# 基于 iRobot 的智能机器人设计

一、iRobot 简介

美国 iRobot 公司于 1990 年由美国麻省理工学院教授罗德尼•布鲁克斯、科林•安格尔和海伦•格 雷纳创立,为全球知名 MIT 计算器科学与人工智能实验室技术转移及投资成立的机器人产品与技术专 业研发公司。 iRobot 发明各种军用、警用、救难、侦测机器人,轻巧实用,被军方、警方、救难单位 用于各种不同场合。近年来,由于 iRobot 具有基本的前进,后退,转弯,碰撞检测等基本功能, iRobot 被广泛的应用于机器人教学,可作为学生学习机器人控制、软件编程的动手实验平台。

NI myRIO 是 NI 公司针对高校教学应用最新推出的嵌入式创新实验与项目开发平台。NI myRIO 基于 Xilinx Zynq 技术,集成了双核 ARM 和 FPGA,使学生可以通过 LabVIEW 图形化编程方式在很短时间内快速实现系统级嵌入式应用的开发。

二、设计目标

利用 myRIO-1900 设计基于 iRobot Create 平台的展示 Demo,通过 PC 上位机控制 iRobot 的运动控制,主要功能如下:

- (1) 实现 iRobot 直线到定点;
- (2) 实现 iRobot 原地旋转,包括:顺时针,逆时针;
- (3) 实现 iRobot 跟踪规划路径,比如:长方形,固定半径圆形,"8"字轨迹等;
- (4) 实现 iRobot 带避障功能的地面自主导航;
- (5) 实现 iRobot 爬 20°以上斜坡,带壁障功能;
- (6) 通过摄像头获取路况等环境信息,并传输到上位机,利用上位机控制 iRobot 运动。

三、系统总体方案设计

该系统由 myRIO-1900 嵌入式开发平台, iRobot Create 平台, iPad, PC 和摄像头五部分组成, 系统总体结构如图 1 所示。



图 1. 系统总体结构组成图

iRobot Create 平台通过充电电池供电,myRIO-1900 供电电源来自 iRobot Create 平台,myRIO-1900 与 iRobot Create 之间通过串口(UART)实现数据交互,比如读取 iRobot 自带传感器数据,发送 iRobot 运动 控制命令等。

网络摄像头通过 USB 将采集到的图像数据传送至 My RIO。My RIO 通过无线子网与上位机连接,将数据 传输至上位机。上位机根据接收到的数据给 My RIO 发送控制命令, My RIO 控制 iRobot Create 控制台的运 动。同时 iRobot Create 控制台上的传感器数据反馈回 My RIO 中。

1、系统工作流程

(1)系统上电初始化: iRobot 开机,向 myRIO 供电,myRIO 开机运行 RT 层程序,然后运行上位机程序,再启动上位机程序,输入上位机的 IP 和相应端口号连接到上位机,实现前面板的投射;

(2)运动控制: 上位机上可以直接操作前面板,比如功能的选择,tab 的切换等各种控件的操作,从 而实现 iRobot 的运动控制,如 Navigation、Hill Climb、Vision 控制等功能;

(3)程序结束:上位机上不仅可以实现各种功能的切换,还可以结束程序的运行。

2、iRobot Create 平台控制

iRobot Cerate 平台的 Open Interface 接口是一个 25 针并口, 外设(myRIO-1900)可以通过串口(UART) 控制 iRobot Create 平台运动, 查询 iRobot 内置传感器的数据, 如 Cliff、Bump、Distance、Angle 等传感器, iRobot Create 内置传感器数据每 15ms 更新一次。

实现主要功能需要用到的命令总结如表1所示,需要使用的传感器数据如表2所示。

关于 myRIO-1900 的数据手册请参考 myRIO 1900 User Guide and Specifications.pdf

关于 iRobot Create 平台的介绍请参考文档 Create iRobot Manual\_Final.pdf。

关于命令和内置传感器的详细介绍及使用方法参考文档 iRobot Create OPEN INTERFACE。

命令	命令码	字节数	说明	
Drive	137	4	V: -500~500mm/s, R: -2000~2000mm	
Drive Direct	145	4	单独控制左轮和右轮速度	
Sensors	142	1	读取一个传感器数据包	
Query List	149	N+1	读取多个传感器数据包	
Stream	148	N+1	连续读取多个传感器数据包,间隔15ms	
Wait Distance	156	2	-32767~32768mm	
Wait Angle	157	2	-32767~32768 degrees	

表1 主要命令总结表

### 3、主要传感器数据

包序号	名称	字节数	说明	
6	Group 6	52	所有传感器数据	
7	Bumps and Wheel Drops	1	Bumper 和 Wheel drop 传感器状态(0,1)	
8	Wall 1 Wall 传感器状态(0, 1)		Wall 传感器状态(0,1)	
9	Cliff Left 1 Cliff Left 传感器状态(0, 1)		Cliff Left 传感器状态(0,1)	
10	Cliff Front Left 1 传感器状态(0,1)		传感器状态(0,1)	
11	Cliff Front Right 1 传感者		传感器状态(0,1)	
12	Cliff Right	1	传感器状态(0,1)	
19	Distance	2	距上次读取运行距离(mm)	
20	Angle 2 #		距上次读取旋转角度(degrees)	
27	Wall Signal	2	Wall 传感器值(0-4095)	
28	Cliff Left Signal	2	Cliff Left 传感器值(0-4095)	
29	Cliff Front Left Signal	2	Cliff Front Left 传感器值(0-4095)	
30	Cliff Front Right Signal	2	Cliff Front Right 传感器值(0-4095)	
31	Cliff Right Signal	2	Cliff Right 传感器值(0-4095)	

#### 表 2 主要传感器数据包

## 4、关于 myRIO-1900 Accelerometer 的校准和滤波

## (1) 校准

myRIO-1900 Accelerometer 的误差主要有两类,Offset 和 Sensitivity。Offset 是指测量值的偏差,比如 在加速度为 0g 的时候,测量结果为 0.05g,则 offset 为 0.05g; Sensitivity 是指测量的灵敏度,例如对重力加 速度 1g 的测量中,结果为 1.05g,则 Sensitivity 为+5%。

$$A_{OUT}[g] = A_{OFF} + (Gain \times A_{ACTUAL}) \tag{1}$$

其中, Aour [g]表示测量输出值, AoFF表示偏差, Gain表示灵敏度, AACTUAL表示真实值。

比较准确的修正方法是每条轴使用两点测量。当一个轴被放置成+g和-g的位置,测量到的输出如下:

$$A_{+1g}[g] = A_{OFF} + (1g \times Gain) \tag{2}$$

$$A_{-1g}[g] = A_{OFF} - (1g \times Gain) \tag{3}$$

由(2)式和(3)式可以推出:

$$A_{OFF}[g] = 0.5 \times \left(A_{+1g} + A_{-1g}\right) \tag{4}$$

$$Gain = 0.5 \times \left(\frac{A_{+1g} - A_{-1g}}{1g}\right) \tag{5}$$

则,校准方程如下:

$$A_{ACTUAL}[g] = \frac{A_{OUT} - A_{OFF}}{Gain}$$
(6)

#### (2) 滤波

由于 myRIO-1900 的三轴加速度计输出的值存在毛刺,所以需要使用低通滤波器进行滤波,这里采用的 滤波方法如下图所示。



#### 图 2 myRIO Accelerometer Filter

**注意**:滤波因子的大小非常重要。滤波因子选取大于0小于1的数值,滤波因子越小滤波效果越好,但 会造成系统延时越多,将比较难及时反映当前传感器的真实值;滤波因子变大滤波效果变差,但系统能更 及时反映传感器的真实值。

经过测试, Demo 演示时滤波因子选取 0.06 时效果较好。

## 四、硬件设计

系统中用到的硬件主要有: myRIO-1900、iRobot Create 平台、网络摄像头和 PC 上位机。 myRIO-1900 与 iRobot Create 平台的硬件连接如图 3 所示。



图 3 myRIO-1900 与 iRobot Create 平台硬件连接图

Demo 实物图如下图所示。



图 4 iRobot Demo 实物图

# 五、软件设计

使用 LabVIEW 编写控制软件,将控制软件部署到 myRIO-1900 上,然后通过 ipad 控制 iRobot 完成相应功能。

用到的软件主要有 NI LabVIEW、myRIO Module、Vision Module。

## 1、前面板

上位机主 VI 为 iRobot Target.vi, RT 层主 VI 为 iRobot Navigation Target.vi。上位机主 VI 前面板如下。



图 5 iRobot Demo 前面板—MODE

MODE	Settings	Camera		
		RO		
		Parameter	Settings	
		Accel Smoothing(0-1) 0.06	Radius 100	
	+	Wheel Speed	Backward Speed	
		Turn Speed	Hil Speed	
	1 and	IMAGE	TODAY	
		new advertising artist the company has pros	<b>i -Robot</b>	

图 6 iRobot Demo 前面板—Settings



图 7 iRobot Demo 前面板—Camera

## 2、程序结构框图(RT 层)



图 6 系统软件结构框图

# 3、状态机框图

iRobot Demo 主要功能及其状态机框图如下所示。

• 机器人直行操作



• 逆时针旋转



• 顺时针旋转



• 方形路线行走



• 圆形路线行走



• "8"字形行走



• 避障模式行走



• 爬坡避障模式



## 六、总结与展望

### 1、总结

该 Demo 主要是利用 myRIO-1900 作为控制器,控制 iRobot Create 平台完成相应任务。Demo 展示的时候,使用 iPad 上的 Data Dashboard 作为人机交互界面,主要完成了以下功能。

- (1) 实现 iRobot 直线到定点;
- (2) 实现 iRobot 原地旋转,包括:顺时针,逆时针;
- (3) 实现 iRobot 跟踪规划路径,比如:长方形,固定半径圆形,"8"字轨迹等;
- (4) 实现 iRobot 地面自主导航,带避障功能;
- (5) 实现 iRobot 爬 20°以上斜坡,带壁障功能。
- (6) 实现 iRobot 地面自主导航到定点,然后爬斜坡,到达平台顶端之后做相应动作。
- (7) 通过摄像头获取路况等环境信息,并传输到 iPad,利用 iPad 三轴陀螺仪远程控制 iRobot 运动。

#### 2、展望

由于 iRobot Create 平台内置传感器数量有限,较多复杂功能无法实现,比如避障是通过 bumper 实现的, 后续可以考虑处理摄像头采集的图像,利用 NI 的 Vision 实现不碰撞就能避障,优化 iRobot 轨迹。