

中国工程机器人大赛暨国际公开赛（RoboWork）

工程创新项目工程创新设计赛  
四足仿生爬坡项目  
技术报告

参赛学校： 西北农林科技大学

队伍名称： XRobot221

参赛队员： 王荐生 王运 唐志辉

带队教师： 王少坤（13572844951）

填写日期： 2022 年 10 月

## 2022 中国工程机器人大赛暨国际公开赛诚信比赛承诺书

本参赛队（队伍编号<sup>2022</sup>03080211246）是 2022 中国工程机器人大赛暨国际公开赛参赛者。本人已认真阅读工程机器人大赛官网发布的相关比赛信息，阅读并理解中国工程机器人大赛暨国际公开赛 2022 年关于比赛的相关规定，并郑重做出如下承诺：

1. 本参赛队承诺签名为本参赛队指导教师和队员本人真实姓名，承诺书一经签署就意味着承诺人接受并承担本承诺书的全部责任和义务。
2. 本参赛队承诺参赛所用机器人、场地以及其他比赛相关工具/设备均真实、有效，符合赛项规则要求，符合其他参赛相关要求。提交的所有参赛材料真实、有效。
3. 本参赛队自觉服从中国工程机器人大赛暨国际公开赛的统一安排，接受组委会的监督和检查。
4. 本参赛队自觉遵守相关法律和比赛纪律、比赛规则，诚信比赛，不违规、不作弊。
5. 本参赛队承诺整个比赛过程严格按照赛事规则进行，按照要求提交材料，所提交参赛材料均真实有效且符合参赛要求。
6. 本参赛队承诺本队参赛机器人属于本参赛队所有，同一机器人没有重复参加比赛。
7. 本参赛队保证所用比赛场地规范、计时计分准确、视频拍摄真实。
8. 本参赛队承诺正式比赛视频在录制意外中断时则需进行重新录制，不会对正式比赛视频中的比赛过程内容进行剪辑。
9. 大赛技术委员会和裁判有权对参赛队伍参赛所使用的机器人、场地以及其他相关比赛用物品的真实性、有效性提出质疑，参赛队伍有责任和义务回复质疑。对有重大异议的作品，大赛技术委员会和裁判有权要求参赛队伍到现场进行重新比测。
10. 非不可抗因素，需要参加现场比赛的队伍要按照要求按时参加现场比赛。

若本参赛队违背上述各项承诺，自愿承受因此产生的后果。指导教师对所有真实性负全责，各高校盖章前要检查真实性。

承诺人签名：

指导教师 1 姓名(手签) 王健 身份证号 \_\_\_\_\_ 联系方式 135784951

指导教师 2 姓名(手签) \_\_\_\_\_ 身份证号 \_\_\_\_\_ 联系方式 \_\_\_\_\_

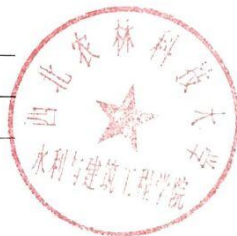
队员 1 姓名(手签) 王生 身份证号 410103200205110017

队员 2 姓名(手签) 唐志辉 身份证号 431121200401132310

队员 3 姓名(手签) 王远 身份证号 142201200209262199

所在单位公章：

2022 年 10 月 9 日



## 关于技术报告使用授权的说明

本人完全了解 2022 中国工程机器人大赛暨国际公开赛（RoboWork）关于保留、使用技术报告和研究论文的规定，即：参赛作品著作权归参赛者本人和比赛组委会共同所有，比赛组委会可以在相关主页上收录并公开参赛作品的设计方案、技术报告以及参赛机器人的视频、图像资料，并将相关内容编纂收录在组委会出版论文集中。

参赛队员签名：王生 唐志辉 王运

带队教师签名：王生

日期：2022年10月9日



## 目录

0 引言/综述 .....	7
1 系统整体设计 .....	7
1.1 比赛规则 .....	7
1.2 结构简介 .....	8
1.3 机器人头和脚板简介 .....	9
1.4 机器人驱动元件 .....	9
1.5 研究内容 .....	10
1.6 研究目标 .....	10
2 机械结构设计 .....	11
2.1 物理结构设计要求 .....	11
2.2 材料选择 .....	11
2.3 自由度配置 .....	11
2.4 物理结构设计 .....	11
2.5 脚部结构的设计 .....	12
2.6 腿部结构的设计 .....	12
2.7 躯干部结构设计 .....	12
2.8 物理结构组装 .....	12
3 硬件设计 .....	12
3.1 舵机 .....	12
3.2 连接件的制作 .....	14
3.3 爬坡机器人物理硬件的设计及其功能 .....	14
3.4 ATmega32 单片机最小系统设计 .....	14
3.5 Atmega32 简介 .....	15
3.6 外设特点 .....	15
3.7 特殊的处理器特点 .....	15
3.6 灰度传感器原理 .....	16
3.7 灰度传感器技术参数和调节方法 .....	16
3.8 本章小结 .....	16
4 软件设计 .....	17
4.1 动作设计思想 .....	18
4.2 动作流程函数 .....	19
5 系统开发与调试 .....	24
5.1 步态调试 .....	24
5.2 稳定性分析 .....	24
5.3 本章小结 .....	25
6 结论 .....	25
6.1 总结 .....	25

6.2 进一步的研究工作 .....	26
6.3 致谢 .....	26
7 第七章 附录 .....	26
附录 1.1 研究及推广价值 .....	26
附录 1.2C 语言程序 .....	27

# 四足仿生机器人

王荐生<sup>1</sup>, 王运<sup>2</sup>, 唐志辉<sup>3</sup>

(1.西北农林科技大学, 陕西省 咸阳市 杨陵区 712100; 2.西北农林科技大学, 陕西省 咸阳市 杨陵区 712100; 3.西北农林科技大学, 陕西省 咸阳市 杨陵区 712100)

**摘要:** 此次设计的四足爬坡机器人主要由机械结构和控制系统组成, 我们若把机械结构看做龟的骨架, 那么控制系统就被认为是龟的大脑, 所以机器人的先进程度与功能强弱通常都直接与其控制系统的性能密切相关。机器人控制系统是根据指令及传感信息控制机器人完成一定的动作或作业任务的装置, 它是机器人的心脏, 决定了机器人性能的优劣。我们本次对仿人障碍机器人的制作, 参考了大量的视频资料以及其他资料, 对机器人的四足结构与步态做出了具有突破性与创新性的调整, 使其更好地适应各种地形尤其是在斜坡的行走更加稳定且速度更快。而传感器是机器人的眼睛, 主导着机器人对行走的判断与规划, 为此我们选用了灰度传感器并且不断对其固定位置, 数量, 以及程序进行调整。最终使得机器人具有敏锐的判断力与强大的执行力能够在比赛中发挥着极高的性能。

**关键词:** 四足仿生机器人; 爬坡; 步态规划; 传感器; 控制系统;

## Quadruped Bionic Robot

Wangjiansheng<sup>1</sup>, WangYun<sup>2</sup>, TangZhiHui<sup>3</sup>

(1. Northwest A&F University, Yangling District, Xianyang 712100, Shaanxi, China; 2. Northwest A&F University, Yangling District, Xianyang 712100, Shaanxi, China; 3. Northwest A&F University, Yangling District, Xianyang 712100, Shaanxi, China)

**Abstract:** The quadruped climbing robot designed this time is mainly composed of mechanical structure and control system. If we regard the mechanical structure as the skeleton of the turtle, then the control system is considered as the brain of the turtle. Therefore, the advanced degree and functional strength of the robot are usually directly related to the performance of its control system. Robot control system is based on the command and sensing information control robot to complete a certain action or task device, it is the heart of the robot, determines the robot performance. We have made a breakthrough and innovative adjustment to the quadruped structure and gait of the robot by referring to a large number of videos and other materials in the production of the humanoid obstacle robot, so that it can better adapt to a variety of terrain, especially on the slope walking more stable and faster. The sensor is the robot's eye, which leads the robot's judgment and planning of walking. For this reason, we choose the grayscale sensor and constantly adjust its fixed position, quantity and program. Finally, it makes the robot with keen judgment and strong executive force to play a very high performance in the competition.

**Key words:** Quadruped bionic robot; Climb a hill; Gait planning; Sensor; Control system;

## 0 引言/综述

仿生爬坡机器人是仿生学各种先进技术与机器人领域应用的结合,是一项综合性技术。本文设计的仿生四足爬坡机器人集步态规划、传感器的应用及控制系统设计等于一体,针对仿生四足机器人的运动特点和功能要求,在步态规划中,采用了一套改进后的步伐 以实现机器人在大坡度的爬坡运动,在控制系统的设计中,采用控制系统与视觉控制系统和舵机控制系统密切配合,联合调度。实验证明机器人运行平稳,实现了在坡面的稳定运动。 仿生四足机器人主要是模仿动物的运动机理,实现在不同环境 下的适应性行走,基于恶劣环境下的军事化机器人或服务行业的家居机器人是近年来机器人行业研究的热点问题之一。仿生四足机器人的平稳度优于双足机器人,同时也避免了六足机器人的复杂性和冗余性。与轮式或履带式机器人相比,四足机器人在跨越障碍、适应复杂地形等方面表现出明显的优势。斜坡地形是自然环境中最为常见的地形之一,四足机器人在斜面上运动时,更加难以实现平稳的控制,因此,能否克服这些困难,将决定四足机器人在斜坡上运动时稳定性能的高低。同时,由于目前大功率高端伺服电机产品体积大、重量大,且价格高。而舵机具有体积小、扭矩大、控制方便等特点,因此本文的设计将采用舵机应用在与机器人的头、四肢等关节结构中。在机器人设计中,通常使用 DSP、FPGA、MCU 等对机器人系统进行控制。四足机器人需要在行走时具有一定的稳定性和抗干扰性,同时需要实时采集运动过程中的路况信息等并及时根据环境变化调整姿态以完成控制器发出的任务。而 AVR 系列的 MCU 具有执行结构简单、功耗低、驱动能力强、编程方便等特点,因此选用 AVR 系列的 ATmega328p 为主控制器,并设计了以此为核心的四足机器人控制系统四足机器人的步态规划、运动稳定性以及自我平衡稳定的能力一直是机器人研究领域的重点。

目前较为常见的四足机器人的行走步态有两种,一种是仿照犬类动物的行走步伐前进,另一种则是仿照龟类动物爬行前进,本文所述的机器人行走方式为仿龟类动物。本文设计机器人基于一般乌龟的肢体机构和运动方式加以改进,最终提高了其运动能力,实现了预期的要求。

## 1 系统整体设计

### 1.1 比赛规则

在准备本次机器人制作过程中,首先需要对机器人整体结构进行分析,如何将机器人整体搭建成符合四足仿生赛机器人结构规则的整体结构,并能够使其完成比赛,由此需要我们认真理解并解读比赛规则。

1. 四足仿生机器人必须有明显的头、躯干和四肢等部分,与四足动物的结构比例相协调。要求机器人的头部有 1 个自由度,爬行过程中有明显的比较协调



的摆头和四肢配合动作。

2. 机器人单足是平板脚底板，每个单足尺寸（即机器人单足脚底板和其上安装的辅助器件，共同构成机器人的单足）不大于(长)40mm×(宽)40mm。

3. 机器人的整体尺寸是指机器人所有可能展开尺寸的最大值，机器人的整体尺寸不大于 400 mm。

4. 机器人四足结构是模仿自然界存在的四足爬行动物，如狗、猫等，采用四足爬行方式移动。要求机器人标识出顶面和前进方向，在爬行过程中机器人保持顶面朝上、沿标识的前进方向前进。

5. 在规则允许的条件下，机器人可扩展多种传感器来对机器人的比赛过程进行精确的控制，以求更好的成绩。

6. 参赛机器人可以是参赛队自主设计和手工制作的机器人，也可以是参赛队购买套件组装调试的机器人。

7. 要求机器人安装传感器感知周围环境，禁止不安传感器以“盲跑”方式行进。

8. 要求机器人采用通过微控制器的自动控制方式，禁止比赛进行期间使用任何通讯设备。

9. 要求机器人依靠搭载在机器人本体的电池供电，禁止依靠外部电源供电。

## 1.2 结构简介

本项目的四足仿生机器人采用乌龟的身体结构及步态。为在保证尺寸要求的前提下实现灵活度的最大化，本机器人每条腿采用两个自由度。并在身体左右两侧安放共六个传感器，进行循线和拐弯的识别判断。

由于机械结构上的改进，摩擦力足够大，实现了左前右后或左后右前对角线上两只脚的同时移动而不会滑坡更加接近乌龟真实的步态，同时由于腿部很短在抬腿时不再需要肚子贴地以保持平衡，有效地增大摩擦。

整个控制系统载体是一台功能完整的~~竞赛类~~四足爬坡机器人。机器人的构架实物如图 1-1 所示。爬坡机器人主要由控制板、灰度传感器、四足、头和电源组成。

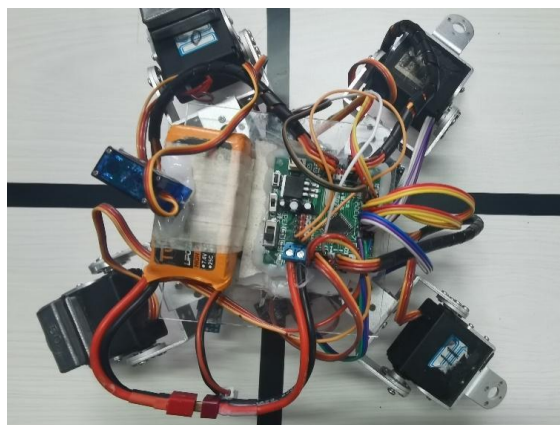


图 1-1 机器人构架实物图

在机器人各个机构零部件的制作以及机器人机构整体装配过程中要注意如



下几点：

（1）在装配机器人的各个部件以及固定螺丝，特别是舵机与连杆部件相连接时，用力不能过大，否则会造成机器人的连接部件、舵机以及螺丝的损坏。

（2）在机器人试运行，可能会由于程序或者驱动元件的损坏而导致机器人变形，那么不够长要换用长螺丝，必要时可以将架子上的洞打大，再用粗螺丝上。这里建议准备几把弯头螺丝刀，以便后面紧螺丝时不用拆卸舵机。需要断开电源，并对机器人进行修整。

（3）第一遍组装时便要做到完美，要确保螺丝上牢（用垫片）。保证舵机不松动。

（4）组装时应注意舵机的极限位置以及左右的对称。

（5）尽可能在第一遍组装时将舵机位置记住，要保证每一个舵机的自由度满足动作的要求。

（6）在连接导线的过程中，要用电工胶带包好导线接头，避免短路现象发生。

### 1.3 机器人头和脚板简介

根据中国机器人大赛对爬坡机器人脚板的尺寸要求，我们充分利用规定的尺寸设计脚板尺寸。我们设计的机器人单足是平板脚底板，每个单足尺寸（即机器人单足脚底板和其上安装的辅助器件，共同构成机器人的单足）小于（长）40mm×（宽）40mm。首先，我们选择 3mm 厚的有铝板作为原材料，制作脚板，3mm 厚的铝板满足爬坡机器人所需要的硬度和韧性要求，然后加装了相同大小的亚克力板，以及一层海绵和天然橡胶，不仅能够充分贴合地形，极大地提升了摩擦力，还能够使得机器人的动作更加稳定，更加灵活。

### 1.4 机器人驱动元件

舵机是一种位置伺服的驱动器，适用于那些需要角度不断变化并可以保持的控制系统。其工作原理是：控制信号由接收机的通道进入信号调制芯片，获得直流偏置电压。它内部有一个基准电路，产生周期为 20ms，宽度为 1.5ms 的基准信号，将获得的直流偏置电压与电位器的电压比较，获得电压差输出。最后，电压差的正负输出到电机驱动芯片决定电机的正反转。当电机转速一定时，通过级联减速齿轮带动电位器旋转，使得电压差为 0，电机停止转动。舵机 CAD 图及实物图分别如图 1-2、图 1-3 所示。

优点：使用高强度全金属齿轮附加金属筒套设计，使用寿命长，控制芯片采用国产芯片，成本较低，具有重复定位的功能，定位准确，输出力量大以及角速度快的优点。

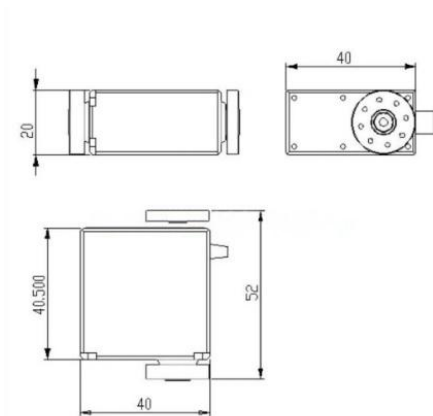


图 1-2 舵机 CAD 图



图 1-3 舵机实物图

## 1.5 研究内容

我们所研究的爬坡机器人，其动作设计和实现主要是采用模仿四足动物乌龟的动作的研究方法，动作具有很强的稳定性，消除了机器人动作的僵硬感。爬坡机器人最大特点在于通过 9 个舵机控制机器人的动作模拟动物的关节构造以模仿动物物体，并最终达到将一个物体移动到另一处的目的。

而把一个动物前进、爬坡、转向等动作转化为机器人的动作，完成动作设计和动作实现，主要研究内容可分解为两步，一是动作程序化——模拟动物的关节动作结构，将要实现的机器人动作表示成舵机位置和间隔时间的程序序列；二是通过程序控制机器人的动作将其表达成我们预想的动作，并加以调试和修改。所有的动作都由程序实现、这些动作的控制数组也存储在 CPU 内。因为整个程序的子程序很多，所以为了保证程序的实时性和表达的准确性，必须保证所有的程序均为非阻塞程序，并且要配合定时中断，合理地进行各程序间的切换。其中舵机的速度控制程序、各程序间隔的长短以及各程序实现速度的选择为算法精华和重点，也是我们研究的重点内容。

机器人的先进程度与功能强弱通常都直接与其控制系统的性能密切相关。机器人控制系统是根据指令及传感信息控制机器人完成一定的动作或作业任务的装置，它是机器人的心脏，决定了机器人性能的优劣。作为机器人的核心部分，很长时间以来国内外的学者提出了诸多的控制算法，也设计出实现某些算法的控制器，但整个技术还处在实验阶段，没有进入市场，特别是一些高级控制方法在实际中效果根本不理想，所以机器人控制系统有待进一步研究。

## 1.6 研究目标

通过查阅相关资料及收集到的历年比赛信息，我们确定本组的研究目标是严格按照比赛要求，根据比赛流程及规定设计一个爬坡机器人，它不仅能准确完成各个要求的编排动作，还可以最大限度地表达竞速的比赛精神。

我们设计的机器人主要由机械结构和控制系统组成。如果认为机械结构是人的骨架，那么控制系统就被认为是人的大脑，所以机器人的先进程度与功能强弱

通常都直接与其控制系统的性能密切相关。机器人控制系统是根据载入控制板的程序，发出指令来控制机器人完成所规定的动作，它是机器人的心脏，决定了机器人性能的优劣。作为机器人的核心部分，机器人控制系统是影响机器人性能的关键部分之一，它从一定程度上影响着机器人的运行和发展。

机器人技术的实现不仅取决于其机械机构的优化，在很大程度上还依赖于一个性能优良控制系统；而一个工作稳定、精度可靠的、实时性强的控制系统不仅需要高级的控制算法，还需要实现控制算法的手段。

为了设计并制作能够完成比赛动作的机器人。我们首先提前对其动作进行了构思，然后设计较为合理的关节自由度以及舵机位置。其次是考虑各个连接部件如舵机，电池和控制板的尺寸以及电池和电路盒的尺寸等。因此，在充分了解这些信息后，我们编排了具体的程序，并在不断的实践和试验中进行修正和改进。基于对爬坡机器人的基本结构、动作要领和比赛规则的了解，我们的研究目的及设计思想基本成熟。

## **2 机械结构设计**

### **2.1 物理结构设计要求**

根据比赛要求及对步态的研究，结构应满足以下几点：

- (1) 机器人的重心较低，走路稳定。
- (2) 机械结构尽量简单，符合全国机器人大赛对四足仿生机器人的尺寸要求，并具有一个可以自由和谐摆动的头部。
- (3) 机械结构的强度要满足四足仿生机器人爬坡过程中可能发生的损耗。

### **2.2 材料选择**

根据机械结构的设计要求可知，设计结构所采用的材料需有一定的强度来支撑机器人快速完成动作，而且材料的密度不易太大。考虑到加工的难易程度我们还应选择便于加工和造型的材料。

通过以上分析，我们选择常见的3mm硬质塑料板和2mm硬质铝板作为机械结构的材料。硬质塑料板容易切割，板面平整且不易变形，可以用作四足仿生机器人的脚板及身体。硬质铝板具有重量轻、硬度高、弹性模量大等特点，可以用来制作机器人的连接件。

### **2.3 自由度配置**

由于大赛要求四足仿生机器人尺寸较小，我们就要考虑用尽量少的舵机来实现尽量多的动作，因此我们的仿生四足机器人腿部自由度选择配置共8个舵机来实现比赛所需动作。

因为四足仿生机器人在比赛中只需完成直线行走、转弯这两个简单动作，不需要其他复杂的动作，所以我们讨论研究决定，在每条腿上分别配置水平方向转动的舵机和竖直方向上迈腿的舵机共两个舵机。

### **2.4 物理结构设计**

为便于机器人各部分的设计安装，我们把四足仿生机器人的物理结构分为三个部分：躯干、腿部、脚部。利用CAD制图将各部分零件绘制，并将其加工成型，最后整体组装起来。

## 2.5 脚部结构的设计

为了满足比赛规则，而且使机器人脚部触地面积最大化，我们研究决定四足仿生机器人的单脚规格为 $40\text{mm} \times 40\text{mm}$ ，符合大赛规定的单脚最大尺寸。为使脚板较平整，我们选用厚度 $3\text{mm}$ 的透明塑料板制作脚板。脚底板为防滑采用天然橡胶，防滑效果较好。同时在胶皮和硬质脚底板之间增加一层海绵使脚底更软和坡面贴合更紧。同时，脚底板存在特定角度，放腿时与斜面紧密贴合使摩擦力最大化，抬腿时脚底即使与斜面接触也形成一定角度实现摩擦力最小化；同时，由于脚板的脚后跟短脚前掌长，在抬腿过程中前腿与地面不发生摩擦，后腿与地面发生摩擦，提供了向前的支撑。

## 2.6 腿部结构的设计

腿部结构的设计在整个机器人的结构设计中至关重要，它不仅是连接脚部和躯干部分的总要枢纽，而且是实施步行的重要部分，保证其稳固使保障运动稳定性的重要组成部分。因此我们选用厚度为 $2\text{mm}$ 的铝板制作连接件。同时为减小机器人的尺寸，我们将腿部控制左右偏移的舵机横向集中安装在一起组成躯干的一部分。并在前两只脚上加装重物，使整体重心更靠前，有效地使整体摩擦力更大。连接件。

## 2.7 躯干部结构设计

躯干部的结构是四足仿生机器人中功能最多的一部分。首先用来连接四条腿，保证四条腿相对于躯干中心的对称；其次它是四足机器人控制板的安装平台；它的结构还将决定四足机器人完成向前、转弯的稳定程度。因此我们决定将控制板安装在躯干部的上部中央，伺服舵机安装在躯干部分四周。躯干部分整体采用四方体的形状，以上下两个亚克力板固定四条腿的横向舵机，将控制板和电池安装在上方亚克力板上。其制作材料我们仍选用 $3\text{mm}$ 厚的亚克力板。

## 2.8 物理结构组装

四足仿生机器人的装配相对来说比较简单，但是也有一定的组装次序，只有选对安装的次序才不会致使某些部件的安装被其他部件所影响，根据我们的安装经验，总的安装首先从脚板开始，依次将四个脚板上的连接件、舵机安装，其次依次向上将腿部的所有结构安装完成后，最后将四足机器人的四条腿固定在一起组成躯干部分。至此，仿生四足机器人的安装已经完成。

# 3 硬件设计

## 3.1 舵机

舵机的机械参数：

1、尺寸： $40\text{mm} \times 20\text{mm} \times 40.5\text{mm}$

- 2、重量：60 克
- 3、技术参数无负载速度 0.14 秒/60 度(4.8V)，0.13 秒/60 度(6.0V)
- 4、堵转扭矩 14.2kg/cm(4.8V)堵转电流 1.5A，16.3kg/cm(6.0V)
- 5、使用温度：0 到 55 摄氏度
- 6、RDS3115 的死区设定:3 微秒
- 7、工作电压：4.8v-6v
- 8、结构要点：贵金属电刷电机
- 9、高纯高强度耐磨金属齿轮结构
- 10、双滚珠轴承
- 11、连接线长度 32 厘米
- 12、RDS3115 最大的特点是可正反 90 度正切安装，多种安装方式，结构方便简洁。

舵机的物理结构及其工作原理：

伺服舵机物理结构主要由舵盘、控制电路板和减速齿轮组。此外，还有位置反馈电位计以及直流电机等部件。详见图 3-1。



图 3-1 舵机原理示意图

所用舵机型号及相关说明：舵机使用 RDS3115/3120 系列机器人专用舵机。

机器人专用舵机 RDS3115/3120 系列秉承了模型用舵机精准、轻巧的特点，同时充分考虑到机器人舵机兼顾耐用、使用方便的需要，在内部结构设计和驱动电路设计上进行了全面优化，特点如下：

（1）RDS3115/3120 双轴数字舵机，是一款为机器人专门设计的数字舵机，运行噪音低、平稳且线性度高。可控角度范围 180 度/270 度，断电可 360 度旋转，特别适合机器人的各关节活动。

（2）该款数字舵机内部伺服控制板采用单片机 MCU 控制，这是相对于传统的模拟舵机的主要区别，但是控制方式是相同的，都是 PWM 脉宽型调节角度，周期 20ms，占空比 0.5ms~2.5ms 的脉宽电平对应舵机 0 度~180 度角度范围，且成线性关系。

（3）与模拟舵机的最大区别是带位置锁定功能，给一次 PWM 脉宽，舵机输出角度可锁定，直到下次给不同的角度脉宽或者断电才可以改变角度(模拟舵

机则不具有该功能, 模拟舵机要锁定位置需要不停的提供 pwm 信号)。另外控制精度高、线性度好, 与控制协议严格一致, 输出角度准确且响应速度快是数字舵机品质好的重要原因。

(4) 通常舵机控制器采用 500~2500 数值对应舵机控制输出角度的占空比 0.5ms~2.5ms 的范围, 这样该款数字舵机的控制精度理论值可达到 0.09 度, 而实际由于舵机齿轮组间的间隙存在, 该款舵机的最小控制精度能达到 0.9 度以下, 相对于上位机软件滑竿控制量的最小调节单元值为 10, 舵机才有明显的转动反应。

### 3.2 连接件的制作

该类型连接件(图 3-2)由质轻强度高的 6067 铝合金制成, 用于连接舵机。在连接件底部有多个钻孔可供选择, 可以实现舵机与舵机间的多种连接方式。该连接件是制作该机器人的主要连接件。



图 3-2 舵机连接件实物图

图纸: 根据结构设计需求, 使用 CAD 制作各连接件图纸。

选材: 选用在建材市场买到的用于制作门窗的铝合金型材, 切割成铝条, 然后根据 CAD 图纸要求制作。

制作流程: 取铝板→划线→加工→打磨→折弯→钻孔等工艺, 做成 U 型连接件。

### 3.3 爬坡机器人物理硬件的设计及其功能

采用一体化连接, 有效利用了伺服电机的尺寸。自由度尽量符合各关节的活动区域。对于脚部构件, 选用具有高硬度、高冲击性能及抗老化、耐磨、耐腐蚀的优良性能的有机塑料。舵机倾斜装配, 最大限度的利用舵机旋转角度。动作更复杂轻而质硬的铝合金材料是金属的高强度使其成为最理想的连接机器人的材料。采用硬铝合金材料, 加工成为机器人的连接部件。

### 3.4 ATmega32 单片机最小系统设计

伺服舵机的控制信号本质上是一系列不同强度不同脉宽脉冲, 因而能够很方便地利用脉冲发生装置实现对伺服舵机的控制。理论上只要能产生脉冲信号的设备都具有控制舵机的能力, 比如脉冲发生器、PLC、以及 AVR 单片机等。AVR(ATmega32)内部装置有 2 个 16 位计数器, 利用它可以产生周期为 20 ms 的脉冲信号, 根据实际需要, 我们可以很容易的改变输出脉冲的宽度。最小系统原理图如图 4-2 所示, 其中 PA0-PA7, PB0-PB7, PC0-PC7 这 24 个接口控制 24 个舵机或



传感器；PD2-PD5 这 4 个接口控制 4 个传感器；PD0, PD1 这两个接口控制分别控制信号输入与输出；XTAL1, XTAL2 产生时钟周期。

### 3.5 Atmega32 简介

本次设计的机器人控制器，采用的是具有 32KB 系统内可编程 Flash 的 8 位 AVR 微控制器 ATmega32。

Atmega32 参数介绍如下：具有 8 位 AVR 微处理器，这种微处理器高性能、低功耗，具有 32 个 8 位通用工作寄存器，131 条指令，其中的大部分指令的执行时间是单个的时钟周期，拥有只需两个时钟周期的硬件乘法器以及非易失性程序和数据存储器，内存为 32KB 的系统内可编程 Flash 擦写寿命：10,000 次并且拥有 1024 字节的 EEPROM 擦写寿命：100,000 次和 2K 字节片内 SRAM。

### 3.6 外设特点

- 两个具有独立预分频器和比较器功能的 8 位定时器/计数器
- 一个具有预分频器、比较功能和捕捉功能的 16 位定时器/计数器
- 具有独立振荡器的实时计数器 RTC
- 四通道 PWM
- 8 路 10 位 ADC
- 8 个单端通道 T

QFP 封装的 7 个差分通道

2 个具有可编程增益（1x, 10x, 或 200x）的差分通道

- 面向字节的两线接口
- 可编程的串行 USART
- 可工作于主机/从机模式的 SPI 串行接口
- 具有独立片内振荡器的可编程看门狗定时器
- 片内模拟比较器

### 3.7 特殊的处理器特点

- 上电复位以及可编程的掉电检测
- 片内经过标定的 RC 振荡器
- 片内/片外中断源
- 6 种睡眠模式：空闲模式、ADC 噪声抑制模式、省电模式、掉电模式、Standby 模式以及扩展的 Standby 模式

• I/O 和封装

- 32 个可编程的 I/O 口
- 40 引脚 PDIP 封装, 44 引脚 TQFP 封装, 与 44 引脚 MLF 封装

工作电压

- ATmega32L: 2.7-5.5V
- ATmega32: 4.5-5.5V

速度等级



- ATmega32L: 0-8MHz
  - ATmega32: 0-16MHz
- ATmega32L 在 1MHz, 3V, 25° C 时的功耗
- 正常模式: 1.1mA
  - 空闲模式: 0.35mA
  - 掉电模式: <1μA

### 3.6 灰度传感器原理

灰度传感器是模拟传感器，有一只发光二极管和一只光敏电阻，安装在同一面上。灰度传感器利用不同颜色的检测面对光的反射程度不同，光敏电阻对不同检测面返回的光其阻值也不同的原理进行颜色深浅检测。在有效的检测距离内，发光二极管发出白光，照射在检测面上，检测面反射部分光线，光敏电阻检测此光线的强度并将其转换为机器人可以识别的信号。

### 3.7 灰度传感器技术参数和调节方法

- 1.主体外形尺寸: 10×16mm (高×外直径)
- 2.重量: 10g 额定电压: 直流电源 5.0V
- 3.线长: 18.4cm±0.2cm (有效距离)
- 4.检测范围(反射面为白纸): 0.3~0.8cm (检测面为白纸时检测高度在 0.5cm 时为)
- 5.调节方式: 多圈电阻式调节, 逆时针方向旋转功率变大, 顺时针方向旋转功率变小
- 6.返回值: 13~190 (检测面为白色检测高度为 0.5cm; 高度和检测面都一样的情况下颜色越浅, 返回值越小)
- 7.安装方式: 单颗 3 螺丝安装
- 8.连接方式: 单条 3 芯排线, 2510 型 3 脚插头

灰度传感器上无信号指示灯, 但是配有检测颜色返回模拟量大小调节器。欲使检测给定的颜色时, 可以将发射/接收头置于给定颜色处, 配合调节器即可调出合适的返回模拟量。方法如下:

将调节器逆时针方向旋转, 返回模拟量变大; 将调节器顺时针方向旋转, 返回模拟量变小; 可以一直调节直到你需要的数值为止。若需要准确的模拟量, 可以用程序在液晶屏幕上显示, 配合调节器即可调出准确的模拟量。

注: 用螺丝刀旋转调节器时, 不要旋的太快, 也不要旋的太用力, 以防旋坏, 在发现旋不动时, 应马上停止。

### 3.8 本章小结

本章节详细介绍了机器人物理硬件的设计过程及思路, 并列出了各物理硬件的参数及其物理功能。并指出了实际运行中的不足之处和改进方法。

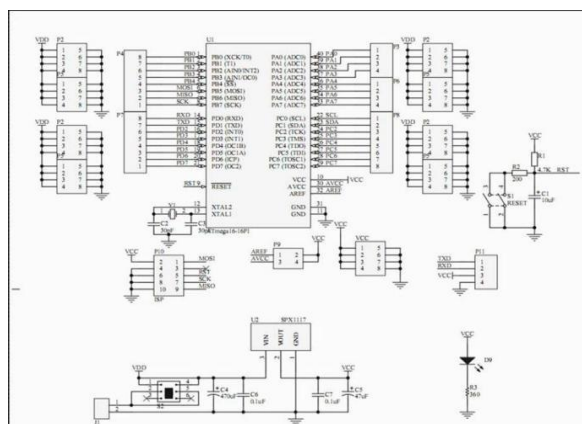


图 3-8 控制板电路原理图

## 4 软件设计



图 4-1 16 路舵机控制板调试软件界面图

在指导老师王少坤的帮助下，我们通过上网查阅资料，参考了相关的动作调试软件，利用 Visual Basic 语言编写 16+路舵机控制板调试软件,采用 USART 串口通信方式，可以在线调试，人机界面友好。安装转串口驱动以后，选择 COM 接口与机器人进行连接。1 到 16 是对机器人的 16 个舵机进行控制，每点击一次使舵机产生 5 个角度的周向位移。对机器人的初始位置进行设置，设置之后连接电源，机器人的形态即为初始位置。右侧为代码区，每个动作的调试完之后都可以生成代码。将生成的代码组合为一个动作模块，每个动作模块的连接用程序框图的格式简单明了的表示出来，从而实现主函数对每个动作模块的调用。该软件界面的可视化以及人机交互性很强。并且配有与下载和编译配合使用的软件，使

用起来相对比较简单方便。16 路舵机控制板调试软件界面如图 4-1 所示。

嵌入式开发中有四种设计开发：硬件层、底层驱动层、操作系统层、应用系统层。硬件层在最底层，驱动层在硬件层之上并且和硬件层紧密相连，操作系统通过驱动层来控制硬件层，应用层运行在操作系统之上。在运动控制器的嵌入式开发中，只有硬件层、驱动层和应用层，没有操作系统层，应用层通过驱动层来控制硬件层。驱动层驱动运动控制器的各个接口，主要包括：无线遥控模块驱动程序、串口通信驱动程序、CAN 总线驱动程序、加速度计模块驱动程序、陀螺仪模块驱动程序和 21 路舵机驱动程序。应用层通过驱动四足龟形机器人平台的总体方案设计层来完成感知传感器的数据实时采集和机器人的各种步态的实时控制。主控机选择的是市场上成熟的控制板，硬件层、驱动层和操作系统层已经做好，本文主要对应用层进行设计。主控机的应用层主要工作是对接收运动控制器上传的感知传感器数据进行处理和判定机器人的姿态，根据机器人不同的姿态下达给运动控制器不同的控制命令。

#### **4.1 动作设计思想**

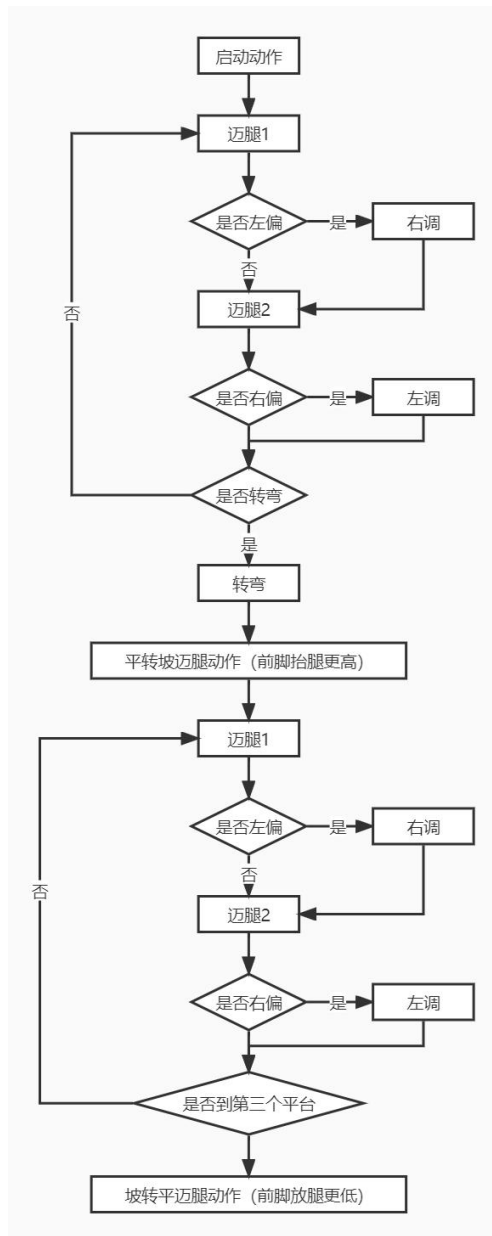


图 4-2 动作函数流程图

本项目的四足仿生机器人将通过三个角度依次增大的斜坡及三个平台最终通过最高处的终点线。通过每次迈步进行一次传感器识别并调整动作状态实现斜坡上的循线直线前进，并通过两个较为突出的灰度传感器实现转弯黑线的识别，并进行转弯动作。为克服平台向第三个斜坡过度时由于斜坡斜度过大可能造成的机器人稳定性显著降低跌落赛道的情况，我们专门设计了由平台向斜坡过渡的动作，以实现平稳上坡。在完全进入第三个坡后，同样由于斜度过大，造成机器人脚掌摩擦力降低，引起巡线动作的稳定性降低。考虑到已不需要转弯识别的传感器，所以我们将所有传感器均用于巡线以避免漏判。最后在到达第三个平台的转换处，由于坡面和平面夹角过大，而机器人腿部较短腹部易卡在转换处，我们将此处设计成专门的坡转平的动作，使前脚放的更低，将机器人腹部顶起越过障碍。

## 4.2 动作流程函数

```

#include <avr\jhl.h>

//子函数列表:

//动作库列表:
void act_103(void); //左转
void act_104(void); //左转循环
void act_101(void); //启动
void act_102(void); //迈腿 1
void act_105(void); //迈腿 2
void act_106(void); //右调 1
void act_107(void); //右调 2
void act_108(void); //左调 1
void act_109(void); //左调 2
void act_110(void); //坡转平 1
void act_111(void); //坡转平 2
void act_112(void); //平转破 1
void act_113(void); //平转破 2
void act_117(void); //右转 1
void act_118(void); //右转循环 1

unsigned char turn_1=0;
unsigned char turn_2=0;
unsigned char line_1=0;
unsigned char line_2=0;
unsigned char i=0;
unsigned char flag=0;
unsigned char line_1l=0;
unsigned char line_2l=0;
unsigned char turn_1l=0;
unsigned char turn_2l=0;
int main(void)
{
SysInit(); //系统初始化
Initfuwei(); //伺服马达端口数据初始化
setfuwei(); //伺服马达启动
setfuwei(); //伺服马达启动
speed=3; //伺服马达速度-3;

```

```

act_101(); //启动;
for(i=0;i<28;i++) //多次循环: 循环变量:i 循环次数:28
{
act_102(); //迈腿 1;
line_11 = GetDS(3); //数字输入: 将通道 3 值赋予变量 line_11;
line_1 = GetDS(5); //数字输入: 将通道 5 值赋予变量 line_1;
if(line_1==1||line_11==1) //条件判断: line_1==1||line_11==1
{
act_106(); //右调 1;
act_107(); //右调 2;
}
act_105(); //迈腿 2;
line_21 = GetDS(4); //数字输入: 将通道 4 值赋予变量 line_21;
line_2 = GetDS(2); //数字输入: 将通道 2 值赋予变量 line_2;
if(line_2==1||line_21==1) //条件判断: line_2==1||line_21==1
{
act_108(); //左调 1;
act_109(); //左调 2;
}
}
speed=2; //伺服马达速度-2;
while(1) //永久循环
{
act_102(); //迈腿 1;
line_11 = GetDS(3); //数字输入: 将通道 3 值赋予变量 line_11;
line_1 = GetDS(5); //数字输入: 将通道 5 值赋予变量 line_1;
if(line_1==1||line_11==1) //条件判断: line_1==1||line_11==1
{
act_106(); //右调 1;
act_107(); //右调 2;
}
turn_1 = GetDS(7); //数字输入: 将通道 7 值赋予变量 turn_1;
if(turn_1==1) //条件判断: turn_1==1
{
act_103(); //左转;
for(i=0;i<3;i++) //多次循环: 循环变量:i 循环次数:3
{
act_104(); //左转循环;

```

```

}
break;
}
turn_2 = GetDS(0); //数字输入：将通道 0 值赋予变量 turn_2;
if(turn_2==1) //条件判断：turn_2==1
{
act_117(); //右转 1;
for(i=0; i<3; i++) //多次循环：循环变量:i 循环次数:3
{
act_118(); //右转循环 1;
}
break;
}
act_105(); //迈腿 2;
line_21 = GetDS(4); //数字输入：将通道 4 值赋予变量 line_21;
line_2 = GetDS(2); //数字输入：将通道 2 值赋予变量 line_2;
if(line_2==1 || line_21==1) //条件判断：line_2==1 || line_21==1
{
act_108(); //左调 1;
act_109(); //左调 2;
}
turn_1 = GetDS(7); //数字输入：将通道 7 值赋予变量 turn_1;
if(turn_1==1) //条件判断：turn_1==1
{
act_103(); //左转;
for(i=0; i<3; i++) //多次循环：循环变量:i 循环次数:3
{
act_104(); //左转循环;
}
break;
}
turn_2 = GetDS(0); //数字输入：将通道 0 值赋予变量 turn_2;
if(turn_2==1) //条件判断：turn_2==1
{
act_117(); //右转 1;
for(i=0; i<3; i++) //多次循环：循环变量:i 循环次数:3
{
act_118(); //右转循环 1;

```



```

}
break;
}
}
speed=2;//伺服马达速度-2;
for(i=0;i<4;i++)//多次循环：循环变量:i 循环次数:4
{
act_112();//平转破 1;
act_113();//平转破 2;
}
for(i=0;i<14;i++)//多次循环：循环变量:i 循环次数:14
{
act_102();//迈腿 1;
line_1 = GetDS(3);//数字输入：将通道 3 值赋予变量 line_1;
line_11 = GetDS(5);//数字输入：将通道 5 值赋予变量 line_11;
turn_1 = GetDS(0);//数字输入：将通道 0 值赋予变量 turn_1;
if(line_1==1||line_11==1||turn_1==1)//    条    件    判    断    :
line_1==1||line_11==1||turn_1==1
{
act_106();//右调 1;
act_107();//右调 2;
}
act_105();//迈腿 2;
line_2 = GetDS(4);//数字输入：将通道 4 值赋予变量 line_2;
line_21 = GetDS(2);//数字输入：将通道 2 值赋予变量 line_21;
turn_2 = GetDS(7);//数字输入：将通道 7 值赋予变量 turn_2;
if(turn_2==1||line_2==1||line_21==1)//    条    件    判    断    :
turn_2==1||line_2==1||line_21==1
{
act_108();//左调 1;
act_109();//左调 2;
}
}
while(1)//永久循环
{
act_110();//坡转平 1;
act_111();//坡转平 2;
}

```

}

## 5 系统开发与调试

### 5.1 步态调试

四足爬坡机器人共有 9 个自由度，每个下肢 2 个自由度。所以，基本思路是让机器人仿照龟的行走方式行走在规划步态时，要考虑每步大小，并且在保证一定速度下，如何起步、双脚支撑时如何摆脚及摆动幅度、如何将双足支撑状态转换到四足支撑状态、如何转换到另一对双足支撑状态等几个基本动作问题，在解决上述几个问题的同时主要综合力学知识来分析如何使机器人行走过程中摆脚时能减少摆动使得机器人行走更加稳定。

### 5.2 稳定性分析

在爬坡机器人的步行过程中，爬坡机器人的脚和地面只有很小的接触面积，因而爬坡机器人较易摔倒，可见爬坡机器人并不是一个稳定系统，因而对爬坡机器人的稳定性分析与合理控制是爬坡机器人研究中的非常重要的内容。

所以只要爬坡机器人不摔倒，爬坡机器人的动力系统就能够使交叉机器人完成预设的动作和功能。主要靠自身的动力学特性完成爬坡机器人的各项功能，所以普遍利用爬坡机器人运动的周期性以及其收敛性校验其稳定性。在机器人动作的调试过程中，发现了一些影响运动稳定性的因素，主要有：机械误差、舵机误差、机器人质量分布、电池电压稳定性、运动参数等因素。

#### (1)机械误差

本课题选用的机器人机械结构在使用一段时间后，关节之间的机械间隙明显增大。在机器人动作调试过程中，在得到机器人各个关节的理论值后，需要对各个关节的目标位置进行补偿。

#### (2)舵机误差

由于舵机中检测输出轴位置的为电阻式电位器，此种电阻器当使用一段时间后，精度明显下降，导致误检测，从而使舵机来回转到，产生舵机不断抖动的现象。舵机误差由于是舵机使用寿命引起的误差，可以采取不断更换电位器或新的舵机来提高舵机的精度。另外，还可以通过实验的方法测出舵机的误差，进行补偿。

#### (3)机器人质量分布

把机器人的所有组件全部嵌入到机器人体内后，机器人整体重量的 85% 集中在机器人躯干部分，这样在行走时对腿部各个关节的冲击就比较大，对舵机输出轴的精度有很大影响。另外，在机器人上坡的过程中，由于上身重量过大，超出机关节舵机的载荷，从而使运动控制器由于电压过度下降而复位。为减轻机器人上位重量，可以考虑结构的重新设计；机器人主控板由于选用的是通用开发板，有很多无用的功能，如果自己开发的话，也可以减轻一些重量。

#### (4)电池电压稳定性

电池在使用一段时间后,电量会减少,输出电压和电量足时相比会有所降低。本课题使用的电池为锂电池,其在电量足时,输出电压为 7.8V,而电量不足时输出电压仅为 5.0V 左右,电压差高达 2.8V 左右。舵机在输出扭矩是受电压影响的,电压高时输出扭矩就大,反之输出扭矩减小。当电量不足时,由于机器人上身重量大,舵机容易超载,而使运动控制器复位。为了避免这种现象,解决的办法是控制系统电源和舵机电源分别供电,并且给舵机电源加稳压限压模块。

### (5)运动参数

本课题主要时对机器人平台的研究,对机器人步态的研究相对来说少些,选择的行走模型和方法比较简单,对舵机的控制采用分段直线驱动法,控制方法比较简单,在运动中也会影响机器人的稳定性。需要采用更好的模型和控制算法,来减小由于运动参数选择不够好对机器人稳定性造成的影响。

## 5.3 本章小结

本章首先通过两个小节详细地描述了我们三个人 5 个月备赛以来,多次对步态进行规划调整:从爬坡机器人第一次步态规划与调试后不能完成比赛既定动作,到第二次的步态规划与调试,可以实现 2 分 40 秒的突破,然后到第 3 次着重考虑了爬坡机器人重心步态规划与调试实现了 1 分 50 秒的成绩。

# 6 结论

## 6.1 总结

本报告侧重对爬坡机器人的行走方式,即步态进行了规划。主要工作内容如下所述:

(1) 多次对爬坡机器人的步态进行了规划与调试,并且对爬坡机器人的每一次步态规划都进行了实际检验。在步态规划与调试期间,通过对每次检验过程中出现的问题思考,找到原因后,对机器人进行了力学分析,并不断对步态进行了调整,最终爬坡机器人的步态达到了我们所期望的要求。

(2) 在爬坡机器人的运动进行稳定性的分析基础上,完成了机器人机械结构的设计,并列出了各物理硬件的参数及其物理功能。并指出了实际运行中的不足之处和改进方法

(3) 基于 Atmega32 完成了爬坡机器人控制器设计,在 Atmega 单片机最小系统的基础上添加了一个电源控制模块。该模块可以输出两种不同的电压。7.4V 的电压供给舵机稳定的驱动电压,同时又给控制板提供 5V 的控制电压,实现了单电源双供电的功能,节约了物理硬件资源,减轻了机器人的重量以及驱动舵机的承重,从而使机器人拥有更加良好的物理性能,更加稳定的实现稳定步行以及其他功能。

(4) 设计控制器,让机器人能快速的实现期望的运动,并且对不在收敛区域内(不是渐近稳定)的运动步态进行调整,使机器人的步态能周期性收敛,从而趋于稳定。

(5) 采用实际的物理样机进行实验, 验证步态与控制器的效果, 实现爬坡机器人的稳定行走。

## 6.2 进一步的研究工作

爬坡机器人的研究是一个很广泛的课题。本报告侧重对爬坡机器人进行了步态规划与调试, 并对爬坡机器人的运动进行了稳定性的分析。虽多次规划了步态并且实际检验了爬坡机器人的步态实现的稳定性, 但是因为在调试过程中对机器人的运动都是手动调试完成, 也因为调整的参数不精确, 导致机器人在行走过程仍然有脚落地时不平稳等问题的出现。这些问题, 都是以后的研究工作中迫切要解决的, 因此在现有的基础上, 可以进行以下方面的研究工作:

(1) 根据 9 个自由度的爬坡机器人的运动学模型, 对爬坡机器人的运动进行数学建模步态规划, 让机器人的每个分动作都可以实现稳定、精确地达到我们预期的位置。

(2) 改进控制器, 对机器人各关节的力矩进行反馈控制, 使其能按照规划的步态并且即时运动。

(3) 对爬坡机器人的双腿支撑期间进行动力学建模。

(4) 采用实际的物理样机进行实验, 验证步态与控制器的效果, 实现机器人的稳定行走。

## 6.3 致谢

本技术报告是在导师王少坤老师的悉心指导和关怀下完成的。王老师知识渊博、思维敏锐、作风严谨, 和蔼可亲。期间, 我在构思方面得到了王老师的悉心指导。他们给我提出了不少宝贵的意见, 督促我顺利完成报告工作, 使我不仅学会了做科学研究的方法, 还学到了很多知识。这些必将给我以后的工作产生深远的影响。在此, 我要向悉心教导我的王老师表示深深的谢意。

在机器人比赛中我们会齐心合力, 集思广益共同完成了比赛的相关的调试和设计工作, 即将参加了 2022 年 11 月的机器人比赛, 我们力争取得好成绩。最后再一次感谢我的指导老师王老师, 谢谢他们的谆谆教诲, 使我收获颇丰。

# 7 第七章 附录

## 附录 1.1 研究及推广价值

为了进一步完善爬坡机器人的研究, 我们在原有动作及实现的功能基础上, 进一步完善结构设计及改进, 在结构完善的基础上, 再来调试及完成更复杂的动作功能设计, 比如可以加快行动速度, 完善动作转变协调性, 具体的可进一步设计。机器人设计及开发随着经济及科技的发展, 这些年来得到了巨大发展, 但好多的机器人设计还是停留在简单的加工和调试上, 真正的智能控制没能充分融入其中, 所以下一步应加大对智能控制的机器人的科研投入研究力度, 鼓励高校和科研机构开展更多的研究机器人的项目。

机器人的相关工作不是一帆风顺的。我们在工作进行中遇到了不少困难和挑

战。这期间是充实的，更是一个提高自我的过程，因为这样一个自己动手自己思考的过程才是一个运用所学的实践过程。通过集体的讨论，个人的奇思，团体的配合，各个的分工，寻求技术支持，每一条通往我们目标的路都不能错过，而这样的路我们走过才会明白很多，才会逐渐成熟老道，才会懂得通往成功该是向着哪个方向。

众所周知，机器人应用于救援、军事、生产等领域已经屡见不鲜。而如今，随着社会的发展，人们都过上安定小康的生活，开始注重的不是温饱，更多的是生活品质与享受。我们设计的双足爬坡机器人不仅能跳舞还能演绎一段历史，带着观众们一起重温经典，这是极大的愉悦与享受，正是经济发展至今所需要的，娱乐大众，带给大众美与享受。

建议学校加大投入力度，借此机会开展第二课堂活动，动员更多的同学参与其中，享受机器人带来的乐趣，锻炼自己动手能力，同时也丰富同学们的课余生活。

### 附录 1.2C 语言程序

```
#include <avr\jhl.h>
```

```
//子函数列表:
```

```
//动作库列表:
```

```
void act_103(void); //左转  
void act_104(void); //左转循环  
void act_101(void); //启动  
void act_102(void); //迈腿 1  
void act_105(void); //迈腿 2  
void act_106(void); //右调 1  
void act_107(void); //右调 2  
void act_108(void); //左调 1  
void act_109(void); //左调 2  
void act_110(void); //坡转平 1  
void act_111(void); //坡转平 2  
void act_112(void); //平转破 1  
void act_113(void); //平转破 2  
void act_117(void); //右转 1  
void act_118(void); //右转循环 1
```

```
unsigned char turn_1=0;  
unsigned char turn_2=0;  
unsigned char line_1=0;
```

```

unsigned char line_2=0;
unsigned char i=0;
unsigned char flag=0;
unsigned char line_1l=0;
unsigned char line_2l=0;
unsigned char turn_1l=0;
unsigned char turn_2l=0;
int main(void)
{
SysInit();//系统初始化
Initfuwei();//伺服马达端口数据初始化
setfuwei();//伺服马达启动
setfuwei();//伺服马达启动
speed=3;//伺服马达速度-3;
act_101();//启动;
for(i=0;i<28;i++)//多次循环：循环变量:i 循环次数:28
{
act_102();//迈腿 1;
line_1l = GetDS(3);//数字输入：将通道 3 值赋予变量 line_1l;
line_1 = GetDS(5);//数字输入：将通道 5 值赋予变量 line_1;
if(line_1==1||line_1l==1)//条件判断：line_1==1||line_1l==1
{
act_106();//右调 1;
act_107();//右调 2;
}
act_105();//迈腿 2;
line_2l = GetDS(4);//数字输入：将通道 4 值赋予变量 line_2l;
line_2 = GetDS(2);//数字输入：将通道 2 值赋予变量 line_2;
if(line_2==1||line_2l==1)//条件判断：line_2==1||line_2l==1
{
act_108();//左调 1;
act_109();//左调 2;
}
}
speed=2;//伺服马达速度-2;
while(1)//永久循环
{
act_102();//迈腿 1;

```

```

line_11 = GetDS(3); //数字输入：将通道 3 值赋予变量 line_11;
line_1 = GetDS(5); //数字输入：将通道 5 值赋予变量 line_1;
if(line_1==1||line_11==1) //条件判断：line_1==1||line_11==1
{
act_106(); //右调 1;
act_107(); //右调 2;
}
turn_1 = GetDS(7); //数字输入：将通道 7 值赋予变量 turn_1;
if(turn_1==1) //条件判断：turn_1==1
{
act_103(); //左转;
for(i=0;i<3;i++) //多次循环：循环变量:i 循环次数:3
{
act_104(); //左转循环;
}
break;
}
turn_2 = GetDS(0); //数字输入：将通道 0 值赋予变量 turn_2;
if(turn_2==1) //条件判断：turn_2==1
{
act_117(); //右转 1;
for(i=0;i<3;i++) //多次循环：循环变量:i 循环次数:3
{
act_118(); //右转循环 1;
}
break;
}
act_105(); //迈腿 2;
line_21 = GetDS(4); //数字输入：将通道 4 值赋予变量 line_21;
line_2 = GetDS(2); //数字输入：将通道 2 值赋予变量 line_2;
if(line_2==1||line_21==1) //条件判断：line_2==1||line_21==1
{
act_108(); //左调 1;
act_109(); //左调 2;
}
turn_1 = GetDS(7); //数字输入：将通道 7 值赋予变量 turn_1;
if(turn_1==1) //条件判断：turn_1==1
{

```



```

act_103();//左转;
for(i=0;i<3;i++)//多次循环：循环变量:i 循环次数:3
{
act_104();//左转循环;
}
break;
}
turn_2 = GetDS(0);//数字输入：将通道 0 值赋予变量 turn_2;
if(turn_2==1)//条件判断： turn_2==1
{
act_117();//右转 1;
for(i=0;i<3;i++)//多次循环：循环变量:i 循环次数:3
{
act_118();//右转循环 1;
}
break;
}
}
speed=2;//伺服马达速度-2;
for(i=0;i<4;i++)//多次循环：循环变量:i 循环次数:4
{
act_112();//平转破 1;
act_113();//平转破 2;
}
for(i=0;i<14;i++)//多次循环：循环变量:i 循环次数:14
{
act_102();//迈腿 1;
line_1 = GetDS(3);//数字输入：将通道 3 值赋予变量 line_1;
line_11 = GetDS(5);//数字输入：将通道 5 值赋予变量 line_11;
turn_1 = GetDS(0);//数字输入：将通道 0 值赋予变量 turn_1;
if(line_1==1||line_11==1||turn_1==1)//    条    件    判    断    :
line_1==1||line_11==1||turn_1==1
{
act_106();//右调 1;
act_107();//右调 2;
}
act_105();//迈腿 2;
line_2 = GetDS(4);//数字输入：将通道 4 值赋予变量 line_2;

```

```

line_21 = GetDS(2); //数字输入：将通道 2 值赋予变量 line_21;
turn_2 = GetDS(7); //数字输入：将通道 7 值赋予变量 turn_2;
if(turn_2==1||line_2==1||line_21==1) // 条 件 判 断 :
turn_2==1||line_2==1||line_21==1
{
act_108(); //左调 1;
act_109(); //左调 2;
}
}
while(1) //永久循环
{
act_110(); //坡转平 1;
act_111(); //坡转平 2;
}
}
void SysInit(void)
{
cli(); //禁止所有中断
DigitalInInit(7); //数字输入初始化
SevroInit(0); //伺服马达初始化
sei(); //开全局中断
}
void Initfuwei(void)
{
unsigned char i;
a[0]=294;
a[1]=298;
a[2]=250;
a[3]=250;
a[4]=315;
a[5]=202;
a[6]=250;
a[7]=250;
a[8]=250;
a[9]=250;
a[10]=288;
a[11]=241;
a[12]=250;

```

```
a[13]=250;
a[14]=253;
a[15]=270;
for(i=0;i<24;i++)b[i]=a[i];
}

void setfuwei(void)
{
zx(294);
zx(1322);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11505);
zx(12538);
zx(13562);
zx(14589);
zx(15630);
zx(50192);
}

//左转
void act_103(void)
{
zx(1307);
zx(10512);
zx(50192);
zx(268);
zx(1307);
zx(2298);
zx(3322);
zx(4416);
zx(5355);
zx(6394);
```

zx(7418);  
zx(8442);  
zx(9466);  
zx(10512);  
zx(11479);  
zx(12538);  
zx(13512);  
zx(14566);  
zx(15629);  
zx(50192);  
zx(1318);  
zx(10532);  
zx(50192);  
zx(4389);  
zx(15608);  
zx(50192);  
zx(294);  
zx(1318);  
zx(2298);  
zx(3322);  
zx(4389);  
zx(5291);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10532);  
zx(11513);  
zx(12538);  
zx(13512);  
zx(14564);  
zx(15608);  
zx(50192);  
zx(4416);  
zx(15631);  
zx(50192);  
zx(1312);  
zx(10503);

```
zx(50192);
zx(271);
zx(1312);
zx(2298);
zx(3322);
zx(4416);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10503);
zx(11479);
zx(12538);
zx(13512);
zx(14591);
zx(15631);
zx(50192);
}
//左转循环
void act_104(void)
{
zx(270);
zx(1326);
zx(10526);
zx(50192);
zx(270);
zx(1326);
zx(2298);
zx(3322);
zx(4384);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10526);
zx(11479);
```

zx(12538);  
zx(13512);  
zx(14591);  
zx(15610);  
zx(50192);  
zx(294);  
zx(1326);  
zx(2298);  
zx(3322);  
zx(4384);  
zx(5294);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10526);  
zx(11511);  
zx(12538);  
zx(13512);  
zx(14564);  
zx(15610);  
zx(50192);  
zx(4415);  
zx(15634);  
zx(50192);  
zx(1309);  
zx(10503);  
zx(50192);  
zx(270);  
zx(1309);  
zx(2298);  
zx(3322);  
zx(4415);  
zx(5326);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);

```
zx(10503);
zx(11479);
zx(12538);
zx(13512);
zx(14592);
zx(15634);
zx(50192);
}
//启动
void act_101(void)
{
zx(294);
zx(1322);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11505);
zx(12538);
zx(13562);
zx(14589);
zx(15630);
zx(50192);
zx(1299);
zx(10499);
zx(50192);
zx(272);
zx(1299);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
```



```
zx(7418);
zx(8442);
zx(9466);
zx(10499);
zx(11534);
zx(12538);
zx(13612);
zx(14589);
zx(15630);
zx(50192);
zx(1325);
zx(10530);
zx(50192);
}
//迈腿1
void act_102(void)
{
zx(305);
zx(1325);
zx(2298);
zx(3322);
zx(4370);
zx(5355);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10543);
zx(11484);
zx(12538);
zx(13562);
zx(14570);
zx(15578);
zx(50192);
zx(305);
zx(1325);
zx(2298);
zx(3322);
```

```
zx(4370);
zx(5303);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10543);
zx(11484);
zx(12538);
zx(13512);
zx(14611);
zx(15578);
zx(50192);
zx(305);
zx(1325);
zx(2298);
zx(3322);
zx(4426);
zx(5303);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10543);
zx(11484);
zx(12538);
zx(13562);
zx(14611);
zx(15629);
zx(50192);
}
//迈腿 2
void act_105(void)
{
zx(305);
zx(1285);
zx(2298);
zx(3322);
```

zx(4426);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10499);  
zx(11484);  
zx(12538);  
zx(13562);  
zx(14570);  
zx(15629);  
zx(50192);  
zx(269);  
zx(1285);  
zx(2298);  
zx(3322);  
zx(4426);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10499);  
zx(11534);  
zx(12538);  
zx(13612);  
zx(14570);  
zx(15629);  
zx(50192);  
zx(269);  
zx(1322);  
zx(2298);  
zx(3322);  
zx(4426);  
zx(5355);  
zx(6394);  
zx(7418);

```
zx(8442);
zx(9466);
zx(10543);
zx(11534);
zx(12538);
zx(13612);
zx(14570);
zx(15629);
zx(50192);
}
//右调1
void act_106(void)
{
zx(316);
zx(1302);
zx(2298);
zx(3322);
zx(4416);
zx(5355);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10495);
zx(11475);
zx(12538);
zx(13512);
zx(14566);
zx(15629);
zx(50192);
zx(290);
zx(1302);
zx(2298);
zx(3322);
zx(4416);
zx(5355);
zx(6394);
zx(7418);
```

```
zx(8442);
zx(9466);
zx(10495);
zx(11501);
zx(12538);
zx(13612);
zx(14566);
zx(15629);
zx(50192);
zx(1324);
zx(10528);
zx(50192);
}
//右调 2
void act_107(void)
{
zx(317);
zx(1324);
zx(2298);
zx(3322);
zx(4385);
zx(5328);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11475);
zx(12538);
zx(13612);
zx(14566);
zx(15605);
zx(50192);
zx(317);
zx(1322);
zx(2298);
zx(3322);
zx(4385);
```

```
zx(5285);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11475);
zx(12538);
zx(13512);
zx(14611);
zx(15605);
zx(50192);
zx(4409);
zx(15628);
zx(50192);
}
//左调 1
void act_108(void)
{
zx(293);
zx(1322);
zx(2298);
zx(3322);
zx(4371);
zx(5359);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11503);
zx(12538);
zx(13562);
zx(14566);
zx(15609);
zx(50192);
zx(293);
zx(1322);
```

```
zx(2298);
zx(3322);
zx(4371);
zx(5328);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10528);
zx(11502);
zx(12538);
zx(13612);
zx(14586);
zx(15609);
zx(50192);
zx(4411);
zx(15630);
zx(50192);
}
//左调 2
void act_109(void)
{
zx(293);
zx(1308);
zx(2298);
zx(3322);
zx(4411);
zx(5355);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10507);
zx(11502);
zx(12538);
zx(13562);
zx(14571);
zx(15630);
```

```
zx(50192);
zx(265);
zx(1308);
zx(2298);
zx(3322);
zx(4411);
zx(5355);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10507);
zx(11524);
zx(12538);
zx(13512);
zx(14571);
zx(15630);
zx(50192);
zx(1327);
zx(10530);
zx(50192);
}
//坡转平1
void act_110(void)
{
zx(315);
zx(1325);
zx(2298);
zx(3322);
zx(4370);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10530);
zx(11474);
zx(12538);
```



zx(13612);  
zx(14589);  
zx(15608);  
zx(50192);  
zx(315);  
zx(1325);  
zx(2298);  
zx(3322);  
zx(4370);  
zx(5295);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10530);  
zx(11474);  
zx(12538);  
zx(13612);  
zx(14616);  
zx(15608);  
zx(50192);  
zx(315);  
zx(1325);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5295);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10530);  
zx(11474);  
zx(12538);  
zx(13612);  
zx(14616);  
zx(15669);  
zx(50192);

```
}  
//坡转平 2  
void act_111(void)  
{  
zx(311);  
zx(1305);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10499);  
zx(11474);  
zx(12538);  
zx(13612);  
zx(14566);  
zx(15669);  
zx(50192);  
zx(267);  
zx(1305);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10499);  
zx(11529);  
zx(12538);  
zx(13512);  
zx(14566);  
zx(15669);  
zx(50192);
```

```
zx(1372);
zx(10528);
zx(50192);
}
//平转破1
void act_112(void)
{
zx(315);
zx(1325);
zx(2298);
zx(3322);
zx(4370);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10530);
zx(11474);
zx(12538);
zx(13562);
zx(14589);
zx(15578);
zx(50192);
zx(315);
zx(1325);
zx(2298);
zx(3322);
zx(4370);
zx(5295);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10530);
zx(11474);
zx(12538);
zx(13612);
```

```
zx(14616);
zx(15578);
zx(50192);
zx(315);
zx(1325);
zx(2298);
zx(3322);
zx(4416);
zx(5295);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10530);
zx(11474);
zx(12538);
zx(13562);
zx(14616);
zx(15629);
zx(50192);
}
//平转破 2
void act_113(void)
{
zx(315);
zx(1275);
zx(2298);
zx(3322);
zx(4416);
zx(5355);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10499);
zx(11474);
zx(12538);
zx(13562);
```

zx(14566);  
zx(15629);  
zx(50192);  
zx(269);  
zx(1275);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10499);  
zx(11529);  
zx(12538);  
zx(13512);  
zx(14566);  
zx(15629);  
zx(50192);  
zx(269);  
zx(1322);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5355);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10528);  
zx(11529);  
zx(12538);  
zx(13512);  
zx(14566);  
zx(15629);  
zx(50192);  
}

```
//右转 1
void act_117(void)
{
zx(288);
zx(1307);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10512);
zx(11500);
zx(12538);
zx(13562);
zx(14589);
zx(15630);
zx(50192);
zx(308);
zx(1307);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10512);
zx(11524);
zx(12538);
zx(13562);
zx(14589);
zx(15630);
zx(50192);
zx(308);
```

zx(1318);  
zx(2298);  
zx(3322);  
zx(4411);  
zx(5322);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10532);  
zx(11524);  
zx(12538);  
zx(13562);  
zx(14589);  
zx(15630);  
zx(50192);  
zx(308);  
zx(1318);  
zx(2298);  
zx(3322);  
zx(4389);  
zx(5322);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10532);  
zx(11524);  
zx(12538);  
zx(13562);  
zx(14589);  
zx(15608);  
zx(50192);  
zx(282);  
zx(1318);  
zx(2298);  
zx(3322);  
zx(4389);

zx(5353);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10532);  
zx(11497);  
zx(12538);  
zx(13562);  
zx(14614);  
zx(15608);  
zx(50192);  
zx(282);  
zx(1318);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5353);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10532);  
zx(11497);  
zx(12538);  
zx(13562);  
zx(14614);  
zx(15631);  
zx(50192);  
zx(282);  
zx(1312);  
zx(2298);  
zx(3322);  
zx(4416);  
zx(5353);  
zx(6394);  
zx(7418);  
zx(8442);



```
zx(9466);
zx(10503);
zx(11497);
zx(12538);
zx(13562);
zx(14614);
zx(15631);
zx(50192);
zx(305);
zx(1312);
zx(2298);
zx(3322);
zx(4416);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
zx(9466);
zx(10503);
zx(11525);
zx(12538);
zx(13562);
zx(14587);
zx(15631);
zx(50192);
}
//右转循环1
void act_118(void)
{
zx(308);
zx(1326);
zx(2298);
zx(3322);
zx(4411);
zx(5322);
zx(6394);
zx(7418);
zx(8442);
```

zx(9466);  
zx(10512);  
zx(11500);  
zx(12538);  
zx(13562);  
zx(14589);  
zx(15630);  
zx(50192);  
zx(308);  
zx(1326);  
zx(2298);  
zx(3322);  
zx(4384);  
zx(5322);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10512);  
zx(11500);  
zx(12538);  
zx(13562);  
zx(14589);  
zx(15610);  
zx(50192);  
zx(284);  
zx(1326);  
zx(2298);  
zx(3322);  
zx(4384);  
zx(5350);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10512);  
zx(11499);  
zx(12538);

zx(13562);  
zx(14614);  
zx(15610);  
zx(50192);  
zx(284);  
zx(1326);  
zx(2298);  
zx(3322);  
zx(4415);  
zx(5350);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10512);  
zx(11499);  
zx(12538);  
zx(13562);  
zx(14614);  
zx(15634);  
zx(50192);  
zx(284);  
zx(1309);  
zx(2298);  
zx(3322);  
zx(4415);  
zx(5350);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10503);  
zx(11499);  
zx(12538);  
zx(13562);  
zx(14614);  
zx(15634);  
zx(50192);

zx(308);  
zx(1309);  
zx(2298);  
zx(3322);  
zx(4415);  
zx(5318);  
zx(6394);  
zx(7418);  
zx(8442);  
zx(9466);  
zx(10503);  
zx(11527);  
zx(12538);  
zx(13562);  
zx(14586);  
zx(15634);  
zx(50192);  
}