北京邮电大学实践课程实验报告

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | 创新创业实践 | | 学院 | 计算机学院 | | 指导老师 | 卢向群老师，戴志涛老师 |
| 知识模块 | 智能机器人 | | 完成时间 | | | 2019年4月19日 | |
| 班级 | 班内序号 | 学号 | | | 学生姓名 | 成绩 | |
| 2017211319 | 18 | 2017213508 | | | 蒋雪枫 |  | |

1. 实验目的与实验综述：

我们从广泛意义上理解所谓的智能机器人，它给人的最深刻的印象是一个独特的进行自我控制的“活物”。其实，这个自控“活物”的主要器官并没有像真正的人那样微妙而复杂，只是基于算法而做出动作而已。本次实验中我们做的是“静态”的机器人，只是事先编程给他们指定“静态”的功能，让他们可以完成相应的行为。而目前的“动态”智能机器人更多是基于强化学习的算法，在给定的环境输入做出相应的响应动作。

在本次实验中，我完成了智能机器人的所有实验（包括所有扩展实验），在老师的指导下初步入门了Arduino的硬件编程，花了较多的时间与精力，从中体验到了计算机与硬件的交互与编程的快乐，可以说是有很大收获了。

1. 实验原理与实验过程：

智能机器人部分：

实验一，二：略

实验三：

本次实验我基于可视化“积木”编程与Arduino代码编程完成了所有要求：

1. 让扬声器（蜂鸣器）按照指定的频率发声。

2. 使用串口发送 1-7 分别使扬声器（蜂鸣器）发出 DO-TI 的音调。

3. 使用扬声器（蜂鸣器）发声和延时，播放乐曲。

关于实验3-1，我们让蜂鸣器按指定频率Do,Re,Me,Fa,Sol,La,Si发声：

#define Do 262

#define Re 294

#define Mi 330

#define Fa 350

#define Sol 393

#define La 441

#define Si 495

#define Doo 882

int buzzer=8;

int len=8;

int scale[]={Do,Re,Mi,Fa,Sol,La,Si,Doo};

float duration[]={2,1,1,1,2,1,1,1};

void setup() {

pinMode(buzzer,OUTPUT);

}

void loop() {

for(int i=0;i<len;i++){

tone(buzzer,scale[i]);

delay(250\*duration[i]);

noTone(buzzer);

delay(100);

}

delay(1000);

}

代码较为简单，对此不与赘述。

对于实验3-2：

char \_ABVAR\_1\_input = ' ' ;

void setup()

{

Serial.begin(9600);

}

void loop()

{

if (( ( Serial.available() ) == ( 1 ) ))

{

\_ABVAR\_1\_input = Serial.read();

}

if (( ( \_ABVAR\_1\_input ) == ('1') ))

{

tone(8, 262, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('2') ))

{

tone(8, 294, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('3') ))

{

tone(8, 330, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('4') ))

{

tone(8, 349, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('5') ))

{

tone(8, 392, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('6') ))

{

tone(8, 440, 1000);

}

if (( ( \_ABVAR\_1\_input ) == ('7') ))

{

tone(8, 494, 1000);

}

}

比3-1多的内容只是加了个串口控制，当我们通过串口输入1~7的时候他会发出不同频率的声音。

对于实验3-3：

#define Do 262

#define Re 294

#define Mi 330

#define Fa 350

#define Sol 393

#define La 441

#define Si 495

#define Doo 882

//音名记法

#define C 262

#define D 294

#define E 330

#define F 350

#define G 393

#define A 441

#define B 495

#define CC 525

#define DD 589

#define EE 661

#define AA 882

/\*\*

\* 1=C

\*/

int buzzer=8; //蜂鸣器口，PWM

int scale[]={G,A,EE,A,G,A,G,A,EE,A,G,

A,EE,A,G,A,E,

G,D,E,G,A,B,

A,EE,A,G,A,G,

A,EE,B,CC,B,CC,B,A,E,

D,E,G,A,B,A,EE,A,G,A,

G,A,EE,A,G,A,EE,A,G,A,

E,G,D,E,G,A,B,A,EE,A,G,A,

G,A,EE,B,CC,B,CC,DD,EE,AA}; //旋律

float duration[]={2,1,1,1,1,7,1,1,1,1,1,

1,1,1,1,3,1,

3,1,1,1,1,1,

1,1,1,1,7,1,

1,1,1,1,1,1,1,1,6,

1,1,1,1,1,1,1,1,1,7,

1,1,1,1,1,1,1,1,1,3,

1,3,1,1,1,1,1,1,1,1,1,7,

1,1,1,1,1,1,1,1,1,4}; //音符时值

int len=0; //该变量后面用于统计音符个数

void setup() {

pinMode(buzzer,OUTPUT);

len=sizeof(scale)/sizeof(scale[0]);

}

void loop() {

for(int i=0;i<len;i++){

tone(buzzer,scale[i]);

delay(250\*duration[i]);

noTone(buzzer);

delay(100);

}

delay(1000);

}

我们根据音乐《潮鸣》的简谱，安排了两个数组，一个数组用于表征乐音的频率，一个数组表征乐音的播放时间，在Main函数里面开始“遍历”这两个数组，让他们执行相应功能，这样就构成了一段音乐。

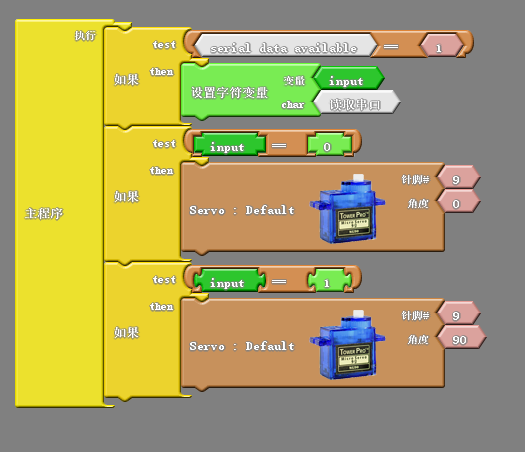
实验四-舵机控制：

本次实验中，我实现了：

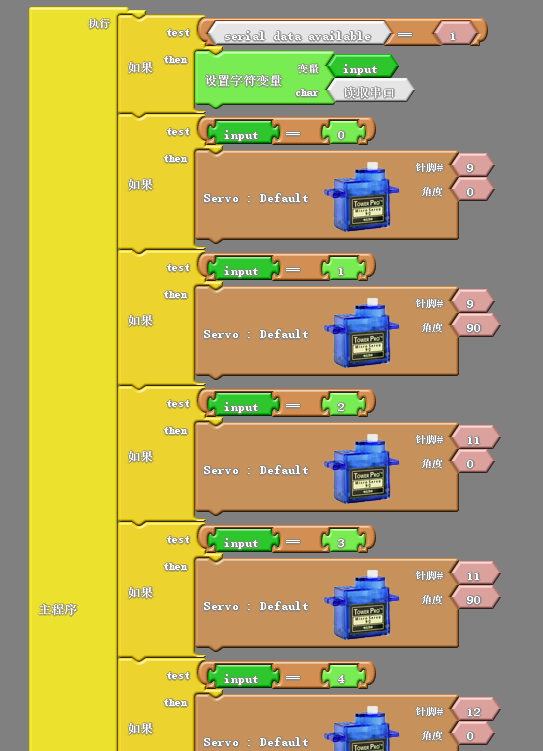
1.使用串口控制机械爪的开合。

1. 通过程序自定义机械臂动作，使机械臂在各个方向运动到最大范围，观察动作和想象中的差距，并分析原因。（个人认为原因是在使用Block编程时，舵机的角度是有限）
2. 编程实现机械臂“跳舞”（机械臂有规律地重复动作）。(在实验四中不予赘述，因为这是一个扩展实验，将在扩展实验中进行说明)。

实验4-1 机械臂开合：



实验4-2，4-3 串口舵机控制：



代码说明：通过串口读取数据，根据输入0~7控制各个舵机的动作。

扩展实验五-机械爪夹取物品：

代码实现：

#include <Servo.h>

char \_ABVAR\_1\_input = ' ' ;

Servo servo\_pin\_12;

Servo servo\_pin\_9;

void setup()

{

Serial.begin(9600);

servo\_pin\_12.attach(12);

servo\_pin\_9.attach(9);

}

void loop()

{

if (( ( Serial.available() ) == ( 1 ) ))

{

\_ABVAR\_1\_input = Serial.read();

}

if (( ( \_ABVAR\_1\_input ) == ('1') ))

{

servo\_pin\_12.write( 90 );

delay( 1000 );

servo\_pin\_9.write( 150 );

delay( 1000 );

servo\_pin\_9.write( 0 );

delay( 1000 );

servo\_pin\_12.write( 160 );

delay( 1000 );

}

if (( ( \_ABVAR\_1\_input ) == ('0') ))

{

servo\_pin\_12.write( 90 );

delay( 1000 );

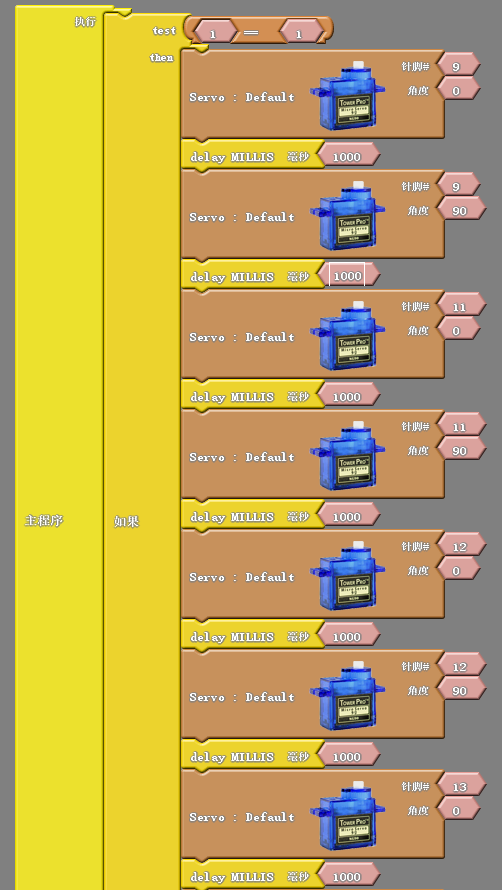
}

}

说明：看起来这么复杂，其实就是让他按照我们的指令完成一系列操作：大臂伸展，机械爪开合，大臂回退，机械爪闭合。

扩展实验六-机器人跳舞：

代码说明：其实就是让他实现自动的一系列操作而已，我这里指定的是机械臂开合，转动0-90°，大臂伸展与闭合以及小臂的伸展与闭合这一系列操作。当然，如果再用点心的话，可以做出更帅气的舞蹈。



1. 实验思考与实验总结

通过本次实验，我依次接触到了计算机与LED灯，蜂鸣器，舵机机器人的交互，每个实验的每个要求的亲自动手实现了一遍，甚至在基础上进行扩展，实验效果在机器人上也较为良好，没出什么意外。

对于实验中出现的问题，印象最深刻的是在做机器人跳舞的时候，最开始设计的block会导致机器人卡住，就是“颤抖”。最后分析了一下，才意识到应该给每个动作加一个“暂停时间”，然后修改后就没什么问题了。

对于实验总结，本次实验拓宽了我的计算机视野，极大地提高了我的实践能力，不论是对于可视化block编程还是Arduino原生类C语言编程，领悟到了通过计算机软件与外部硬件进行“人机交互”的乐趣，同时，老师悉心的讲解与辛苦的备课也是让本次实验做得比较顺利的必备条件，这对于将来的学习一定会很有帮助。