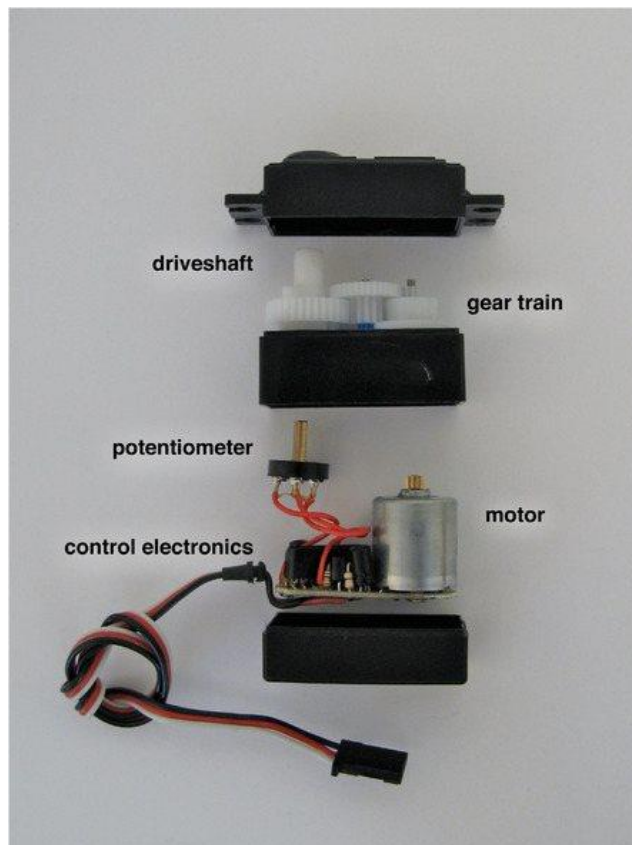


1 我猜你肯定在机器人和电动玩具中见到过这个小东西，至少也听到过它转起来时那与众不同的“吱吱吱”的叫声。对，它就是遥控舵机，常用在机器人技术、电影效果制作和木偶控制当中，不过让人大跌眼镜的是，它竟是为控制玩具汽车和飞机才设计的。

- 舵机的旋转不像普通电机那样只是古板的转圈圈，它可以根据你的指令旋转到 0 至 180 度之间的任意角度然后精准的停下来。如果你想让某个东西按你的想法运动，舵机可是个不错的选择，它控制方便、最易实现，而且种类繁多，总能有一款适合你哟。

- 用不着太复杂的改动，舵机就可摇身一变成为一个高性能的、数字控制的、并且可调速的齿轮电机。在这篇文章中，我会介绍舵机使用的的一些基础知识以及怎样制作一个连续运转舵机。

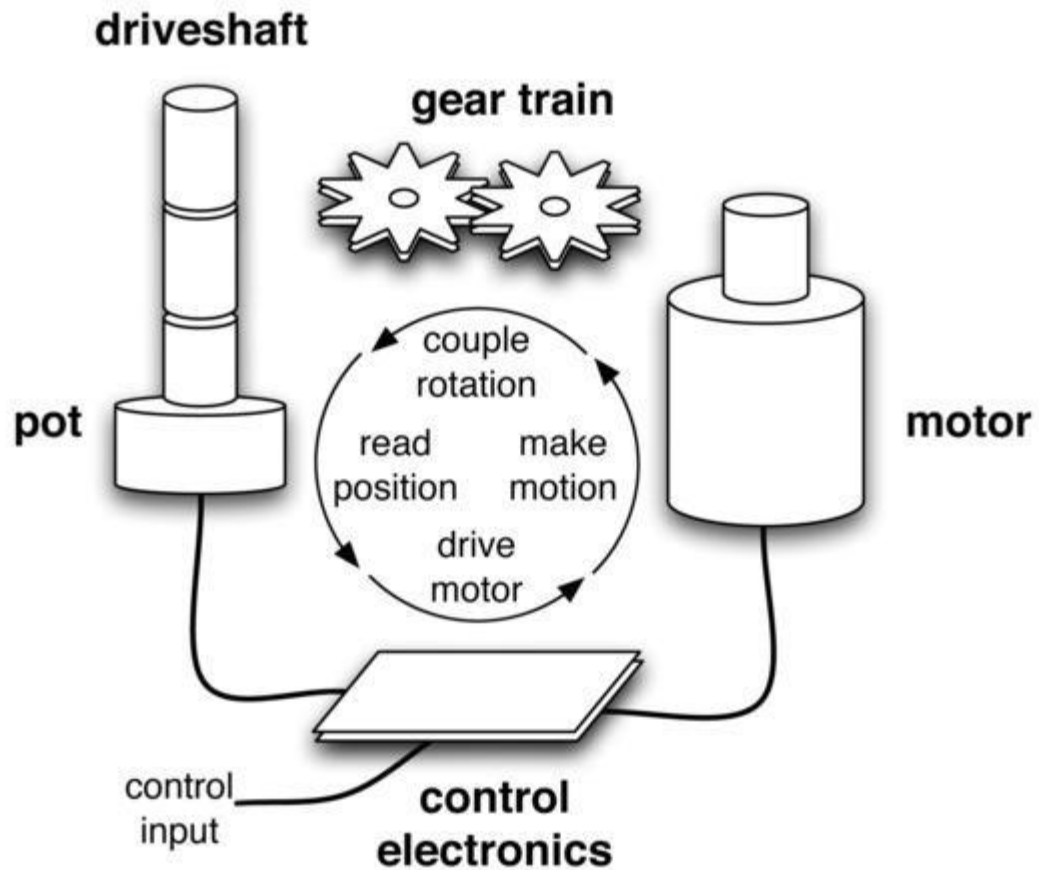
2 舵机的结构和原理



A. 标准舵机图解

- 遥控舵机（或简称舵机）是个糅合了多项技术的科技结晶体，它由直流电机、减速齿轮组、传感器和控制电路组成，是一套自动控制装置，神马叫自动控制呢？所谓自动控制就是用是一个闭环反馈控制回路不断校正输出的偏差，使系统的输出保持恒定。我们在生活中常见的恒温加热系统就是自动控制装置的一个范例，其利用温度传感器检测温度，将温度

作为反馈量，利用加热元件的输出，当温度低于设定值时，加热器启动，温度达到设定值时，加热器关闭，这样不就使温度始终保持恒定了吗。



B. 闭环反馈控制

- 对于舵机而言呢，位置检测器是它的输入传感器，舵机转动的位置一变，位置检测器的电阻值就会跟着变。通过控制电路读取该电阻值的大小，就能根据阻值适当调整电机的速度和方向，使电机向指定角度旋转。图 A 显示的是一个标准舵机的部件分解图。图 B 显示的是舵机闭环反馈控制的工作过程。

3 选择舵机



C.大扭力/微型/标准舵机

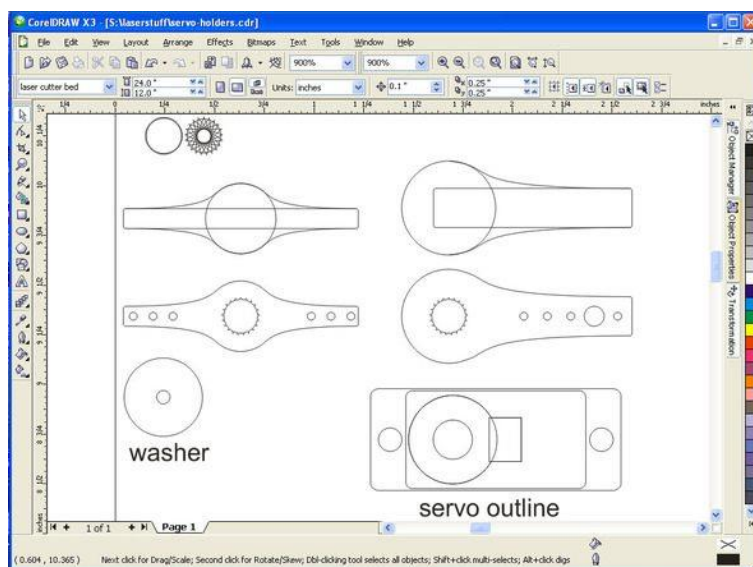
- 舵机的形状和大小多到让人眼花缭乱，但大致可以如图 C 所示分类。最右边身材不错的是常见的标准舵机，中间两个小不点是体积最小的微型舵机，左边的魁梧的那个是体积最大的大扭力舵机。它们都是同样的三线控制，因此你可以根据需求换个大的或小个的。
- 除了大小和重量，舵机还有两个主要的性能指标：扭力和转速，这两个指标由齿轮组和电机所决定。扭力，通俗讲就是舵机有多大的劲儿。在 5V 的电压下，标准舵机的扭力是 5.5 千克/厘米（75 盎司/英寸），转速很容易理解，就是指从一个位置转到另一个位置要多长时间。在 5V 电压下，舵机标准转度是 0.2 秒移动 60 度。总之，和我们人一样，舵机的个子越大，转的就越慢但也越有劲儿。
- 赶快想好你要做的东西，让我们开始动手吧。确定做什么之后，选择哪种大小的舵机（标准型、微型、绞盘型）就是小 case 了，你可以绅士般的从中选个最便宜的。在这个项目中，我选的就是微型系列的 HexTronik 公司生产的 HXT500 型舵机，额定数值是扭力 0.8 千克，转速 0.10 秒，只花不到 4 美元就搞定了。

4 舵机的支架和连接装置

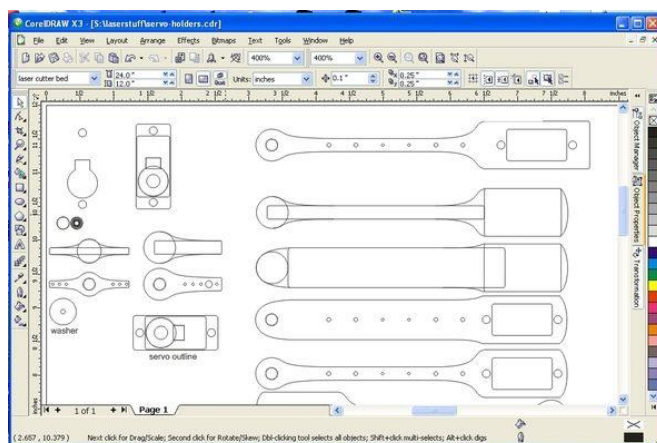


D.多种舵盘

- 想在你的项目中用上舵机，就要满足两个条件：一是需要个能把舵机固定到基座上的支架，二是得有个能将驱动轴和物体连在一起的连接装置。支架一般舵机上就有，而且带有拧螺丝用的安装孔。如果你仅仅是测试的话，用点儿热熔胶或者双面泡沫胶带就能轻松的固定住舵机。
- 怎样连接驱动轴呢，你会发现舵机都附带了一些有孔的小东西，这就是舵盘，它可以套在驱动轴，臂上打上了些小孔。你只要用连接棒或者线把物体连到孔上，就可以将舵机的旋转运动变成物体的直线运动了，当然了，选用不同的舵盘或固定孔就能产生不同的运动啦。
- 图示的是几种不同的舵盘。前面 4 个白色的是舵机附带的舵盘，右边四个是用激光切割机切割塑料得到的 DIY 舵盘。最右边的 2 个是舵盘和支架的组合，如果你想实现两个舵机的组合运动，把这个舵盘的支架固定到另一个舵机的支架上就 OK 了。



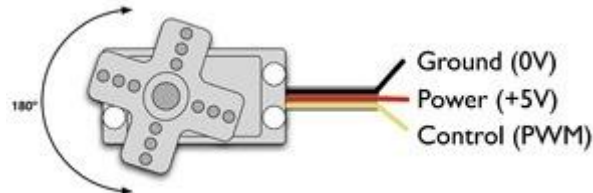
E. 普通舵盘设计



F.其他舵盘

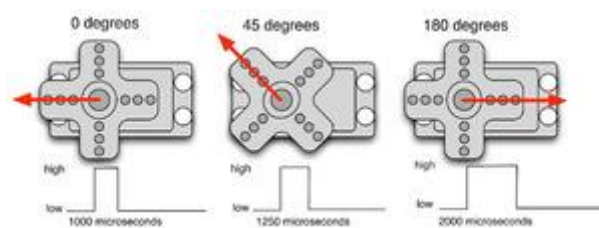
- 制作普通舵盘对于童鞋们来说是比较容易的，先用矢量作图软件画一个多边形，这个多边形的半径和顶点数都要和舵机驱动轴匹配，这样它就能连接到驱动轴上了，其他种类的也是这样画出来的。

5 如何控制舵机



G.3 线接口

- 像图所示那样，舵机有一个三线的接口。黑色（或棕色）的线是接地线，红线接+5V 电压，黄线（或是白色或橙色）接控制信号端。

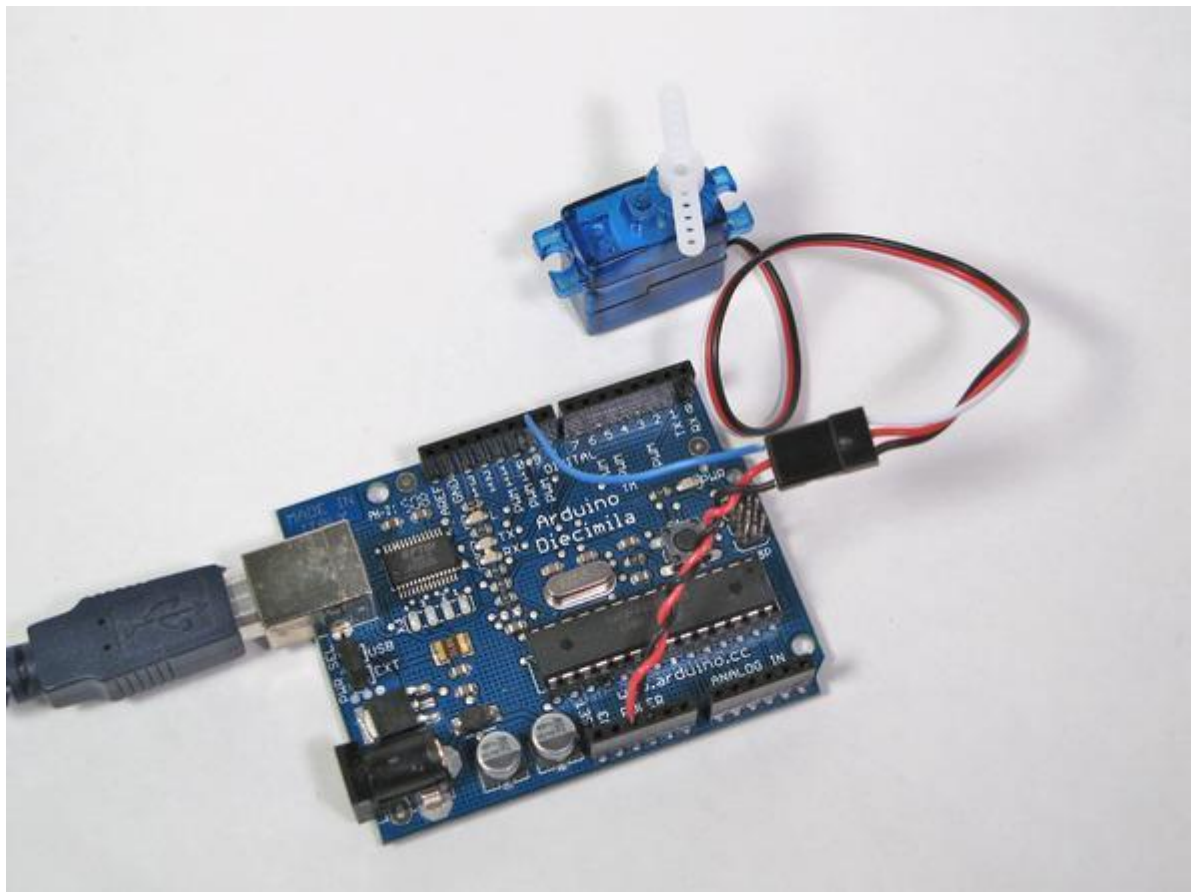


H.控制信号

- 控制信号（如图 H）是一种脉宽调制（PWM）信号，凡是微控制器能轻松的产生这种信号。在此文中，我用的是常用的 Arduino 开发环境下的微控制器。
- 脉冲的高电平持续 1 到 2 毫秒（ms），也就是 1000 到 2000 微秒(μ s)。在 1000 μ s 时，舵机左满舵。在 2000 μ s 时，右满舵。不过你可以通过调整脉宽来实现更大或者更小范围内的运动。
- 控制脉冲的低电平持续 20 毫秒。每经过 20 毫秒（50 次每秒），就要再次跳变为高电平，否则舵机就可能罢工，难以保持稳定。不过你要是想让它一瘸一拐的跳舞，倒可以采取这种方法。

```
int servoPin = 9;↵
↵
int servoPosition = 1500; // position in microseconds↵
↵
void setup() {↵
    pinMode(servoPin, OUTPUT);↵
}↵
↵
void loop() {↵
    digitalWrite(servoPin, HIGH);↵
    delayMicroseconds(servoPosition);↵
    digitalWrite(servoPin, LOW);↵
    delay(20); // wait 20 milliseconds↵
}↵
```

这是一个完整的 Arduino 设计程序，在这个程序下，舵机始终在正中间位置，控制起来很容易



I. 舵机连接 Arduino 实验板

- 红色和黑色的线分别接到 Arduino 开发板的 5V 电源脚和接地脚上。控制线接到 Arduino 开发板的数字输入/输出脚 9 脚上。
- 用 Arduino 控制舵机也有不太给力的地方，就是 Arduino 程序把绝大部分时间都浪费在等待延迟命令上，不过童鞋们暂时不要失望，Arduino 中内置有舵机函数，你可以用它内置的计数器来同时控制两个舵机（分别在 9 脚和 10 脚），是不是又豁然开朗了，这样我们就不能把节省下的编程代码干别的事情了吗。

```
#include <Servo.h>

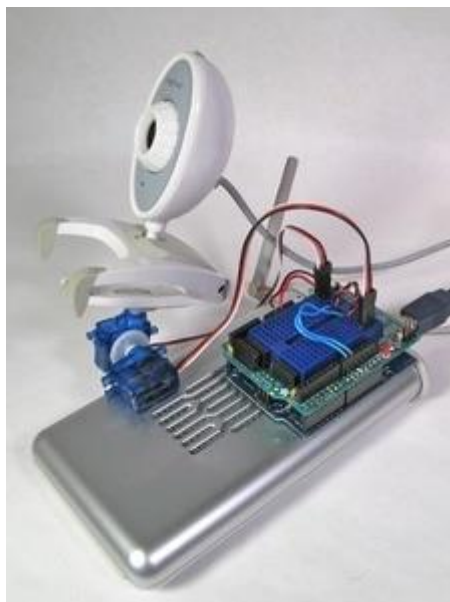
Servo myservo;

void setup() {
    myservo.attach(9); // servo is on pin 9 like before
    myservo.write(90); // set servo to 90 degree position
}

void loop() {
    // free to do anything, our servo is still being driven for us
}
```

这是一个调用了舵机函数的程序

6 舵机应用：云台网络摄像头



J.舵机控制的云台网络摄像头

- 看了这么多内容了，是不是有点迫不及待练练手的冲动，那就先来个简单的，材料就是下面这些，两个舵机、一个 Arduino 板、一个用来装摄像头的可转动基座。先用热胶把第一个舵机的舵盘固定到摄像头的底部，然后把第二个舵机固定到基座上，同时把它的舵盘固定到第一个舵机的一侧，最后把舵盘套到各自舵机上，哇塞，一个云台网络摄像头就这样诞生了。
- 图中是一个纯手工打造的云台网络摄像机，它用的是 OpenWrt Linux 系统的华硕 wi-fi 路由器。
- 网络摄像头和 Arduino 控制板都是用 USB 集线器连接到路由器上的。

```
#include <Servo.h>

Servo servoPan;

Servo servoTilt;

void setup() {

    servoPan.attach(9); // pan servo is on pin 9
    servoTilt.attach(10); // tilt servo is on pin 10
    servoPan.write(90); // home both servos to center
    servoTilt.write(90); // home both servos to center
}

void loop() {

    if( Serial.available() >= 2 ) { // two bytes waiting for us
        int pan = Serial.read(); // 1st byte is Pan position
        int tilt = Serial.read(); // 2nd byte is Tilt position
        servoPan.write(pan); // move pan servo
        servoTilt.write(tilt); // move tilt servo
    }
}
```

通过 Arduino 的 USB 口同时控制两个舵机的程序

- 大致的流程是这样滴，当串口上有两个字节到来时，程序开始工作，赋给第一个字节 0-180 的值，让它调节摇摆舵机（调左右），同样赋给第二个字节 0-180 的值，让它调节倾斜舵机（调上下）。

7 如何 DIY 连续旋转的舵机



K.舵机的内部“解剖”结构

- 任何舵机都能变成一个双向、可调速的降速齿轮电机。通常情况下，需要驱动芯片和其他一些零件才能控制电机的转速和方向，这些部件舵机中都会附带，所以要想得到一个用到机器人上的数控连续旋转舵机，最简单也最便宜的的方法就是自己动手改造一个，哈哈，考验动手能力的时候又来了。



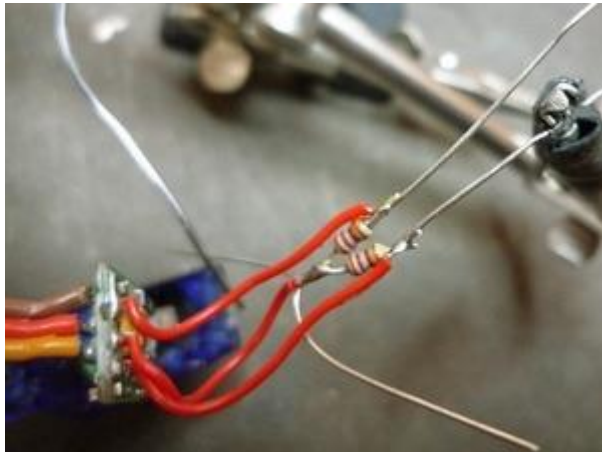
L.拿掉金属挡板

- 需要改动的是部分的电路模块和机械模块，电路模块中，我们要找两个阻值相同的电阻来充当电位计，机械模块中，则要去掉防止电机超速的挡板。



M. 卸下塑料挡板

- 下面我们就开始吧，首先，卸开舵机外壳，HTX500 舵机的外壳由 3 个塑料部分扣在一起。你可以用个小一字改锥或是类似的片状工具把他撬开，然后从轴上取下齿轮组,(记得标记好各个小齿轮的位置哦)，再从下面小心的取出舵机的电路板。
- 舵机上有两个机械制动挡板，用尖嘴钳卸下驱动轴基座上的金属挡板（图 L），用斜嘴钳卸下外壳顶部的塑料挡板（图 M）。



N. 焊上电阻



O. 缠上胶带

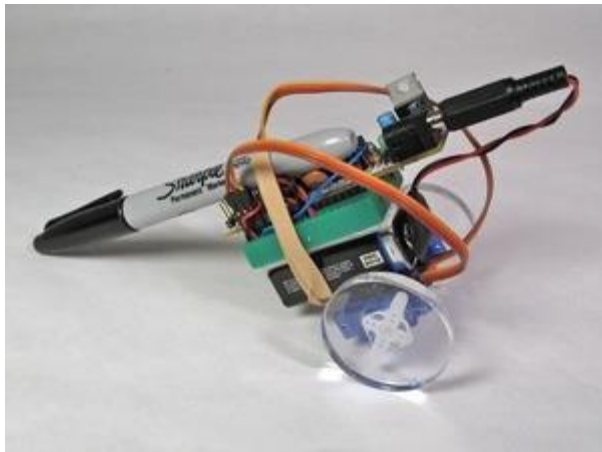
- 用两个阻值相加约 $5\text{k}\Omega$ 的电阻来替代 $5\text{k}\Omega$ 的电位计，实际制作中，选一对 $2.2\text{k}\Omega$ 的电阻就能满足要求了。把电位计上的 3 根线焊下来，像图 N 那样焊到电阻上。再把这个重新组装成的家伙用绝缘胶带或是绝缘管缠好（图 O），最后再和电路板一起重新塞进舵机外壳中，扣好外壳，一个改造好的舵机就呈现在我们面前了。

- 手工制作阶段到此就结束了，但是现在还能高兴的太早，因为只有找到基准点才能算是大功告成。在理想条件下，如果两个电阻完全相同，舵机就能精确的停到 90° 的位置上。不过呢，理想和现实总是会差那么一点点，因此舵机就没像理想中那样么精确。为了使舵机控制更精确，我们要找到一个基准点，方法是把上面编的程序灌进电路中，通过实验来看舵机究竟停在哪个角度，这个角度每个舵机都不相同，所以得出结果后要记录下来。

- 我们业余爱好者常用的舵机一般是用电位计来检测驱动轴转动到的角度，而用在工业机器人、电脑数控机床等大型系统中的舵机一般则要用旋转编码器来确定位置。光学旋转编码器的原理是这样的，把一个带有窄缝的圆盘固定在转轴上，然后用一个 LED 灯和一个光敏元件来记录光通过窄缝照到光敏器件上的次数来计算当前旋转到的位置。其实生活中这种技术也很常见，我们每天都要用的光电鼠标就是用的这个原理制作成的。

注：如果你不想撬开你心爱的舵机，Parallax 公司（BASIC Stamp 微处理器的制造商）有一款即用型，标准尺寸的连续转动舵机可供你使用。

8 连续旋转舵机的应用：5 分钟的绘图机器人



P. 安装好的绘图机器人*

- 想做个会画画的机器人吗，那就去找两个连续旋转舵机来吧，我们这就开始。图 O 这个绘图机器人中包含了舵机两个，9V 电池，面包板，Arduino 电路板，三福记号笔各一个，外加一对塑料轮子。

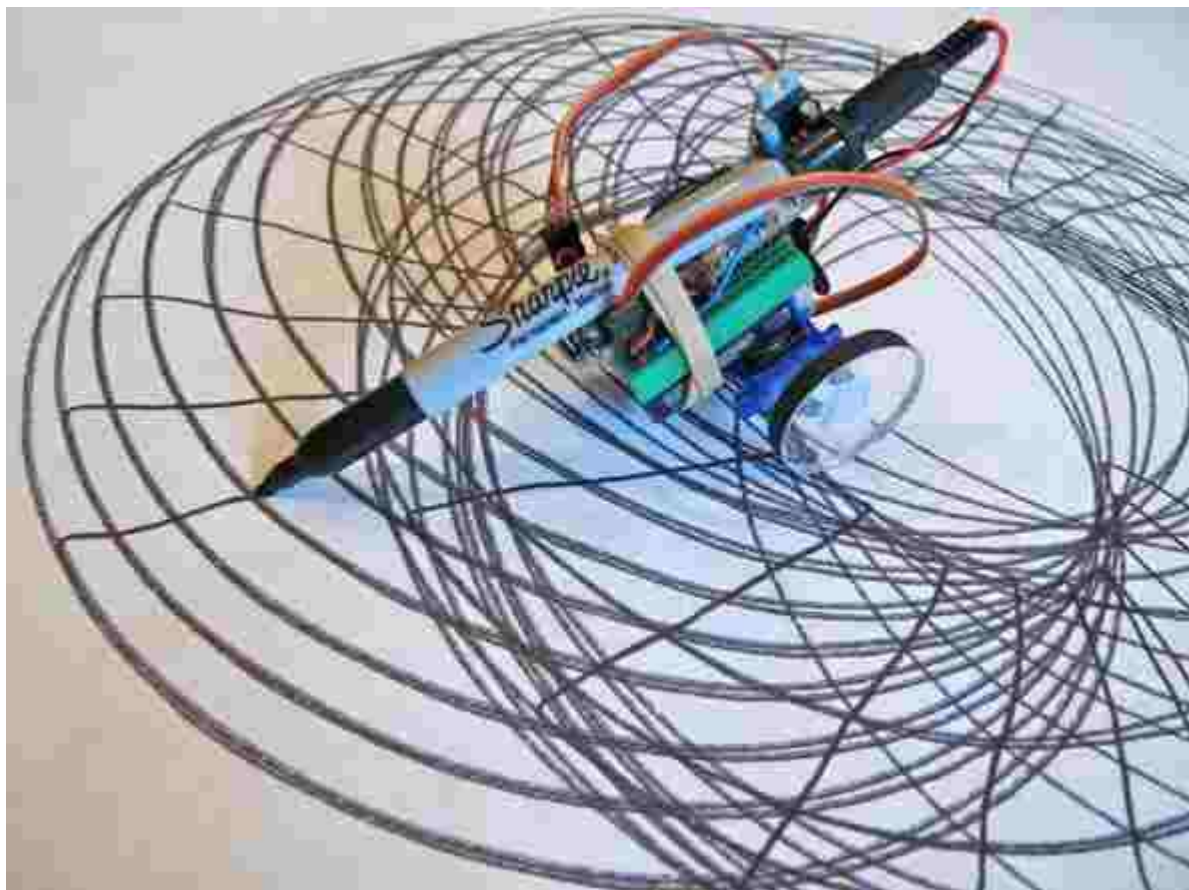
- 它的电路和云台摄像头一样，我们直接拿来用，而且它的部件都可以用热胶粘到一起。关于轮子的选择，更是简单，只要是直径在 1 到 3 英寸的圆东西都能用，比如塑料瓶盖之类的。为了减小摩擦，增大牵引力，我们在车轮上缠上塑料胶带。

- 这样组装阶段就完成了。接下来就是程序了，它的程序用一个包含基准点的变量来制动舵机，这个基准点我们上面已经通过实验测出（你的基准点可能不同）。程序的控制流程

为，先让一个舵机朝一个方向运动一段时间，然后换成另一个舵机转动，这样就能得到一个螺线形的图画了。

- 代码在此：

```
#include
Servo servoL;
Servo servoR;
int servoLZero = 83; // experimentally found to stop L motor
int servoRZero = 91; // experimentally found to stop R motor
boolean turnleft = false;
void setup() {
servoL.attach(9);
servoR.attach(10);
servoL.write(servoLZero); // start out not moving
servoR.write(servoRZero); // start out not moving
}
void loop() {
turnleft = !turnleft;
if( turnleft ) {
servoL.write( servoLZero - 10 );
servoR.write( servoRZero );
delay(1000);
} else {
servoL.write( servoLZero );
servoR.write( servoRZero + 10 );
delay(4000); // turn more one way than the other
}
}
```



Q.运动中的绘图机器人

- 注意：永久记号笔画的痕迹不好清除，童鞋们千万小心哈，最好让绘图机器人在硬纸板或其他不透水的纸的画画，或者索性换成支水溶性的记号笔。