传感器，因为电子陀螺仪模块通常是集成测加速度、测角速度等功能的传感器。因运用场景不同，有些电子陀螺仪传感器模块甚至集成了测温度，测气压、测地磁等功能。

在电子陀螺仪众多的型号中，mpu6050 传感器模块是一款比较常见比较常用的测角速度的传感器模块，其核心是一颗 mpu6050 芯片。该芯片是一款六轴的电子陀螺仪，可检测三个轴向角速度以及三个轴向的加速度。

加速度测量是通过电容的改变测得。内置小型的特殊装置制作的电容，当外部受到力的作用时，会使得小型的特殊装置发生位移引起电容电极距离改变，从而引起电容大小的改变。角速度的测量则是利用了科里奥利效应，通过检测科里奥利力大小来获得角速度。

mpu6050 通过 I2C 方式进行通信。角速度测量可选量程有 $\pm 250^\circ/s$ 、 $\pm 500^\circ/s$ 、 $\pm 1000^\circ/s$ 、 $\pm 2000^\circ/s$ ，加速度测量可选量程有 $\pm 2g$ 、 $\pm 4g$ 、 $\pm 8g$ 、 $\pm 16g$ ，数据输出为十六位数据。对于独轮机器人的是使用是完全可以满足的。

本文所设计的独轮机器人对电子陀螺仪的精度、体积、功耗等均没有严格要求，市面上的大部分电子陀螺仪均可服务于本文所设计的独轮机器人。故从经济的角度出发，mpu6050 模块是最适合本独轮机器人的一款六轴电子陀螺仪。

由于在笔者设计本独轮机器人时，恰巧闲置着一个型号为 mpu9250 的电子陀螺仪模块。

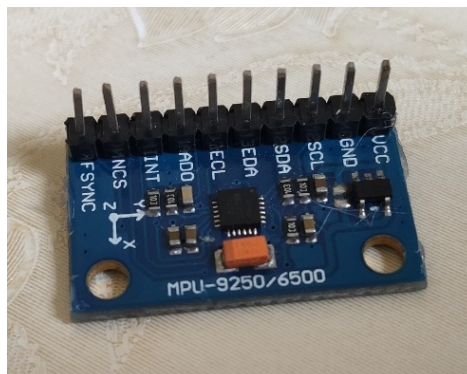


图 3.1 mpu9250 模块实物

图 3.1 是 mpu9250 模块的实物图，它是一款九轴的电子陀螺仪模块，其核心为一颗 mpu9250 芯片。其基本信息见表 3.1 所示。

表 3.1 mpu9250 基本信息

芯片	mpu9250
电压	3V-5V
通讯方式	I2C/SPI
数据输出	16 位有符号整型数据
角速度计范围	$\pm 250^\circ/s$ 、 $\pm 500^\circ/s$ 、 $\pm 1000^\circ/s$ 、 $\pm 2000^\circ/s$
加速度计范围	$\pm 2g$ 、 $\pm 4g$ 、 $\pm 8g$ 、 $\pm 16g$
磁力计范围	$\pm 4800\mu T$
外观尺寸	15mm \times 25mm

mpu9250 内含三个轴向的加速度计、三个轴向的角速度计、三个轴向的磁力计。角速度测量可选量程有 $\pm 250^\circ/s$ 、 $\pm 500^\circ/s$ 、 $\pm 1000^\circ/s$ 、 $\pm 2000^\circ/s$ ，加速度测量可选量程有 $\pm 2g$ 、 $\pm 4g$ 、 $\pm 8g$ 、 $\pm 16g$ ，磁场测量量程为 $\pm 4800\mu T$ ，数据输出为十六位数据。基本功能及相应参数与 mpu6050 无异，对于本文所设计的独轮机器人亦是足够。

从经济的角度分析，合理地利用闲置的资源减少经济浪费也不失为良计。故本文所设计的独轮机器人将放弃选用 mpu6050 模块，改用 mpu9250 模块。

3.2 电机

本文所设计的独轮机器人，体积较小，固使用的电机也需要体积较小。同时独轮机器人的平衡需要电机具有较强的爆发力，能及时作出相应的响应。在智能机器人中，有一款小型直流电机被经常性使用，这便是 GA12-N20 电机。GA12-N20 电机使用的是直流电，电机控制简单，自带减速器，可减少很多的工作量，且体积小扭矩大，可适用于很多地方。

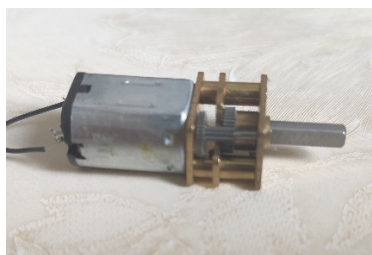


图 3.2 GA12-N20 直流电机

图 3.2 是 GA12-N20 电机的实物图，电机的整体高度 10mm，宽度 12mm，长度 25mm，输出轴 10mm。可见该电机是一个非常小巧的电机。本文所设计的独轮机器人飞轮电机需要侧重转速，车轮电机负载较大侧重扭矩。所以，本文所设计的独轮机器人需要两种不同转速的电机。高转速的电机参数为：额定电压 6V，额定转速 1500r/min，额定扭矩 0.02kg·cm；低转速的电机参数：额定电压 6V，额定转速 340r/min，额定扭矩 0.17kg·cm。

3.3 电机驱动

独轮机器人的控制信号是由控制器产生的，而控制器的输出能力往往很有限，只能充当信号使用。而电机的转动往往需要较大的电流，即使是迷你小电机也需要几十毫安到几百毫安的电流。这对于控制器而言实在太大，控制器不能够直接驱动电机旋转，需要外加一个驱动设备。

GA12-N20 电机是一款无刷的直流小电机，直流电机的控制可通过控制电压大小以达到控制转速的效果。由于独轮机器人的各个电机均需要做到可正转可反转，即需要电机驱动能够改变电压方向。

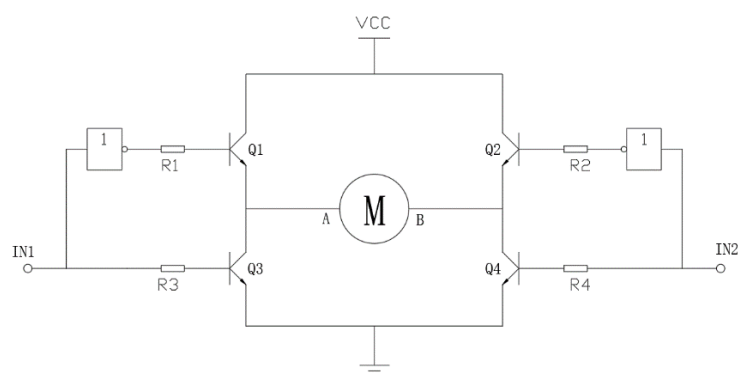


图 3.3 电机驱动电路

图 3.3 所示的是一款简易的电机驱动电路。电路的核心是四个三极管，通过控制三极管的打开与关闭来达到控制电机的转动。当 IN1 输入为负，IN2 输入为正：Q1 三极管与 Q4 三极管打开，Q2 三极管与 Q3 三极管关闭，电机的电流由 A 点流入，B 点流出，电机正转；当 IN1 输入为正，IN2 输入为负：Q2 三极管与 Q3 三极管打开，Q1 三极管与 Q4 三极管关闭，电机的电流由 B 点流入，A 点流出，电机反转。当 IN1 与 IN2 均为正，A 点与 B 点相当于接地，电机停止旋转。

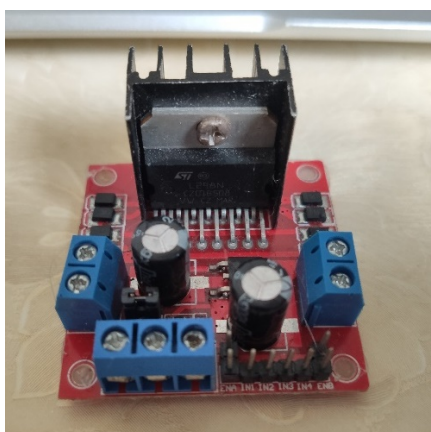


图 3.4 L298N 模块实物

L298N 芯片，是由上述图 3.3 电路为基本原理封装起来的电机驱动芯片，图 3.4 是 L298N 模块实物图。L298N 拥有两个全桥输出，可同时驱动两个电机。独轮机器人有两

个核心电机，分别控制飞轮转动和车轮转动，L298N 能刚好满足需求。

表 3.2 L298N 基本信息

类型	双路全桥
输出数	4
输出电流	2A
驱动电压	5V-35V
逻辑电压	5V

表 3.2 所示是 L298N 电机驱动芯片的基本信息。独轮机器人使用的电机中额定电流最大的是 0.15A，而 L298N 的电流输出达到 2A，故 L298N 完全能够满足独轮机器人的需求。本文所设计的独轮机器人将使用 L298N 作为电机驱动芯片。

3.4 电池

电池的类型相对较少，大体上分为一次性电池和可充电电池。通常在电子设备极其廉价、使用寿命较短、使用频率较低或者耗电功率极小等情况才会考虑采用一次性电池，否则首选考虑可充电电池。本文所设计的独轮机器人显然不符合上述情况。独轮机器人不属于极低功耗的设备，使用一次性电池将会导致经常性更换电池，这会对环境造成较大污染。

现今较为常用的可充电电池有两种。一种是较为老式的铅蓄电池，一种是常出现在电子设备上的锂电池。两种电池各有特点。铅蓄电池的特点是价格便宜，鲁棒性强，铅蓄电池在充电的时候可以较为简单粗暴。在很多电子设备上的铅蓄电池充电电路连稳压恒流都不设计。充电电路简单地将家庭用电使用阻容降压，然后全桥整流便直接接到电池上给电池充电。充电电路连充电完成指示灯都不设计，充电情况全靠估计。即使如此粗暴对待，铅蓄电池依旧能保持较长的使用寿命。但铅蓄电池的能量密度较低，通常铅蓄电池显得特别笨重，这是它的一个较严重的缺点。因而在许多智能设备上，一般不会选择使用铅蓄电池。

锂电池相比铅蓄电池优点是能量密度高、自放电率低，所以锂电池能做到比较小比较轻便。但锂电池同样存在着其他问题。相比铅蓄电池，锂电池实在娇气。不恰当的充电放电将会导致锂电池寿命大减、膨胀甚至是爆炸。锂电池的充电比较麻烦，需要专用的电路为锂电池充电。锂电池充电一般分为三个阶段，第一阶段为涓流充电，该阶段是在电池长期放置使得电池已经完全没有电时进行的预充电。充电器将提供细小的恒定电流为电池充电，待电池恢复 3.7V 电压后进入第二个充电阶段。第二阶段为恒流充电，该阶段充电器将提供恒定电流进行充电，电流大小一般为 0.1 到 1.5 倍电池容量。当电池电压上升到 4.2V 后，进入最后的恒压充电阶段，该阶段充电器将提供 4.2V 恒定电压对电池进行充电，直到充电电流小于某个阈值，充电结束。

尽管锂电池的充电相对麻烦，但是体积和重量是独轮机器人更不希望增加的。大的

体积和重量将会造成独轮机器人电机有很大负担，控制难度也将上升，这将得不偿失。且独轮机器人属于智能设备，有良好的智能电路，解决锂电池充电问题相对简单。因此，本文所设计的独轮机器人将选择型号为 18650 的锂电池作为电源。两节电池串联供电，单节电池容量为2800mAh。



图 3.5 18650 锂电池实物图

图 3.5 是本文所将要使用的 18650 锂电池实物图。18650 锂电池是一款十分常用的圆柱形锂电池，其型号含义为直径 18mm，高度 65mm 的圆柱形锂电池。21700 锂电池比 18650 的体积更大，做工更好，价格相对 18650 高。本文所设计的独轮机器人使用 18650 锂电池即可。

3.5 控制器

现今的控制器有众多的类型。传统的有单片机，有利用单片机做二次开发的 PLC 与 arduino，还有性能接近家用计算机的树莓派。单片机中又有相对老式的八位、十六位单片机，有性能相对较好的三十二位单片机。控制器是智能产品中的核心的元件，控制器在设备中起到响应传感器的输入、处理数据、发出控制信号等作用。

PLC 又称为可编程逻辑控制器，PLC 的产生是为了替代工业上继电器实现的逻辑控制。PLC 相对单片机有抗干扰能力强、编程语言简单、电流输出能力较强等优点。但 PLC 的体积较大，价格也比较昂贵。实际上，PLC 更适合在工业上的、大型设备的情况下使用。

树莓派又名 Raspberry Pi，事实上树莓派应该说是一个计算机。它是为了学习计算机编程教育而设计的。因为现在有不少的科研工作者使用树莓派作机器人的控制器，故笔者在此将提到树莓派。树莓派是一个微型计算机，体积仅一个信用卡大小。树莓派需要安装系统才可使用，树莓派使用的是 Linux 系统。近期发布的 Windows 10 IoT 系统，使得树莓派可以使用上大家更为熟悉的 Windows 系统。树莓派尽管较小却五脏俱全，树莓派可以连接鼠标、键盘、显示器、摄像头、扬声器等外部设备。树莓派的性能优越，能处理大量复杂的数据、运行较大的程序、实现机器视觉。树莓派性能强悍，但其设计初衷是一台微型计算机，尽管可以通过 I/O 口控制外部设备，但如此使用树莓派作控制

器难免会显得有些浪费资源。一般而言树莓派作为机器人控制器时，往往是作上位机使用，仅进行复杂数据处理、运行核心算法。底层硬件控制会选择使用其他控制器实现。

在高校中单片机是非常经常被使用到的控制器，高校学生中比较常用到的是 51 系列的单片机，其次为 stm32 的单片机。stm32 是一款三十二位的单片机，处理能力较为强悍，因而受到开发者的青睐。但可惜的是实际上，高校中很多的项目往往并不需要如此高的处理能力，多数是大材小用。对于简单重复的工作，对响应速度要求不高的场合使用 stm32 单片机并不能体现出其优势。stm32 单片机因为其强大的能力，也导致出现了一个弊端。该系列的单片机的使用比较麻烦，较多的功能意味着要进行较多的功能设定操作。

51 系列的单片机是一款比较老式的单片机，在性能上几乎没有优势可言。但由于该系列的单片机在当时简单好用，所以被视为一款经典的单片机。往后发展的单片机在设计上延续了 51 系列优点，因而单片机入门者可选择从 51 系列入手。尽管直观上 51 系列的单片机没有优势，但因其经典、入门者首选、常被教科书使用等等原因。51 系列的单片机隐含着其他方面的优势。51 单片机使用简单，资源丰富，存在着大量的教程和源代码。而且 51 系列的单片机价格低廉。

arduino 也是由单片机再开发的一款控制器，使用 C 语言编程。arduino 在实际使用的体验上，更接近于一款单片机，通常 arduino 的大小与一张银行卡相似。arduino 诞生的初衷是为了得到一款廉价好用的控制器，所以 arduino 的价格要比 PLC 廉价很多，但依然比单片机昂贵。arduino 相对单片机而言具有许多优点。arduino 在经过前人的努力后，许多常用的功能已经被封装成函数，可直接调用。在程序烧录上，arduino 的编程软件可直接烧录，接线上直接使用数据线连接 PC 机即可进行通信。能通过串口监视器与串口绘图器实时获取 arduino 的数据。arduino 相对于单片机而言体积稍大些，价格上也稍昂贵些。

在独轮机器人中，控制器需要读取 mpu9250 的数据、对数据进行滤波、分析数据、发出控制信号。由于本文所设计的独轮机器人，控制需求相对比较简单。对运算能力相对而言要求较低，并不含有大型的复杂的程序需要运行。因而笔者并不需要使用到性能高的控制器。

与 PC 机不同。计算机的选购往往需要超过现在使用的性能要求，因计算机的使用时间较长，应用场景复杂具有很多不确定性，软件不断地迭代更新也将会对计算机性能有更高的要求。计算机的性能一旦跟不上需求，使用起来将会带来很多的麻烦，给使用者造成很多不愉快的地方。

然而机器人控制器的性能选择往往是够用即可，高性能或许仅在后期改良更新功能的时候带来些许便利，过高的性能往往只会白白增加整机费用。机器人的功能往往是在设计时被一次性确定下来的，工作环境也不会有较大的改变。机器人通常是就一种实际需求，在相对单一的工作环境下，实现一些相对确定的工作。所以机器人的控制器不需

要过高的性能，适用即可。独轮机器人亦是如此，并不需要过度追求控制器的性能，只

需要就实际使用情况进行分析，选择性能最合适价格较低廉的即可。结合上述分析，为使得开发简单，后期升级方便，本文所设计的独轮机器人将采用 arduino uno 作为控制核心。

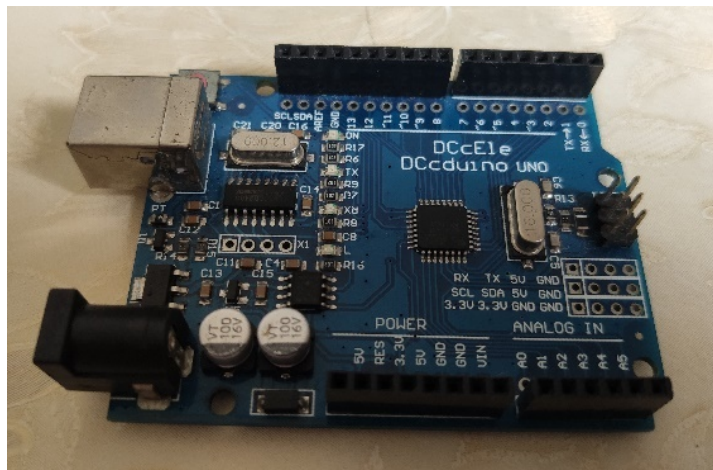


图 3.6 arduino uno 板实物

图 3.6 是 arduino 的实物图，本文将使用该板作为独轮机器人的控制核心。

4 控制系统设计

4.1 mpu9250 数据修正处理

直接从 mpu9250 模块读取的数据是不可直接使用的，必须经过一定的处理。就像天平使用前需要校零，同样的电子陀螺仪在使用前也需要对其进行修正处理。由于制作的误差等原因，静止状态的电子陀螺仪读取的角速度数据一般都不是零，而是在零附近的某个值，本文将称此值为偏差值。这并不表示电子陀螺仪检测到了角速度，实则是此电子陀螺仪以此数值来表示零角速度。大于此值则表示检测到于轴向相同的角速度，小于此值则表示检测到于轴向相反的角速度。加速度计同样存在上述问题，但不同的是加速度计还会受到重力的影响。水平方向的轴，在静止时不受重力影响。竖直向下的轴，在静止时读取的数据包含了一个重力加速度和一个偏差值。

要正常使用 mpu9250 读取独轮机器人的姿态，需要先处理偏差值。对于角速度计处理，只需尽可能保持 mpu9250 模块处于静止状态。读取 X、Y、Z 三个轴向的角速度数据，即偏差值。并将此数据保存到程序中，往后每次读取角速度数据时，均需要减去对应的偏差值即可。对于加速度计处理，先尽可能保持 mpu9250 模块处于静止状态，并且尽可能使得 mpu9250 的某两个轴如 X 轴、Y 轴处于水平方向。读取两个水平方向的轴的加速度数据，即偏差值。再翻转 mpu9250，使得另外一轴如 Z 轴尽可能处于水平方向且尽可能保持 mpu9250 模块处于静止状态。读取该轴的加速度数据，即偏差值。此时加速度计的各轴偏差值均已得到，将偏差值保存在程序中。往后每次读取加速度数据时，均需要减去对应偏差值即可。

4.2 mpu9250 数据滤波处理

从 mpu9250 模块读取的数据经过修正处理后仍然不可直接使用。电子陀螺仪除了存在零点偏移，数据中还包含着大量复杂的噪声。这是电子电路常有的问题，对于存在噪声的信号，我们会对其进行滤波处理。

滤波的含义是将信号中的某些特定频率的波形剔除的操作。噪声通常表现为高频率、无规律、振幅较小，相对而言信号常表现为低频率、有规律、振幅较大。由于噪声与信号的频率不同，我们便可以将噪声与信号分离。

为解决传感器的噪声问题，人们找到了很多的滤波算法。有以卡尔曼滤波算法为基础的滤波算法，如扩展卡尔曼滤波算法、无迹卡尔曼滤波算法及联邦卡尔曼滤波算法等。还有的是以互补滤波为基础的滤波算法，如经典互补滤波算法、显性互补滤波算法及基于梯度下降法的互补滤波算法等^[8]。

本文将不直接使用他人的滤波算法，而是自建模型，编写相应的算法程序并对其进行检测、结果分析。

4.2.1 建立滤波模型

电容器在电子电路中，是极其常见的一种电子元件。电容器又称电容，常出现在稳压电路、滤波电路中。电容器结构极其简单，仅两块靠近但不接触的金属片便可形成电容器。电容器在电路中具有充电与放电的功能，正因其充电与放电的能力，使得其能够稳定电路中的电压。电容器的电容值越大，其稳定电压的能力越强。但电路希望电压发生改变时，电容值越大，电压的改变将被延迟得越久。

电容是一个真实存在的实物，处理的信号是电路中时时存在的模拟信号。电子陀螺仪读取的数据却是离散的数字信号，是虚拟的不存在于实际的信号。要想利用电容处理电子陀螺仪数据，需要对电容进行分析，保留其滤波等核心特性，去除其非必要的特性，并对其做细微调整使得能处理离散的数字信号，最后在程序中虚拟出一个具有滤波稳压功能的“电容器”。

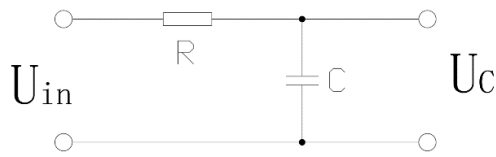


图 4.1 低通滤波电路

图 4.1 是一个最基本的低通滤波电路，该电路具有滤除高频信号，保留低频信号的特点。电容电压表达式为：

$$u_c = \frac{Q}{C} \quad (\text{式 4.1})$$

其中 Q 是电荷量，其值可通过计算初始电荷量加上电容充电电流对时间进行积分获得。其数学表达式为：

$$Q = Q_0 + \int_0^t i dt \quad (\text{式 4.2})$$

公式 4.2 中的电流 i 是电容的充电电流，当其值为正时表示电容正在充电，当其值为负时表示电容正在放电。由串联电路可得，该电流等于流过电阻 R 的电流。则电流可表示为：

$$i = \frac{u_{in} - u_c}{R} \quad (\text{式 4.3})$$

结合上述式 4.1、式 4.2、式 4.3 得到式 4.4。其中， u_0 等于 $\frac{Q_0}{C}$ ，其物理含义为电容的初始电压。

$$u_c = u_0 + \frac{1}{CR} \int_0^t (u_{in} - u_c) dt \quad (\text{式 4.4})$$

得到的式 4.4 是一个带有积分项的表达式。显然这对于计算机而言是无法处理的，需要对上述的公式进行离散化处理。

电子陀螺仪读取的数据，是离散的不连续的。读取到的数据相当于图 4.1 电路中的输入电压 u_{in} ，滤波后的数据相当于图 4.1 中的输出电压，即电容电压 u_c 。公式 4.4 中的变量改为数列形式，积分改用前 i 项求和形式。

$$u_{ci} = u_0 + \frac{\sum_{k=1}^i (u_{ink} - u_{c(k-1)})}{CR} dt \quad (\text{式 4.5})$$

公式 4.5 是经过处理的电压表达式。i 时刻的电容的充电电流等于 i 时刻的输入电压减去 i-1 时刻的电容电压的差,再除于电阻 R。由公式 4.4 可得 i+1 时刻的电容电压表达式:

$$u_{c(i+1)} = u_0 + \frac{\sum_{k=1}^{i+1} (u_{ink} - u_{c(k-1)})}{CR} dt \quad (\text{式 4.6})$$

式 4.6 减去式 4.5, 并进行化简处理, 得到电容电压的递推表达式:

$$u_{c(i+1)} = u_{in(i+1)} \frac{dt}{CR} + u_{ci} \left(1 - \frac{dt}{CR}\right) \quad (\text{式 4.7})$$

式 4.7 使用递推关系求解电容电压, 去除了耗运算耗时间耗内存的求和。此式只要求调出上一次的电压数据即可完成滤波, 能大大提高数据处理的效率。显然式 4.7 已经非常适合运用在计算机中进行数据处理。该模型得到的表达式与一阶滞后滤波算法十分相似, 不同的在于该模型包含时间在内。两次读取数据所运行的时间不发生变化的程序中, 电容滤波和一阶滞后滤波是一样的。而对于读取数据所运行的时间不能保证一致的程序中, 电容滤波和一阶滞后滤波便存在微妙的差异。

4.2.2 算法效果测试

本文将利用 MATLAB 软件, 就上述数学模型编写程序以检验模型的有效性。并对滤波效果进行分析。详细程序见附录 1。

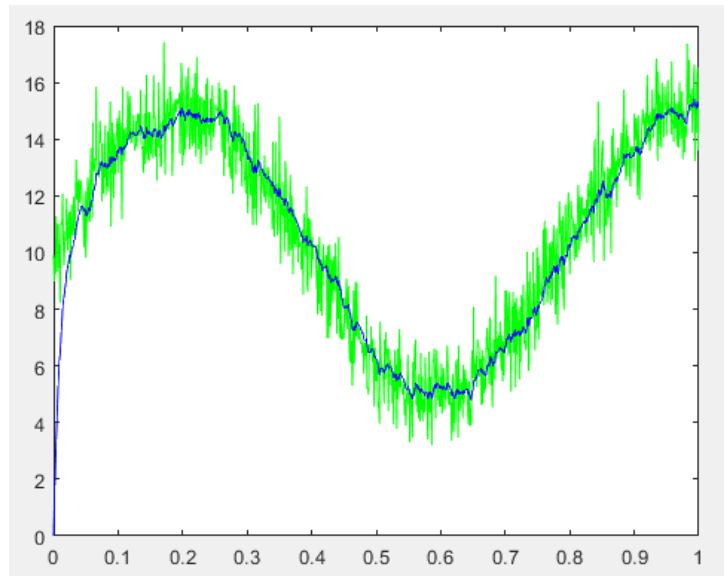


图 4.2 滤波后的信号与带噪声的信号

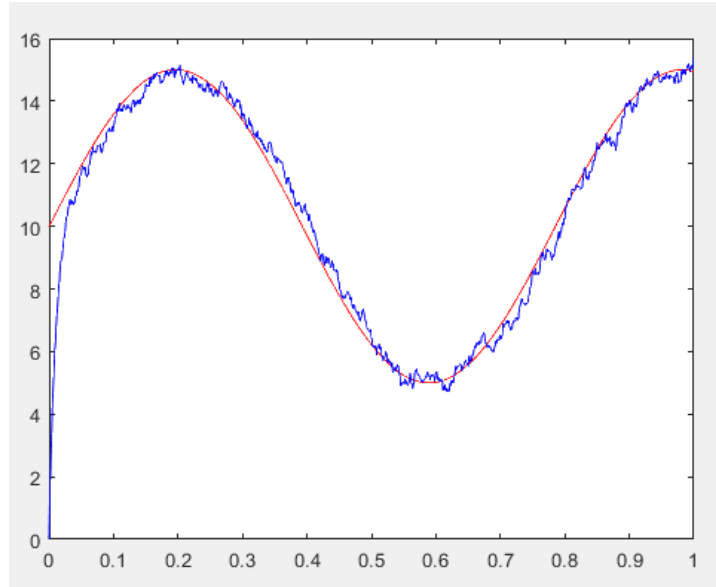


图 4.3 原信号与滤波后的信号

图 4.2 与图 4.3 的数据是在电容值为 0.0001、电阻值为 100、采样间隔时间为 0.001s、总时长为 1s 的情况下获得的。输入电压是一个正弦信号叠加了一个随机的噪声信号和一个常数项。正弦信号的角速度为 8，振幅为 5，初始相位为 0。噪声信号是一个服从标准正态分布的随机数。常数项为 10。

图 4.2 中可以看出滤波后的数据已经被大大地削弱了噪声，图 4.3 可以看出其还原后的曲线与原信号基本重合。

电容值、电阻值以及采样时间间隔与滤波效果密切相关，更改将会得到不同的效果。更改电容值为 0.0002、电阻值为 100，采样间隔时间为 0.0001s。得到以下数据图：

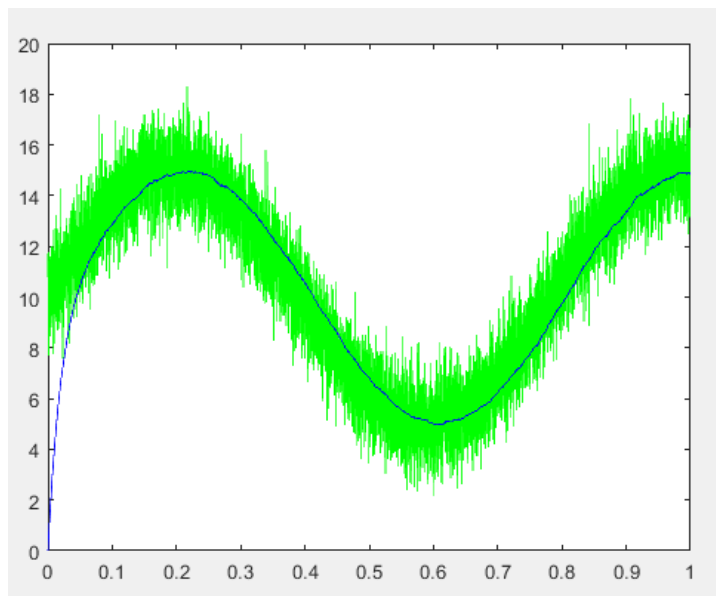


图 4.4 滤波后的信号与带噪声的信号

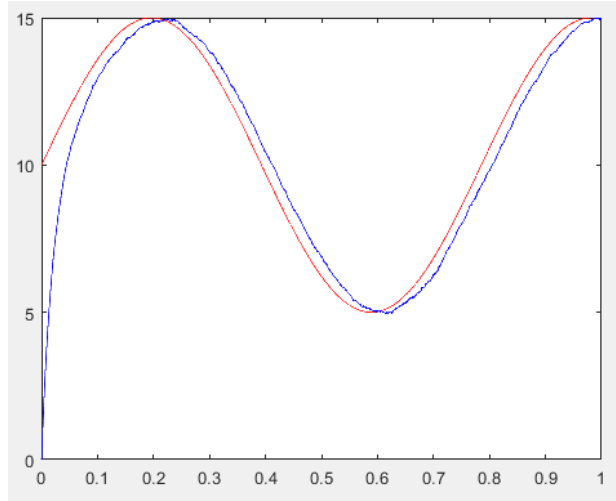


图 4.5 原信号与滤波后的信号

由图 4.2 与图 4.4 分析可得，该电容模型的整体滤波效果不错，能将大部分的噪声干扰过滤掉，由图 4.3 与图 4.5 分析可得，该电容模型能较好地复原出原信号，输出信号尽管有少许波动，但整体上波动较小。

在选择较大电容值及较小的采样时间间隔进行测验时，输出信号相对比较平滑稳定，但该参数下的电容模型付出了信号延迟滞后的代价。在选择较小电容值及较大的采样时间间隔进行测验时，输出信号能很好地跟随原信号的变化，但该参数下的电容模型将存在较多抖动，数据不够稳定。

表 4.1 噪声数据

最大正向噪声	4.078170
最大负向噪声	-4.197979
噪声的数学期望	-0.004753
噪声的方差	0.989805

表 4.2 滤波后的数据

最大正向误差	0.620305
最大负向误差	-0.686514
误差的数学期望	0.009389
误差的方差	0.137666

由于程序的开始时电容模型尚未“充电”，输出信号与输入信号间有较大的差距，电容需要经过一小段时间进行“充电”才能进入稳定工作状态。为使数据合理有效，误差分析应取电容工作稳定后的阶段进行。表 4.1 与表 4.2 所示的数据，均是从电容开始工作后的 0.2s 开始获取并进行分析的。其数据对应的电容值为 0.0002、电阻值为 100，采样间隔时间为 0.0001s。

由表 4.1 可得，电容模型稳定工作后，随机干扰的最大正向噪声为 4.078170，最大负向噪声为-4.197979，噪声的数学期望为-0.004753，噪声的方差为 0.989805。

由表 4.2 可得，电容模型稳定工作后，输出信号与输入信号的最大正向误差为 0.620305，输出信号与输入信号的最大负向误差为-0.686514，误差的数学期望为 0.009389，误差的方差为 0.137666。

对比表 4.1 与表 4.2 数据可知，该电容模型确实能有效滤除大部分的噪声，与原信号的误差能很好地控制在 0.7 以内。对于原信号有较好的还原效果。

4.2.3 实际的滤波分析

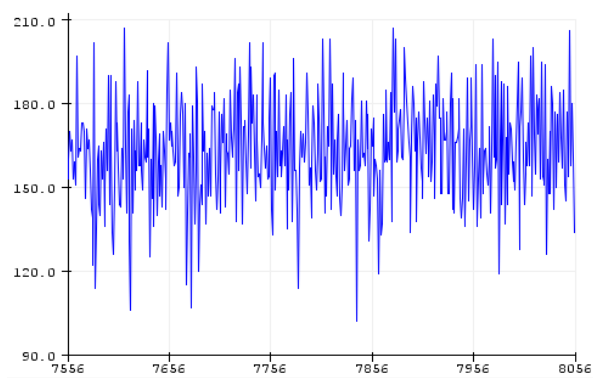


图 4.6 电子陀螺仪读取到的原始数据

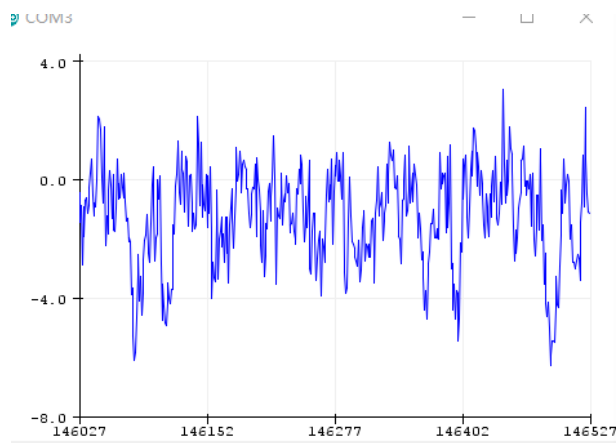


图 4.7 经过修正与滤波处理后的数据

图 4.6 是保持静止状态时，从 mpu9250 中直接读取到的角速度数据。纵坐标为数值，横坐标表示数据是开机后读取到的第 n 个数据，每个数据之间的时间间隔约为 6ms。从图中可以看出，数据包含了很大的噪声，峰峰值达到 100 左右。数据的数学期望也不在 0 上，而是大概处于 160 的位置。

图 4.7 是保持静止状态时，从 mpu9250 中读取到的角速度数据经过修正和滤波得到的数据。纵坐标为数值，横坐标表示数据是开机后读取到的第 n 个数据，每个数据之间的时间间隔约为 6ms。从图中可以看出，数据在经过修正和滤波以后，得到的数据基本

上都在 0 的附近，噪声信号已被大大削弱，峰峰值控制在了个位数。这个数值的影响对于满量程时的 32767 比较实在是微不足道。

由此可见，该模型能对数据进行有效的处理，滤除大部分的噪声信号，且简单实用。

4.3 数据物理意义化

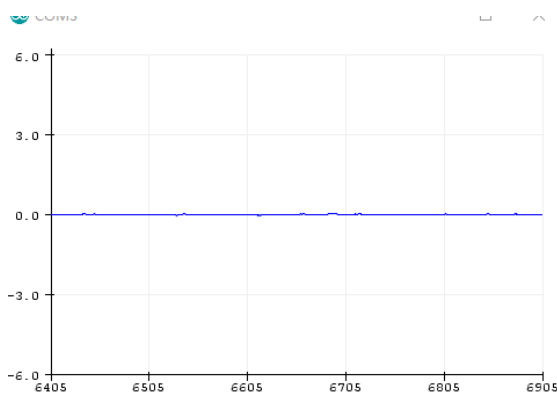
在经过修正与滤波之后的数据已经比较平滑比较接近真实数据。但该数据还不具有物理意义，为使得数据容易理解容易分析。因此我们需要对数据进行相应的转换使其化为具有物理意义的参数。

本文中读取角速度参数时，使用的是 $250^\circ/s$ 的量程。而 mpu9250 的数据输出是 16 位有符号整型数据，本文将需要把 16 位数据转换成 $-250^\circ/s$ 到 $250^\circ/s$ 之间的角速度数值。16 位有符号整型数据对应的是十进制 -32768 到 32767，正负各占有 32768 个数。也就是说当 mpu9250 输出 32767 时，表示角速度为 $250^\circ/s$ 。所以电子陀螺仪的输出参数需要乘 250 再除以 32768。

角加速度需要对角速度进行差分。在获取到角速度数据时，记录 arduino 的运行时间，待再次获取到角速度数据时，再记录 arduino 的运行时间，由此可得到两个数据的时间间隔。角速度数据后者减去前者，再除以时间间隔，便可获得角加速度数据。

角度可以通过对角速度进行积分获得。在学习定积分时，我们认为在 dt 的时间间隔中，速度不会发生变化，该时间内的小位移是小矩形面积。这个计算方式在时间间隔 dt 无限逼近于 0 时，求解结果是准确无误的。但由于数据获取、数据处理、发送控制信号等等的功能在运行，都需要耗费时间。电子陀螺仪的数据时间间隔无法取极限，若使用矩形求面积的方式会导致误差比较大，本文将使用梯形面积进行求和。方法是将现获取的角速度加上上一个角速度数据，再乘以 dt 除以 2。

由此分析编写 arduino 数据处理程序，详细程序见附录 2。



^① 时间微元，表示极短的时间。

图 4.8 角速度图像

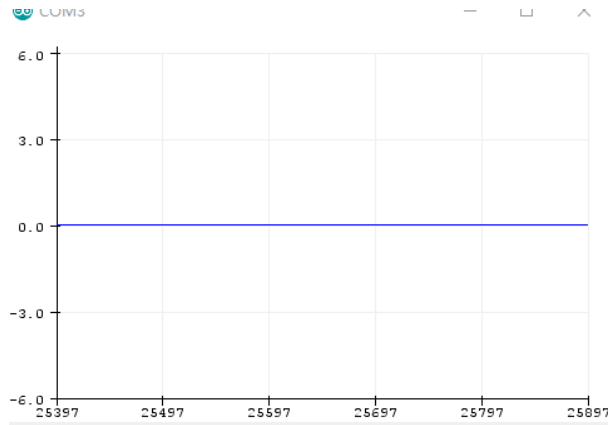


图 4.9 角度图像

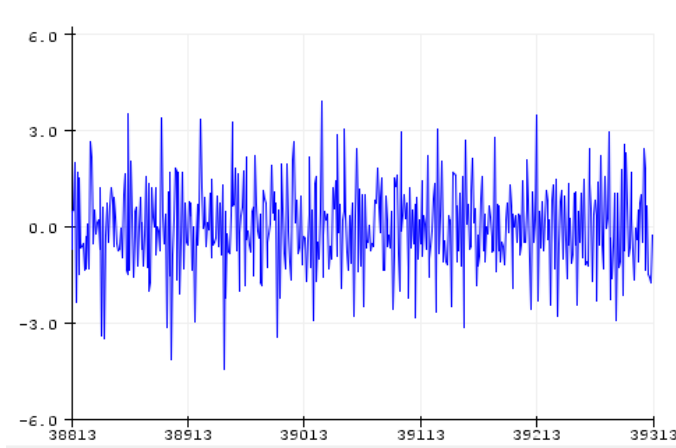


图 4.10 角加速度图像

图 4.8 是经过物理意义化的角速度数据，此时的 mpu9250 处于静止状态。纵坐标为角速度，单位是 $^{\circ}/s$ ，横坐标表示数据是开机后读取到的第 n 个数据，每个数据之间的时间间隔约为 6ms。仔细观察数据可以发现，数据中存在极小的波动，这属于正常现象，因为在修正与滤波后的数据任然存在小的噪声。

图 4.9 是经过物理意义化的角度数据，此时的 mpu9250 处于静止状态。纵坐标为角度，单位是度，横坐标表示数据是开机后读取到的第 n 个数据，每个数据之间的时间间隔约为 6ms。角度是由角速度积分得到的，角速度的波动已经很小。所以角度数据曲线非常稳定。

图 4.10 是经过物理意义化的角加速度数据，此时的 mpu9250 处于静止状态。纵坐标为角速度，单位是 $^{\circ}/s^2$ ，横坐标表示数据是开机后读取到的第 n 个数据，每个数据之间的时间间隔约为 6ms。角加速度数据是由角速度求导得到，角速度存在小的高频波动，求导后将波动放大。所以角加速度数据曲线存在小的波动。

4.4 独轮机器人左右倾倒 PID 控制

PID 控制算法，是一个十分常用的控制算法。PID 控制算法由三部分组成：比例、积分、微分。比例项是将系统输入与输出的偏差进行线性放大，作用效果十分显著，设置参数时先确定大概范围，再在范围内由大到小调节；微分项输入的是信号的变化率，实际的控制系统中，变化率可能有较大的变化。参数设置时不宜过大，过大影响比例项控制，应由小到大调节；积分项是为消除静态误差的。积分项有累积效应，静态时存在的偏差会导致随时间的增加而有较大的输出响应，积分项也将会导致系统有较大的“惯性”，系统会有较大的超调，应从小到大开始调节^[9]。

独轮机器人的动力学系统比较复杂，为使问题简化。本文对倾倒问题拆分为左右倾倒和前后倾倒两个方向。每个方向的运动分析和控制都独立进行。

4.4.1 左右倾倒动力学建模

左右倾倒是由飞轮进行控制的，在实际的控制中，会有众多的因素对控制造成影响。为简化模型，忽略次要因素，建模前需要进行以下假设：

- 1、忽略空气阻力及内部阻尼的影响。
- 2、假设车轮与地面的摩擦力足够大，工作过程中不发生打滑现象。
- 3、假设车轮与地面的接触点不因车轮弧度的影响而发生改变。

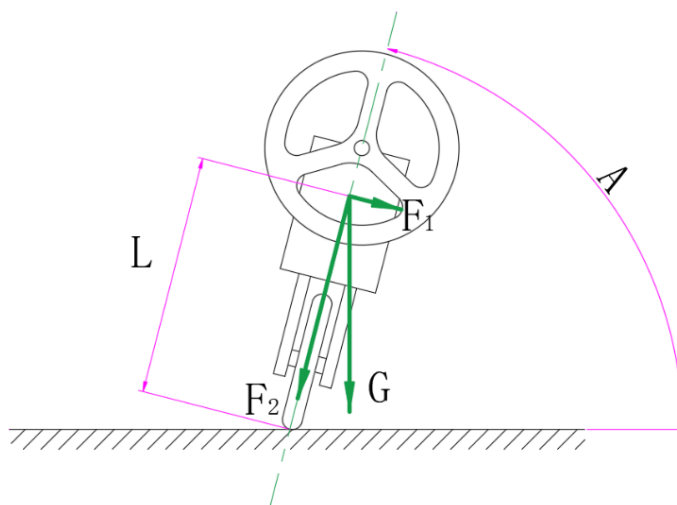


图 4.11 左右倾倒受力分析图

图 4.11 所示，是独轮机器人左右倾倒时的受力分析图。其中 A 是独轮机器人与地面的夹角， L 是独轮机器人重心到车轮触地点的距离。独轮机器人倒下的时候，是因为受到重力的作用。图中把重力分解成 F_1 、 F_2 两个力，在其中真正起作用的是 F_1 。由三角关系可以得出 F_1 与角度的关系：

$$F_1 = G \times \cos A \quad (\text{式 4.8})$$

独轮机器人倾倒的过程中，角度 A 在变小，故 F_1 在不断增大。故独轮机器人的倾倒是一个加速的过程。角度表达式如下：

$$A = A_0 + \int_0^t \int_0^t \frac{M}{J} dt + \omega_0 dt \quad (\text{式 4.9})$$

$$M = M_f - L \times F_1 \quad (\text{式 4.10})$$

其中， A_0 是独轮机器人初始时刻的角度， M 是受到的扭矩， J 是绕触地点旋转对应的转动惯量， ω_0 是初始时刻的角速度。 M_f 为飞轮提供的扭矩，总扭矩 M 是由飞轮和重力的分力 F_1 提供，方向与 L 与 F_1 叉乘相反。

4.4.2 左右倾倒的控制系统建立与仿真分析

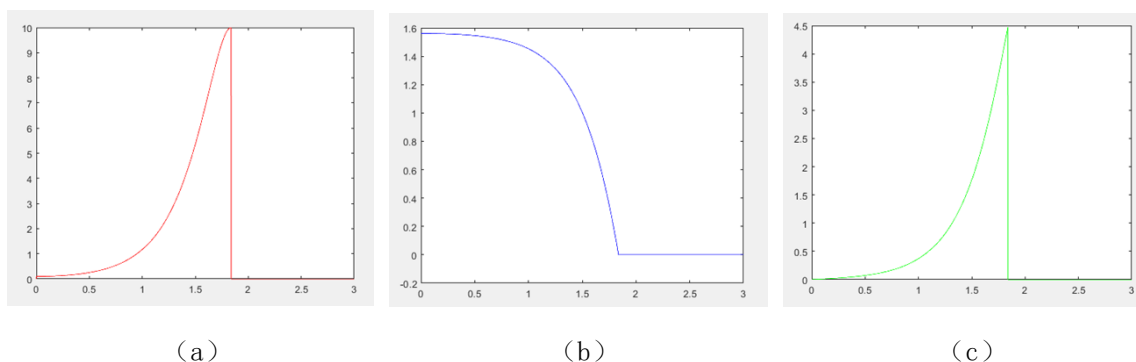


图 4.12 左右倾倒过程的运动学曲线

图 4.12 是在 MATLAB 中，模拟独轮机器人无控制左右倾倒而得到的三组曲线。图 4.12 (a) 为受力曲线图，图 4.12 (b) 为角度曲线图，图 4.12 (c) 为角速度曲线图。模拟程序见附录 3。由曲线可以看出，独轮机器人倾倒的前半部分时间，角度、角速度、以及受到的力变化都比较微小缓慢。在后半段时间内，受到的力、角速度迅速变化，爆发式增长。角度也随之迅速下降。

所以我们需要在独轮机器人倾倒的前半段时间内进行控制，以角速度为基本输入。微分得到的角加速度，能使得独轮机器人更早地预测到倾倒，并作出响应。

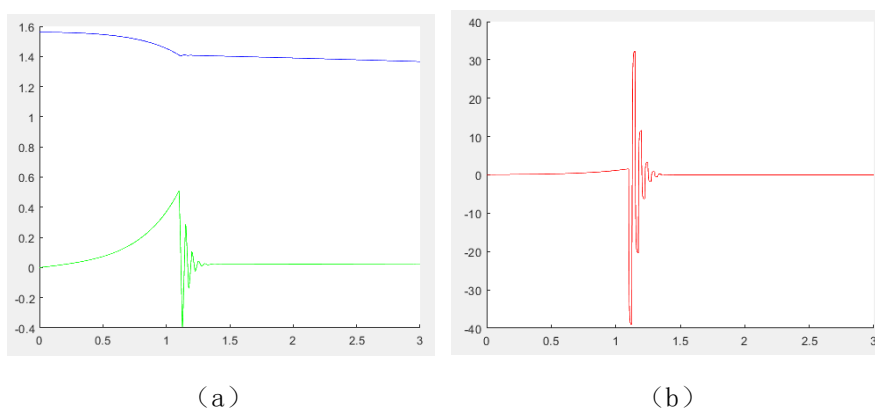


图 4.13 PD 控制仿真图

图 4.13 是使用了比例与微分两个参数进行控制得到的仿真图，具体仿真程序见

附录 4。图 4.13 (a) 中蓝色为角度图，绿色为角速度图，图 4.13 (b) 中红色曲线为合力图（重力分力与电机输出扭力的合成）。该图是先让机器人自由倾倒到一定角度后再进行 PD 控制得到的曲线图。

从图中可以看出，进入控制后角速度经过一段小的抖动后稳定在了 0 附近。但观察角度曲线发现，其实独轮机器人并没有恢复到直立状态。独轮机器人似乎停在了某个角度上，而并没有恢复到直立时的角度。

笔者分析了所使用到的控制系统以及仿真曲线的变化后，明白了其中的漏洞。该控制使用到的参数是角速度以及角加速度。进入控制以后，独轮机器人角速度需要进过速度减小到零再反向加速。但在反向加速，也就是独轮机器人回摆的时候。系统错误地判断为独轮机器人向另外一边倾倒了。所以系统又反向控制，直到角速度保持在零附近的位置。可惜的是此时系统并没有意识到独轮机器人处于倾斜状态。系统一直保持微小的输出，却不能够抵消重力的作用。独轮机器人便缓慢的倒下。

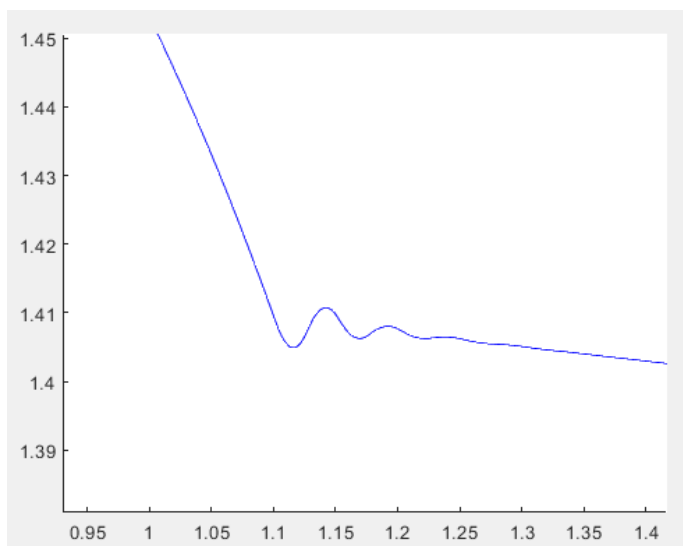


图 4.14 角度曲线的局部放大图

图 4.14 是图 4.13 (a) 中角度曲线的局部放大图。图中蓝色的角度曲线确实有一小段的角度增加，可见独轮机器人确实有一小段的回摆。但可惜的是这并没能使得独轮机器人恢复直立，因为控制系统无法区分回摆和倾倒的区别。

由此可见单纯的 PD 控制，还不能使得独轮机器人恢复直立。试想一下，我们是如何得知独轮机器人是在回摆的过程而不是倾倒的过程的？对，是用眼睛看到的，眼睛看到的信息是什么？是角度，肉眼观测很容易知道回摆和倾倒的区别，一个是在远离竖直方向，一个在靠近竖直方向。

所以控制系统需要引入积分项。积分项得到的数据便是角度。加入角度进行控制，便相当于加入了一双眼睛，能区分回摆与倾倒的区别。

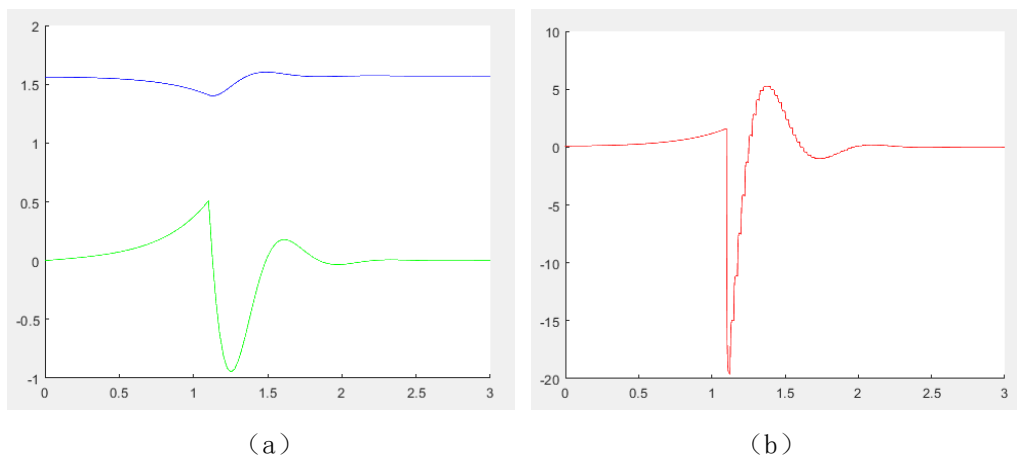


图 4.15 PID 控制仿真曲线

图 4.15 是加入了积分项得到的仿真曲线图。由图中的曲线可知，独轮机器人进入自稳控制后，独轮机器人经过几次抖动，便基本恢复到直立状态。可见该控制系统能够控制独轮机器人的左右倾倒。

4.5 独轮机器人前后倾倒 PID 控制

4.5.1 前后倾倒动力学建模

前后倾倒是由车轮进行控制，在实际的控制中，会有众多的因素对控制造成影响。为简化模型，忽略次要因素，建模前需要进行以下假设：

- 1、假设地面水平，没有倾角和起伏。
- 2、忽略空气阻力与内部阻尼造成的影响。
- 3、假设车轮与地面的摩擦力足够大，工作过程中不发生打滑现象。

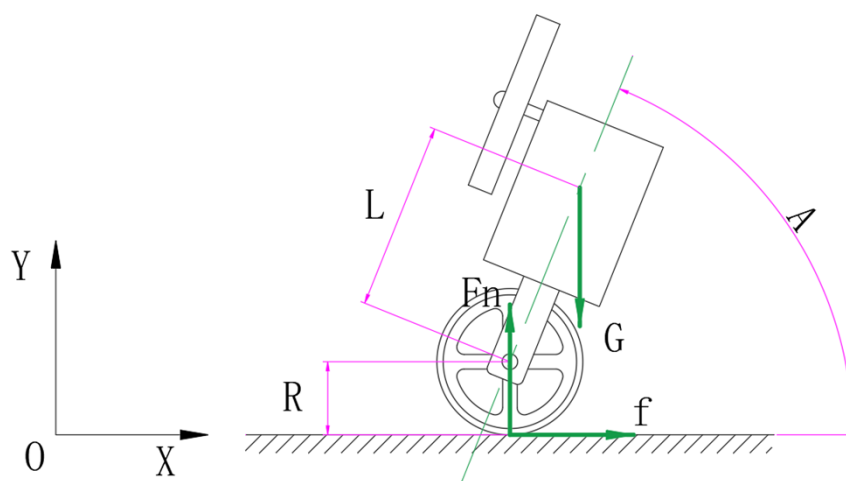


图 4.16 前后倾倒受力分析图

图 4.16 是对独轮机器人发生前后倾倒时的受力分析图。如图所示建立平面直角坐

标系，其中 X 轴与地面共面，Y 轴方向竖直向上。 F_n 是轮机器人受到的支持力， f 是与地面的摩擦力， R 为车轮半径， L 是重心到车轮轴的距离。由于 F_n 和 f 力的作用线都不经过重心，这两个力将会使得独轮机器人受到扭矩而发生旋转。此外， F_n 可以减缓独轮机器人倒下的速度， f 会导致独轮机器人发生水平方向的位移。

由上面的受力情况可知，在前后发生倾倒时，独轮机器人既有旋转也有平移运动。把平移与旋转运动拆分。可得到如下关系：

$$A = A_0 + \int_0^t \int_0^t \frac{M}{J} dt + \omega_0 dt \quad (\text{式 4.11})$$

$$M = f(R + L \sin A) - F_n L \cos A \quad (\text{式 4.12})$$

$$x = x_0 + \int_0^t \int_0^t \frac{f}{m} dt + V_{x0} dt \quad (\text{式 4.13})$$

$$y = R + L \sin A \quad (\text{式 4.14})$$

式 4.11 中， A_0 是独轮机器人初始时刻的角度， M 是受到的总扭矩， J 是绕重心旋转对应的转动惯量， ω_0 是初始时刻的角速度。式 4.13 中， x 是 t 时刻独轮机器人重心位置的 X 轴坐标， x_0 是独轮机器人初始位置的 X 轴坐标， V_{x0} 是独轮机器人初始时刻沿着 X 轴正方向的速度。式 4.14 中， y 是独轮机器人重心位置的 Y 轴坐标。

独轮机器人受到的重力 G 固定不变，大小可知；摩擦力 f 属于静摩擦由电机输出产生，大小可知；支持力 F_n 是因与地面接触而产生，大小未知。我们知道，在独轮机器人工作的过程中，车轮轴不会发生上下变化，始终与地面保持距离 R 。故在此点的 Y 轴方向上有：位移为零，速度为零，加速度为零。

由加速度为零可得到以下等式：

$$\frac{F_n - G}{m} - \frac{M}{J} L \cos A = 0 \quad (\text{式 4.15})$$

等式左边由两部分组成，前者是由平移运动而产生的加速度，后者是由旋转运动而产生的加速度再分解到 Y 轴上的加速度。结合式 4.12 可以得到支持力 F_n 的表达式：

$$F_n = \frac{GJ + mf(R + L \sin A)L \cos A}{J + mL^2 \cos^2 A} \quad (\text{式 4.16})$$

由式 4.11、式 4.12、式 4.13、式 4.14、式 4.16 可计算出任意时刻，独轮机器人的位置与角度信息。

4.5.2 前后倾倒的控制系统建立与仿真分析

因有左右倾倒控制经验，前后倾倒的控制直接使用比例、积分、微分三项。并使用建立的动力学模型，在 MATLAB 中进行仿真模拟。根据前后倾倒的受力分析可知，独轮机器人除发生旋转运动外，还有平移运动。整体的运动比较复杂，为使控制效果更为直观，本文在仿真中加入了动画，用于实时观察独轮机器人的运动姿态。仿真的具体程序见附录 5。

图 4.17 是由 MATLAB 模拟前后倾倒控制的数据图。图中上半部分是模拟动画，为简化程序独轮机器人已简化成木棒形状，下半部分是实时的角度-时间曲线。图中的独轮机器人初始时角度为 73° ，初始线速度为零，初始角速度为零。

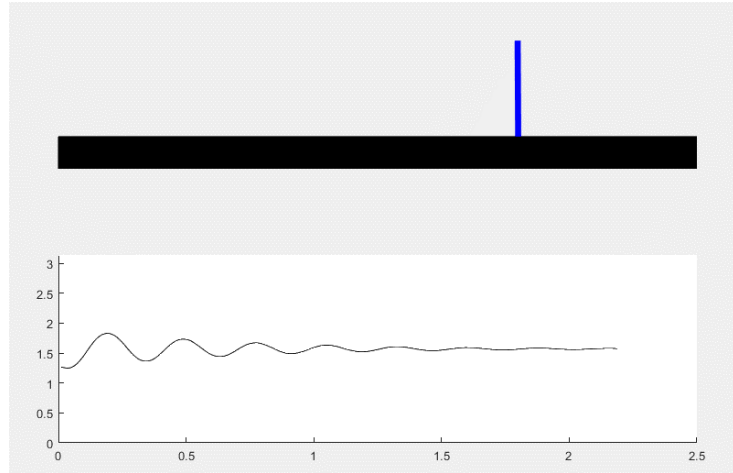


图 4.17 前后倾倒的仿真数据

由图 4.17 角度-时间图像可看出，独轮机器人经过几次摆动后，角度平稳地保持在 1.5 左右的位置，也就是 90° 的位置。模拟动画中的独轮机器人也是与地面处于垂直状态。但角度-时间图像中可以看到，独轮机器人控制中经历的摆动稍多。

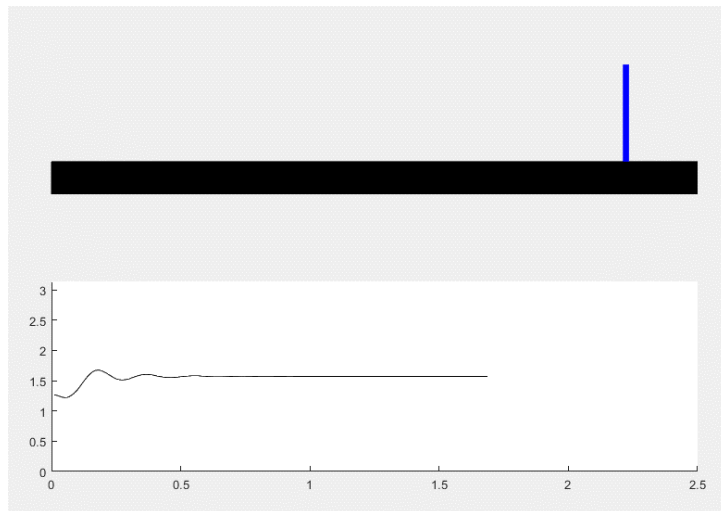


图 4.18 改变了控制参数的仿真数据

图 4.18 是通过改变 PID 的参数得到的新的仿真数据图。通过对 PID 控制系统中的各个系数进行细微的调整，可以使得控制过程的摆动更少，更快地恢复到直立状态。图中可见，独轮机器人的控制系统收敛得很快，经过两三个摆动以后便恢复到直立状态，超调量也降低了一些。由此可见，该控制系统可以对独轮机器人的前后倾倒进行控制。

由此编写 arduino 控制系统程序，详细程序见附录 6。

5 润色功能

为使本文的独轮机器人更具特色，本文对独轮机器人进行了润色。本文将为独轮机器人加入了无线充电装置。

5.1 无线充电介绍与原理分析

随着技术的进步以及需求的增加，现在的智能手机出现了无线充电的功能。具有无线充电功能的产品在市场上相对少见，较新鲜的功能对消费者而言是有一定的吸引力的。笔者相信无线充电的强大能量，也希望某日能实现所有用电设备均不需要拖拽着电线。尽管现今的无线充电尚存在功率较小、传输距离较短等问题。但因无线充电的使用十分方便，产品外观上也能更美观，所以笔者仍然对其抱有很高期望。

无线充电运用的是无线输电技术，是将电能通过高频振荡电路转换成电磁能的形式输送到远方，接收端利用共振原理实现高效电能传输。这看似新兴的技术，实则早在一百多年前就有先驱者成功地实现了无线输电。在 1891 年，美国发明家尼古拉·特斯拉便实现了无线输电，往后的日子里特斯拉先生致力于实现全球范围的无线输电。但由于在当时，无线输电的技术过于超前以及无线输电无利可图，使得特斯拉先生没有办法实现梦想。在此之后也少有人研发利用。无线输电长距离传输时能量衰竭得很严重，使得无线输电的传输距离很短。即使是在今天，许多无线充电的产品都是需要几乎紧贴充电发射端才能进行充电。这可以说是无线输电最严重的问题。

5.2 一代无线充电实物制作

本文所要设计的无线输电设备，是为独轮机器人充电所用，功率较小。中大型特斯拉线圈使用的电路图，运用于小型特斯拉线圈中存在一些不妥的地方。大型特斯拉线圈采用两次升压，初次升压便达到几千伏特，且需要使用到打火器实现 LC 振荡。两次升压是为了大幅度提高放电顶端的电压，提高功率，小功率特斯拉线圈没有这个必要。打火器在打火时将损失掉一部分能量，大功率特斯拉线圈不需要在乎一点点的能量，但对于小功率特斯拉线圈而言是很大的能量浪费，且打火器高压大电流，工作时火光四射，比较不安全^[10]。其次，电路中需要用到共振提高效率，就需要计算放电顶端的对地电容，该计算非常复杂，特斯拉先生是计算固有频率的天才，但笔者不是。所以本文将使用追频特斯拉电路制作无线充电装置。

图 5.1 是一款追频特斯拉线圈电路图。该电路由 11.1V 航模电池供电。三极管 Q 为 NPN 型三极管，型号 882，加散热片。电阻 R 的阻值为 1kΩ。LED 灯为工作指示灯，电路正常工作时指示灯亮起。次级线圈 L2 使用 0.06mm 线径漆包线，在直径 32mm 的 PVC 水管上单层紧密绕制了 178.3mm，约三千圈。初级线圈使用 1.0mm 线径漆包线，在直径 32mm 的 PVC 水管上绕制三圈，让其自然松开，再微微调整大小。初级线圈与次级线圈的绕行方向需要相反。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要
下载或阅读全文，请访问：

<https://d.book118.com/10534120222011131>