

普通高等教育“十三五”规划教材

徐振平 编著

# 机器人 控制技术基础



——基于Arduino的四旋翼飞行器设计与实现  
Fundamentals of Robot Control Technology

——Design and Implementation of Quadcopter Based on Arduino



国防工业出版社  
National Defense Industry Press

普通高等教育“十三五”规划教材

# 机器人控制技术基础

——基于 Arduino 的四旋翼飞行器设计与实现

徐振平 编著

国防工业出版社

·北京·

## 内 容 简 介

本书采用 Arduino 单片机的硬件和软件,结合网上公开的四旋翼飞行器源代码,系统介绍了机器人制作的基础技术。其主要内容包括机器人通用的输入输出设备、传感器及其滤波技术、机器人位姿分析、控制技术以及与计算机通信的串口技术、综合应用的 GPS 技术,并分析了四旋翼飞行器源代码的文件构成和相关制作技术等。

本书可作为高等院校对单片机有一定了解和初步掌握 C++ 语言相关的工程类专业学生的教材,也可供对机器人有兴趣的人员进行学习和开发。

### 图书在版编目(CIP)数据

机器人控制技术基础:基于 Arduino 的四旋翼飞行器设计与实现/徐振平编著. —北京:国防工业出版社, 2017. 4

普通高等教育“十三五”规划教材

ISBN 978-7-118-11007-4

I. ①机… II. ①徐… III. ①智能机器人—机器人控制—高等学校—教材 IV. ①TP242. 6

中国版本图书馆 CIP 数据核字(2017)第 055821 号

※

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

腾飞印务有限公司印刷

新华书店经售

\*

开本 787×1092 1/16 印张 20 $\frac{1}{4}$  字数 472 千字

2017 年 4 月第 1 版第 1 次印刷 印数 1—4000 册 定价 48.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)88540777

发行邮购:(010)88540776

发行传真:(010)88540755

发行业务:(010)88540717

# 前 言

机器人是新时期的一个经济增长点,让机器人技术深入人们的日常生活,当代大学生和机器人爱好者责无旁贷。本书将计算机技术、自动控制技术、电子制作技术和机械与材料工程等相关知识结合起来。只要懂 C++ 语言,就可以学习本书,进而开发出自己所需要的机器人。

机器人控制技术是一门综合知识应用的学科。首先在自动控制原理理论方面,涉及 Laplace 变换与控制系统的稳定性、准确性和快速性的分析以及系统校正的知识,需要有一套对这些知识加以系统理解和综合应用的代码;其次在机械与材料工程方面,涉及机械零部件的制作和组合,需要加上智能化的单片机对机械系统进行控制;最后在电子制作技术方面,不仅需要掌握电路的设计与布线以及传感器校准等知识,同时也需要进行计算机语言的优化编程。在计算机编程方面,需要将控制系统和各种硬件编程有机地结合起来才能发挥优势。Arduino 系统集成了硬件和软件两个方面,可以输入/输出数字信号和模拟信号,能与传感器、上位机通信方便,操作简单,采用独特的配置函数和循环函数进行编程,不需要操作系统支持,对深入理解控制理论的实现非常有益。

对于机器人系统,本书选用四旋翼飞行器,首先得益于该机器人不但已经应用于社会生产的多个方面,而且也是广大青少年及航模爱好者之首选。本书力图将这些零散的知识集中起来,有机组合,让读者对机器人控制技术有一个综合、深入的理解,为进一步的创新开发打好基础。

飞控代码更新日新月异,作者经历了 1.8 版~2.4 版的更替,通过将所学的机器人技术和计算机技术结合起来,写了这本书,与大家分享。飞行机器人只是机器人的一个方面,但机器人系统的基础开发是近似的,可以触类旁通有利于其他机器人的开发。

本书分为十章,第一章对机器人的分类、组成、应用以及技术参数进行了简要的阐述,因为主要针对四旋翼飞行器,所以对四旋翼的部件也进行了简介;第二章对 Arduino 单片机的硬件和软件知识进行了详细的讲述,并对硬件和软件所结合的寄存器及中断进行列表,方便读者查阅;第三章机器人的输入和输出讲解了遥控和电机的应用代码以及单片机资源的运用;第四章是传感器数据的获取及其校准,涉及传感器通信的 I2C 协议和与飞控有关的陀螺仪、加速度传感器、磁场传感器及气压测高仪等传感器;第五章主要对遥控、电机以及传感器数据的滤波处理进行详细介绍,除了平滑滤波算法外,还讲解了 Kalman 滤波和互补滤波器等;第六章对机器人的运动学、微分学、动力学理论进行了讲解,然后在飞控的姿态计算中加以运用,为了改进算法,最后还对四元数进行了讲解;第七章是机器人控制技术基础,讲述了 Laplace 变换、传递函数的建立过程以及控制系统的稳定性、准确性和快速性的分析以及系统的 PID 校正的知识,并对飞控中的 PID 算法进行了详细注解;第八章是机器人与计算机通信的串口协议的介绍,包含了串口协议的底层开发以及串

口协议在单片机端和计算机端(采用 Java 编写)的应用;第九章对 GPS 模块的应用进行系统介绍,包括 GPS 的原理,飞控中程序的介绍和其模块程序在飞控主函数中的作用;第十章注解了机械臂和飞控的源代码以及介绍了相关的航模知识。

本书的编写参考了网上论坛的讨论,并得到了老师和学生大力支持。徐雄和肖辉参与了四旋翼飞行器制作,董瑞智参与了 Arduino 硬件和软件、传感器、GPS 的编写,郭顺杰参与了 I2C 协议、串口协议的编写,李泽文参与了绪论和航模知识介绍的编写,朱应成参与了滤波算法列举和机械臂控制代码的编写。同时,得到了长江大学机器人协会、文汉云的国家自然科学基金面上项目“随钻测量井下网络化光纤传感器及信息传输关键技术研究”(编号 41372155)和长江大学博士启动基金“随钻电阻率测井解析算法研究”(编号 801160010123)的大力支持,在此一并表示衷心的感谢!

本书主要是对四旋翼飞行器的代码进行解读和注释,注释的过程也是一个学习的过程,作者在编写的过程中学到了很多知识,所以贡献出来与大家分享。由于作者知识和语言表达能力有限,不妥之处,敬请读者谅解,也欢迎大家来信提出宝贵意见,不胜感激。联系方式:Email:xzp18@sohu.com。

# 目 录

第一章 绪论 .....	1
第二章 Arduino 语言及单片机介绍 .....	7
第一节 Arduino 简介 .....	7
第二节 Arduino 的硬件部分 .....	8
第三节 Arduino 的软件部分 .....	16
第四节 AVR 寄存器介绍 .....	21
第五节 中断 .....	35
第三章 机器人的输入和输出 .....	37
第一节 存储器的读写 .....	37
第二节 遥控器 PPM 编码及其硬件介绍 .....	49
第三节 电动机及其驱动 .....	52
第四节 机器人其他 I/O 资源的输入和输出 .....	61
第四章 传感器及其校正 .....	79
第一节 I2C 协议 .....	79
第二节 传感器介绍 .....	94
第五章 机器人数据的滤波算法 .....	135
第一节 滤波算法列举 .....	135
第二节 飞控中的滤波算法 .....	147
第六章 机器人位姿分析 .....	167
第一节 机器人运动学 .....	167
第二节 机器人运动的微分变换 .....	177
第三节 机器人动力学 .....	179
第四节 飞控上位姿计算 .....	181
第五节 四元数和旋转矩阵 .....	192
第七章 机器人的控制技术基础 .....	200
第一节 机器人控制技术理论基础 .....	200

第二节	飞控中的控制代码	212
<b>第八章</b>	<b>串口通信及其数据处理</b>	<b>224</b>
第一节	串口通信的底层协议	224
第二节	串口通信应用过程(单片机端)	233
第三节	串口通信应用过程(上位机端)	247
<b>第九章</b>	<b>GPS 应用</b>	<b>257</b>
第一节	GPS 原理	257
第二节	飞控中的代码介绍	259
第三节	GPS 模块代码在飞控主函数中的作用	263
<b>第十章</b>	<b>实例分析</b>	<b>266</b>
第一节	机器人机械臂控制的代码开发	266
第二节	四轴飞控的代码解读	269
<b>参考文献</b>		<b>316</b>

# 第一章 绪 论

提起机器人,大多数人会觉得它像电影里的一样,外观像人但比人拥有特殊能力的铁怪物。其实简单来说,机器人是一种能够进行编程并在自动控制下执行某些操作和移动作业任务的机械装置。按照结构形态、负载能力和动作空间划分,它可分为超大机器人、大型机器人、中型机器人、小型机器人和超小型机器人;按照开发内容和目的划分,可分为工业机器人(Industrial Robot,如焊接、喷漆、装配机器人)、操纵机器人(Teleoperator Robot,如主从手、遥控排险、水下作业机器人)和智能机器人(Intelligent Robot 如演奏、表演、下棋、探险机器人)。

特种机器人、微型机器人和微动机器人是目前机器人发展的重要方向。其中:特种机器人(Special Robots),如航天飞机上的机械手、海洋探测机器人、军用机器人、防核防化机器人、爬壁机器人、微小物体操作机器人等;微型机器人(Micro-robots)体积小,如管道机器人、血管疏通机器人;微动机器人(Micro-movement Robots)动作小、精度高,如细胞切割机器人、微操作和微装配机器人等。

机器人研究与应用在各个方面,机器人技术主要涉及计算机技术、自动控制技术、机械工程、电子信息技术等专业的知识。尽管机器人的涉及面广、种类和功能又各不相同,但现在应用最多的工业机器人都有一个共同的特点,其核心都主要由执行机构、驱动和传动装置、传感器和控制器四大部分构成,见图 1.1。

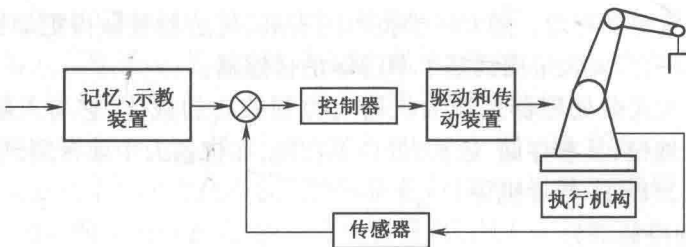


图 1.1 工业机器人系统结构

**执行机构:**机器人的足、腿、手、臂、腰及关节等,它是机器人运动和完成某项任务所必不可少的组成部分。

**驱动和传动装置:**用来有效地驱动执行机构的装置,通常采用液压、电动和气动,有直接驱动和间接驱动两种方式。

**传感器:**是机器人获取环境信息的工具,如视觉、听觉、嗅觉、触觉、力觉、滑觉和接近觉传感器等,它们的功能相当于人的眼、耳、鼻、皮肤及筋骨。

**控制器:**是机器人的核心,它负责对机器人的运动和各种动作控制及对环境的识别。



## 1. 机器人的组成

不论工业机器人还是其他方面的机器人,其核心组成都是相同的,均由机械硬件部分、电子硬件部分、控制部分和底层软件部分组成。每部分有各自的特点,但又紧密联系,相互协调完成任务。

### 1) 机器人的机械硬件部分

机械系统又称为操作机构或执行机构系统,不论外观是什么样的机器人,都是由一系列连杆、关节或其他形式的运动附件所组成。机械系统通常包括机座、立柱、臂关节、腕关节和手部等。工业机器人的机械系统由机身、手臂、末端执行器三大部件组成,每一大部件都有若干自由度,构成一个多自由度的机械系统。若机身具备行走机构,便构成行走机器人;若机身不具备行走及腰转机构,则构成单机器人臂(Single Robot Arm)。手臂一般由上臂、下臂和手腕组成。末端执行器是直接装在手腕上的一个重要部件,它可以是两手指或多手指的手爪,也可以是喷枪、焊枪等作业工具。机器人机械结构部分较多且较复杂,所以在安排这些部件时不仅要符合载重、灵活、平稳的要求,还必须考虑能源通常采用蓄电池等情况。

### 2) 机器人的电子硬件部分

机器人的电子硬件部分主要是组成该机器人的电子元器件。其主要包括电动机及其驱动装置、传感器等感知装置、单片机等嵌入式微处理器。

电动机及其驱动装置主要指驱动机械系统的驱动装置。根据驱动源的不同,驱动系统可以分为电动、液压、气动以及它们结合起来应用的综合系统。驱动系统可以与机械系统直接连接,也可以通过同步带、链条、齿轮、谐波传动装置等与机械系统间接相连。

传感器等感知系统由内部传感器模块和外部传感器模块组成,获取内部和外部环境状态中有意义的信息。智能传感器的使用提高了机器人的机动性、适应性和智能化水平。人类感知系统对感知外部世界的信息是及其灵巧的,然而对于一些特殊的信息,传感器要比人类的感知系统更加有效。随着科学技术的发展,传感器被做得更加精确化、小型化,在一块感知系统中,需要大量用到基于 MEMS 的传感器。

单片机等嵌入式微处理器是作为机器人控制核心的载体,它与人脑或计算机中的 CPU 具有等同的地位,具有存储、运算、处理等功能,在机器人中通常用到的有 Arduino 单片机、Pic 单片机、STM32 单片机等。

### 3) 机器人的控制部分

机器人的控制系统是机器人的核心,控制系统的任务是根据机器人的作业指令程序以及从传感器反馈回来的信号,支配机器人的执行机构完成规定的运动和功能。若工业机器人不具备信息反馈特征,则为开环控制系统;若具备信息反馈特征,则为闭环控制系统。

控制系统根据控制原理可分为程序控制系统、自适应性控制系统和人工智能控制系统。程序控制系统,这种系统的设定值是变化的,但它是时间的已知函数,即设定值按人规定的时间程序性地变化,如程序控制机床的控制系统的输出量应与给定的变化规律相同,这类系统在间歇生产过程中应用比较普遍,如多种液体自动混合加热控制就属此类,其组成项目有开关信号、输入回路程序控制器、输出回路和执行机械等组成部份;自适应性控制系统是不断地测量系统的输入、状态、输出或性能参数,逐步了解和掌握对象,更新

系统的控制结构和参数,根据控制运动的形式可分为点位控制和轨迹控制;人工智能控制系统是将模糊逻辑控制、模糊预测控制神经网络控制和基于知识的分层控制等技术应用于控制系统,使控制系统具有自适应、自学习和自组织的能力。但不论什么样的分类,它们对底层的要求均满足机体平衡、动作平滑及反应迅速、精确等特点,在底层的控制中多采用 PID 控制,应用上位机可以增加人工智能等复杂的控制。

#### 4) 机器人的底层软件部分

机器人的底层软件部分主要是对机器人进行开发,完成对信号的检测、传输和 PID 控制等。

机器人的开发语言是在单片机等嵌入式微处理器上进行编码的,所以通常采用类 C/C++ 等语言,当然也有用其他语言编写的,并不是仅限于此。

对信号的检测、传输等是通过传感器采集的数据进行处理,然后进行通信。机器人的通信分为:单片机与传感器的通信,通常采用 I2C 协议;单片机与上位机(如 ARM)的通信,通常采用串口协议、Zigbee 协议等;单片机与输入设备的通信,通常采用无线通信协议,如 PPM 遥控器、Zigbee 协议等。通过传输数据的反馈操作,对机器人进行更加平稳的控制。

### 2. 机器人的上位机算法

当机器人处理一些问题时,有时会遇到一些比较复杂的算法,若机器人终端处理不了或需要的存储空间比较大,则需要机器人终端将需要处理的数据通过通信手段传送到上位机服务器(如计算机),由上位机服务器对其进行处理,然后将处理好的数据传回机器人内,再进行下一步操作。

### 3. 机器人的应用

研制机器人的最初目的是帮助人们摆脱繁重的劳动或简单重复的工作,以及替代人类到有辐射等危险环境中进行作业,因此机器人最早在汽车制造业和核工业领域得以应用。随着机器人技术的不断发展,工业领域的焊接、喷漆、搬运、装配和铸造等场合,已经开始大量使用机器人。另外,在军事、海洋探测、航天、医疗、农业、林业甚至服务娱乐行业,也都开始使用机器人。

从机器人的用途来分,可以分为军用机器人和民用机器人两大类。

军用机器人主要用于军事上代替或辅助军队进行作战、侦察、探险等工作。根据不同的作战空间,可分为地面军用机器人、空中军用机器人(即无人飞行器)、水下军用机器人和空间军用机器人等。军用机器人的控制方式一般有自主操控式、半自主操控式、遥控式等多种方式。

在民用机器人中,各种生产制造领域中的工业机器人在数量上占绝对多数,成为机器人家族中的主力军;其他种类的机器人也开始在不同的领域得到研究、开发和应用。总体看来,若按用途分,民用机器人可以分为以下几个主要类别。

#### 1) 工业机器人

制造工业部门应用机器人(工业机器人)的主要目的在于削减人员编制和提高产品质量。与传统的机器相比,它具有两个主要的优点:

(1) 生产过程的几乎完全自动化。

(2) 生产设备的高度适应能力。现在工业机器人主要用于汽车工业、机电工业(包

括电信工业)、通用机械工业、建筑业、金属加工、铸造以及其他重型工业和轻工业部门。

## 2) 服务型机器人

服务型机器人是一种半自主或全自主工作的机器人,它完成的是有益于人类健康的服务工作,但不包括那些从事生产的设备。

服务型机器人有医用机器人、送信机器人、导游机器人、加油机器人、建筑机器人、农业及林业机器人等。其中,爬壁机器人既可用于清洁,又可用于建筑。服务型机器人尚处于开发及普及的早期阶段。

## 4. 四轴飞行器

四轴飞行器(Quadrotor)也称四旋翼飞行器。四轴飞行器的四个螺旋桨都是电动机直连的简单机构,十字形的布局允许飞行器通过改变电动机转速获得旋转机身的力,从而调整自身姿态。因为它固有的复杂性,历史上从未有过大型的商用四轴飞行器。近年来得益于微机电控制技术的发展,稳定的四轴飞行器得到了广泛的关注,应用前景十分可观。

四轴飞行器应用广泛,在机器人的研究中具有代表性。通过对四轴飞行器的研究分析,能了解和掌握机器人控制的原理方法。而且,四轴飞行器对外的拓展功能十分强大。它还深受 DIY 爱好者的青睐,表 1.1 是四轴需要 DIY 的零件。

表 1.1 四轴需要 DIY 的零件

动力总成	无刷电动机(4个)
	电子调速器(俗称电调,4个)
	螺旋桨(4个,需要2个正桨,2个反桨)
控制系统	飞行控制器(俗称飞控)
	遥控器(四通道以上遥控器)
动力储备	电池
	充电器
传感器	陀螺仪
	加速度传感器
	磁场传感器
	气压测高仪
	可选的 GPS 等传感器
结构件	机架

四轴飞行器具有以下特点:

首先,它有相对简单的机械构造。正因为简单,所以安全指数大大提高。无论是作为航空模型还是作为遥控平台,安全永远是第一位的。

其次,有相对稳定性较高。飞行姿态平滑稳定,机械振动尽可能地减小,这是四轴飞行器的又一魅力,装载图像设备再好不过了。

第三,相对成本低廉。花尽可能少的钱获取最大的性价比是我们追求的境界,这为工业开发其商业用途奠定了必要的基础。

四轴飞行器是电子工业高速发展的产物,也局限了它的超越性的成长。其局限性包

括:有效载荷是它的一大瓶颈;有限的飞行时间又是它的致命不足;抗风能力表现脆弱也是大家有目共睹的事实。当然,这些都会随着技术的发展而逐步改善。

还有,单就操纵性能和视觉效果看,传统的遥控直升机要比四轴飞行器更震撼!本书以四轴飞行器为基础对机器人进行讲解。四轴飞行器公开的源代码见网页 [http://www.multiwii.com/wiki/index.php?title=Main\\_Page](http://www.multiwii.com/wiki/index.php?title=Main_Page)。

## 5. 机器人的技术参数

(1) 自由度。自由度是指描述物体运动所需要的独立坐标数。机器人的自由度表示机器人动作灵活的尺度,一般以轴的直线移动、摆动或旋转动作的数目来表示,手部的动作不包括在内。

机器人的自由度越多,就越能接近人手的动作机能,通用性就越好;但是自由度越多,结构越复杂,对机器人的整体要求就越高,这是机器人设计中的一个矛盾。

工业机器人的自由度一般多为4~6个,7个以上的自由度是冗余自由度,用来避障碍物。

(2) 工作空间。机器人的工作空间是指机器人手臂或手部安装点所能达到的所有空间区域,不包括手部本身所能达到的区域。机器人所具有的自由度数目及其组合不同,其运动图形也不同;而自由度的变化量(即直线运动的距离和回转角度的大小)则决定着运动图形的大小。

(3) 工作速度。工作速度是指机器人在工作载荷条件下、匀速运动过程中,机械接口中心或工具中心点在单位时间内所移动的距离或转动的角度。

确定机器人手臂的最大行程后,根据循环时间安排每个动作的时间,并确定各动作同时进行或顺序进行,就可确定各动作的运动速度。分配动作时间除考虑工艺动作要求外,还要考虑惯性和行程大小、驱动和控制方式、定位和精度要求。

为了提高生产效率,要求缩短整个运动循环时间。运动循环包括加速启动、等速运行和减速制动三个过程。过大的加减速速度会导致惯性力加大,影响动作的平稳和精度。为了保证定位精度,加减速过程往往占去较长时间。

(4) 工作载荷。工作载荷是指机器人在规定的性能范围内,机械接口处能承受的最大负载量(包括手部),用质量、力矩、惯性矩来表示。

负载大小主要考虑机器人各运动轴上的受力和力矩,包括手部的重量、抓取工件的重量,以及由运动速度变化而产生的惯性力和惯性力矩。一般低速运行时,承载能力大。为安全考虑,规定在高速运行时所能抓取的工件重量作为承载能力指标。

目前使用的工业机器人,其承载能力范围较大,最大可大9kN。

(5) 控制方式。控制方式是指机器人用于控制轴的方式,是伺服还是非伺服,伺服控制方式是实现连续轨迹还是点到点的运动。

(6) 驱动方式。驱动方式是指关节执行器的动力源形式。

(7) 精度、重复精度和分辨率。

精度:一个位置相对于其参照系的绝对度量,指机器人手部实际到达位置与所需要到达的理想位置之间的差距。机器人的精度主要依存于机械误差、控制算法误差与分辨率系统误差。机器人的精度=1/2 基准分辨率+机构误差。

重复精度:在相同的运动位置命令下,机器人连续若干次运动轨迹之间的误差度量。

如果机器人重复执行某位置给定指令,它每次走过的距离并不相同,而是在一平均值附近变化,该平均值代表精度,而变化的幅度代表重复精度。

分辨率:指机器人每根轴能够实现的最小移动距离或最小转动角度。精度和分辨率不一定相关。一台设备的运动精度是指命令设定的运动位置与该设备执行此命令后能够达到的运动位置之间的差距,分辨率则反映了实际需要的运动位置和命令所能够设定的位置之间的差距。分辨率分为编程分辨率与控制分辨率,统称为系统分辨率。编程分辨率是指程序中可以设定的最小距离单位,又称基准分辨率。控制分辨率是位置反馈回路能够检测到的最小位移量。它们的关系如图 1.2 所示。

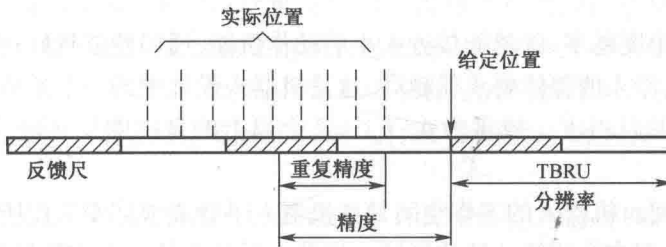


图 1.2 分辨率、精度和重复精度的关系

工业机器人的精度、重复精度和分辨率是根据其使用要求确定的。机器人本身所能达到的精度取决于机器人结构的刚度、运动速度控制和驱动方式、定位和缓冲等因素。

(8) 重复定位精度。重复定位精度是关于精度的统计数据。

(9) 工作范围。工作范围是指机器人手臂末端或手腕中心所能到达的所有点的集合,也称工作区域。

(10) 最大工作速度。最大工作速度是指主要自由度上最大的稳定速度或手臂末端最大的合成速度。

(11) 承载能力。承载能力是指机器人在工作范围内的任何位置所能承受的最大质量。

## 第二章 Arduino 语言及单片机介绍

作为四旋翼飞行机器人里的核心控制部分,Arduino 单片机起着重要的作用。对其语言和各部分的学习不仅有助于对机器人控制部分的深入理解,而且对其进一步的开发与拓展十分必要。

本章介绍制作机器人常采用控制芯片 Arduino 的硬件、软件部分(开发环境、开发程序)和硬软件结合的寄存器和中断。

Arduino 的硬件部分包括不同类型开发板,以及开发板的外部特性。主要介绍 Arduino 外接设备的端口,包括数字引脚、模拟引脚、串行通信、外部中断、PWM 输出、SPI 通信、TWI 通信等。

Arduino 的软件部分包括对开发环境的熟悉和各种语法函数的理解与应用,并通过两个普遍的例子对上述语法函数进行运用,从而可以知道 Arduino 的编程框架结构和语句的使用,并通过串口监视器对其执行结果进行显示。

接着对寄存器及其功用进行介绍,寄存器的详细使用方法将在后文第三章第一节中详细给出。最后讲解了中断的工作过程。

### 第一节 Arduino 简介

Arduino 是一款便捷灵活、方便上手的开源电子原型平台,它是一块单板的微控制器和一整套的开发软件,包含硬件(各种型号的 Arduino 板)和软件(Arduino IDE)。硬件包括一个以 Atmel AVR 单片机为核心的开发板和其他各种 I/O 板。软件包括一个标准语言开发环境和在开发板上运行的烧录程序。其核心是一块单片机芯片,包括处理器、RAM、EEPROM 或闪存和输入/输出的 I/O 引脚。

Arduino 构建于开放的源代码 Simple I/O 介面版,并且具有使用类似 Java、C 语言的 Processing/Wiring 开发环境。主要包含两个主要的部分:一个是硬件部分,即用来做电路连接的 Arduino 电路板;另一个是计算机中的程序开发环境 Arduino IDE。用户在 IDE 中编写程序代码,并且将程序下载到 Arduino 电路板后,程序便会告诉 Arduino 电路板要做什么。那么 Arduino 有哪些用途呢?由于 Arduino 还具有输入输出的引脚,其输入可以是数字信号(开或关)或模拟信号(电压值),这样就可以连接很多不同的传感器来感知外界信息,如光、温度、声音……,输出也可以是数字或模拟信号,所以可以拓张其硬件对外界进行控制,如灯、马达……。这样就可以实现对外界的感知和控制。板子上的微控制器可以通过类 C 语言来编写程序,编译成二进制文件,烧录进微控制器。对 Arduino 的编程是利用 Arduino 编程语言(基于 Wiring)和 Arduino 开发环境(基于 Processing)来实现的。基于 Arduino 的项目,可以只包含 Arduino,也可以包含 Arduino 和其他一些在 PC 上运行的软件,它们之间进行通信(比如 Flash, Processing, MaxMSP)来实现。

Arduino 已经广泛应用于包括四轴飞行器等简易机器人的自动控制系统之中。

## 第二节 Arduino 的硬件部分

提到 Arduino,大多数人想到的都是小型、长方形的(很可能是蓝色的)PCB(Printed Circuit Board,印刷电路板),即 I/O 电路板,如图 2.1 所示。

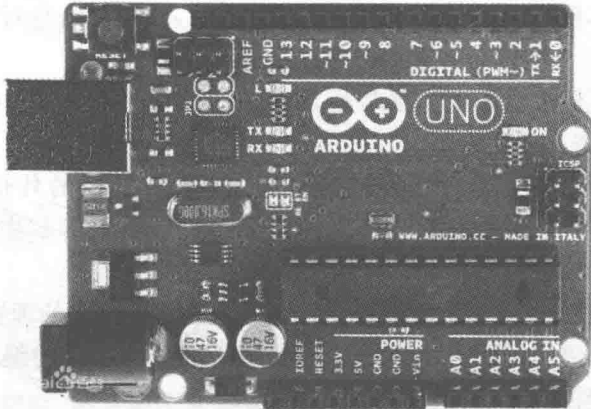


图 2.1 Arduino 实物图

I/O 电路板传统上是基于 Atmel 的 AVR ATmega8 及其后续型号的。I/O 电路板上有机口、电源电路、扩展插槽和其他一些必要的原件。图 2.2 为 I/O 电路板简化的方框图。

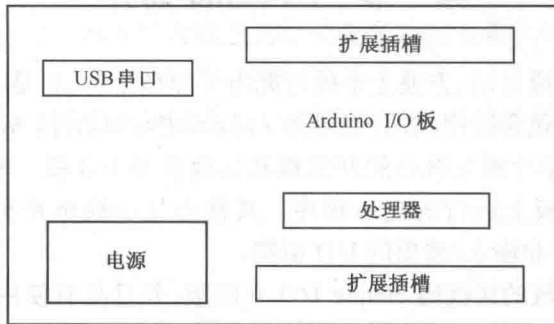


图 2.2 Arduino 电路板简化的方框图

Arduino 的系列包括 Arduino Duemilanove、Arduino Nano、Arduino mini、LilyPad Arduino、Arduino Uno、Arduino Mega2650 等。由于 Arduino Uno 不仅是目前最新、最通用的产品系列,而且能兼容其他的开发板,因此本节重点介绍 Arduino Uno。

Arduino Uno 是 2011 年 9 月 25 日在纽约创客大会(New York Maker Faire)上发布的。型号名字 Uno 是意大利语“一”的意思,表示 Arduino 1.0 版本,即 Uno Punto Zero(意大利语的“1.0”)版。之前的版本,编号为 0001~0022,被认为是 Alpha 版或预先发布版。它的硬件尺寸属于旧版,与之前的 Arduino 板最大的差异在于它不是使用 FTDI USB-to-Serial 串行驱动芯片,而是采用 ATmega8U2 芯片进行 USB 到串行数据的转换。

## 1. 处理器

Arduino Uno 的核心是 Atmel AVR ATmega328, 是一个黑色、长方形、两侧各有一排引脚的塑料块。它实质上就是单芯片的计算机, 封装了中央处理单元(CPU)、内存阵列、外围设备和时钟。ATmega 328 的框架图如图 2.3 所示。

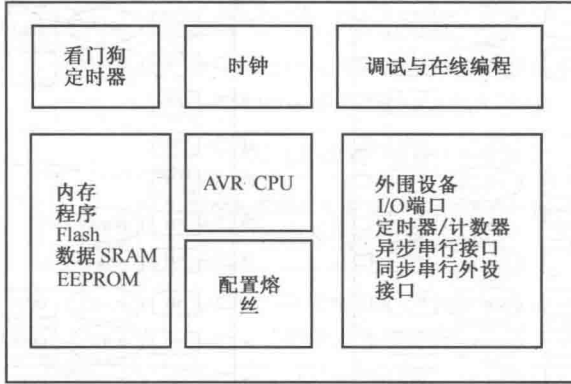


图 2.3 ATmega328 的框架图

ATmega328 芯片是从最初的 Arduino 用的处理器 ATmega8 发展过来的, 比之前的型号内存更大, 片内外围设备功能更多, 同时功耗更小。ATmega328 处理器可以在很宽的供电电压下工作, 从 1.8~5.5V 都可以, 因此很适合用于电池供电的应用程序。在最低供电电压下, 处理器最高只能在 4MHz(每秒 400 万个周期)的时钟频率下工作。供电电压提高到 2.7V, 时钟频率就可以提高到 10MHz。如果要以最高的 20MHz 的时钟频率工作, 芯片至少需要 4.5V 的供电电压。Arduino I/O 电路板供给 ATmega328 芯片的是 5.0V, 因此它可以工作在最高 20MHz 以内的任何时钟频率上。

## 2. 各端口介绍

Arduino 对外界的感知和控制是通过各端口的外接设备来实现的, 因此有必要对各端口进行详细的了解, 以便后续进一步学习。

Arduino Uno 包括: 13 个数字引脚 0~13, 模拟引脚 A0~A5(为区分数字引脚, 在引脚前面加 A), 串行通信 0、1(0 作为 RX, 接收数据, 1 作为 TX, 发送数据), 外部中断 2、3, PWM 输出 3、5、6、9、10、11, SPI 通信 10(SS)、11(MOSI)、12(MISO)、13(SCK), 板上 LED13, TWI 通信 A4(SDA)、A5(SCL), 如图 2.4 所示。

其中各种图形代表意义:

□: 端口引脚。

□: A/D 转换模拟输入引脚。

○: 引脚变化中断。

□: 功能引脚。

○: Arduino 实际引脚。

▤: 串口引脚。

◇: PWM 引脚。



