

基于Arduino的仿生机械手设计

张诗寒 张 勇 刘思佳 谢宇轩 张 倩
安徽大学 电气工程与自动化学院 安徽合肥 230601

摘要: 本文设计了一种基于物联网的手势跟随机械臂, 深入分析了 Arduino 作为机械臂主控器的原理和程序, 实现机械臂的手势跟随, 同时也让特种机器人的工作更加智能化。

关键词: 物联网; 手势跟随; Arduino

1 引言

目前随着人工智能的蓬勃发展, 在人机交互的条件下, 使机械臂的应用场景更加多样化。优傲机器人公司推出全新 UR16e, UR16e 机器人拥有更高的载荷, 其半径为 900mm, 重复精度为 $\pm 0.05\text{mm}$ 。结合这些优势, UR16e 非常适合执行一些高负重的自动化任务, 如重型物料处理、重型零件搬运和机床管理^[1]。从近几年国外企业推出的工业机械臂产品参数与功能来看, 工业机械臂技术已经开始向人工智能控制、模块可重构、微型系统化的方向发展。Arduino 是一款便捷灵活、方便上手的开源电子原型平台。Arduino 能通过各种各样的传感器来感知环境, 通过控制灯光、马达和其他的装置来反馈、影响环境。因此可以利用 Arduino 作为核心控制部件, 完成手掌与机械臂之间的动作同步设计。

2 控制原理

本设计可分为数据手套与机械臂两部分。数据手套部分由主控器、传感器与无线发送模块所组成。由传感器采集人手变化获取手指弯曲信息。主控器负责接收由传感器传来的信号, 经过处理器处理后, 由无线模块发送端发送到从控制器。机械臂部分由无线接收模块、从控制器以及机械臂舵机组成。从控制器利用无线模块接收到数据, 控制机械臂舵机的运动。

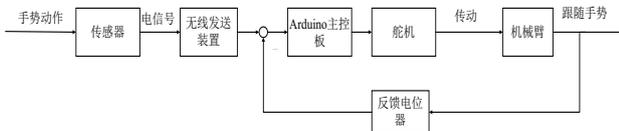


图 1

主控板连接多个舵机, 不同的手势对应不同的舵机旋转角度。经过查阅角度数据如下表所示^[2]:

表 1

	拇指	食指	中指	无名指	小指
伸直	0°	0°	0°	0°	0°
OK	140°	60°	0°	0°	0°

续表:

	拇指	食指	中指	无名指	小指
握拳	140°	120°	120°	120°	140°

控制过程的流程图:

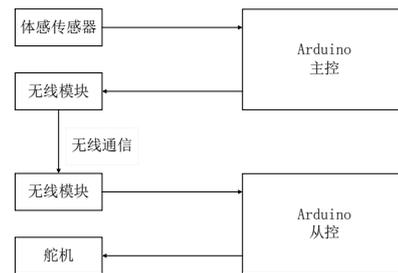


图 2

3 手套板程序设计

3.1 手套板控制流程图

手套版的控制流程图:

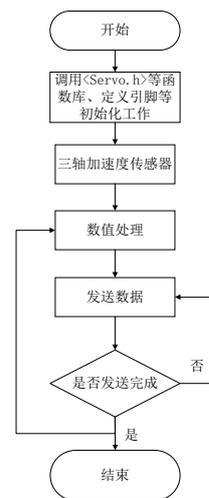


图 3

3.2 手套板程序分析

手套版的程序进入主体程序前的准备有: 1、调用的函数库声明: <Servo.h>、<Wire.h>、<I2Cdev.h>、<ADXL345.h>, 其中<Servo.h>是 Arduino 舵机控制库, 通过设置参数可以便捷地控制舵机的旋转角度, <Wire.h>

<I2Cdev.h>是Arduino和I2C设备通讯使用的库,实现数据传输,<ADXL345.h>是三轴加速度传感器ADXL的函数库。2、定义各种需要的引脚、电位器数量。3、创建需要的舵机类数组等,初始化各种电位器数组的初值。

至此进入主程序中的setup函数(只运行一次)。在该函数的主体内容有:1、初始化串口监视器波特率、初始化各种引脚输入输出模式(如LED引脚就是输出模式)、初始化wire库,2、读取电位器的极值和陀螺仪校准值、连接舵机和引脚,初始化舵机,3、启动,利用进入函数前的事先声明将各种需要的引脚设置高低电平。

手套版的核心程序是循环执行的loop函数。需要先自定义LED闪烁函数loop_nled(),使用millis()函数(此函数为Arduino自带)控制Arduino开发板LED的点亮和熄灭。在进行手势动作前需要先进行校准动作,进行陀螺仪和手指的校准,需要写一个自定义校验函数loop_key_j(),分为两次校准,分别两次写入手指采集的最大值与最小值(握手值:在舒适的方式下最大程度的握紧手指,此时按下校准按钮、伸手值:在舒适的方式下最大程度的伸开手指,此时按下校准按钮,分别写入陀螺仪x,y的校准值到EEPROM。校准过后就要进行传感器的数据收集,传输,这一部分由自定义旋钮控制函数loop_Gihand_R()完成。首先自定义函数adc_read2buf()获得五个手指数据,用myAdxl345加速度计函数库里的函数(传感器相关),然后进行Pwm控制,对舵机进行调速,最后对电位器数据进行处理,电位器数据备份值处理。

4 机械手臂控制板程序设计

4.1 机械手臂控制板程序流程图

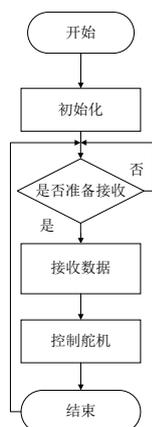


图4

4.2 机械手臂控制板程序分析

主控板程序首先声明各个引脚及所需要的变量,声明调用Servo.h和flash函数库。还需定义偏差存储的地址、表明每隔20ms处理一次(累加)舵机的PWM增量。

由于要控制舵机,故创建一个舵机类数组和定义一个数组用来存储每小组动作组。用结构体变量声明一个舵机变量组,设定舵机当前值、目标值、执行时间和舵机值增量(以20ms为周期)。

完成主控板的程序准备工作后进入setup函数(只运行一次),进行串口初始化、读取全局变量、LED灯闪烁初始化、调用启动提示函数(蜂鸣器作用发出声音),调用舵机初始化函数,设置引脚的输入输出模式,定义所需要的变量。

完成loop函数初始化各项部件,至此进入loop函数(循环执行)。首先如果确定启动控制板,会调用loop_nled()函数,此函数为自行定义,让LED灯闪烁,每秒闪烁一次。后续调用自行定义的函数loop_uart()、loop_action()、loop_servo(),分别的作用是解析串口接收到的字符串指令、循环执行是否需要读取数据执行动作组、舵机PWM增量处理,每隔SERVO_TIME_PERIOD毫秒处理一次,这样就实现了舵机的连续控制。

5 实物效果

5.1 运行前的准备

在应用Arduino开发环境之前,先在电脑端添加新硬件Arduino uno控制板,它是采用ATmega3280芯片进行USB到串行数据的转换,然后将Arduino Sensor ShieldV5.0扩展板会通过串行接口连接到控制板上,再将舵机连接到扩展板的I/O接口上,并连接电源。控制板的DC输入端供电电压为9V,给控制板供电,再将两节3.7V的电池通过DC降压模块将其降至5V后连接到扩展板的Uoc接口给舵机供电。在开发环境中编辑好程序后,下载到Arduino控制板,接通电源,扩展板会接受到指令控制机械臂立刻执行操作。

5.2 实物展示



图5

参考文献:

[1] 于红, 优傲机器人高载荷协作机器人UR16e亮相工博会【J】, 今日制造与升级, 2019(10):40
[2] 张涛, 郑丽英, 黄厚志, 汤永清, 人体体感感知机械手设计与实现【J】, 电子制作, 2019(13):18-19