

# 欢迎使用

## BOX Plotclock 2021

### 安装调试说明

# 1.零件



## 螺丝清单

活动关节/机架/  
铜柱/底座的固定  
S023005



X 14



固定Arduino主板  
S063004



X 4



Arduino主板铜柱  
T3006



X 4



舵机固定螺丝  
S042308



X 6



舵机主轴固定螺丝  
S042005



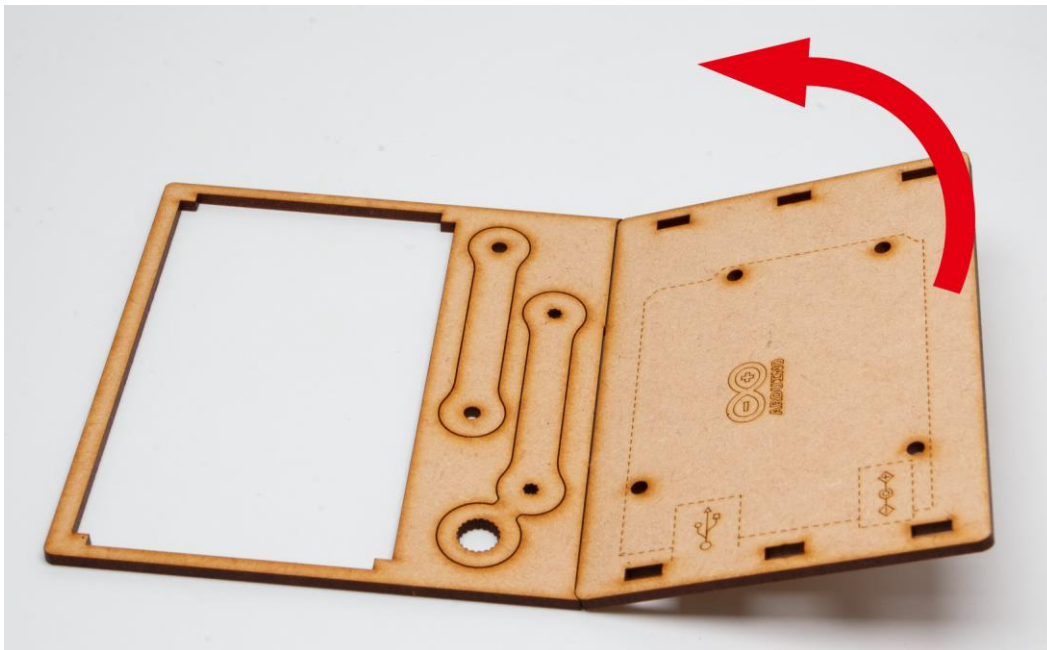
X 3



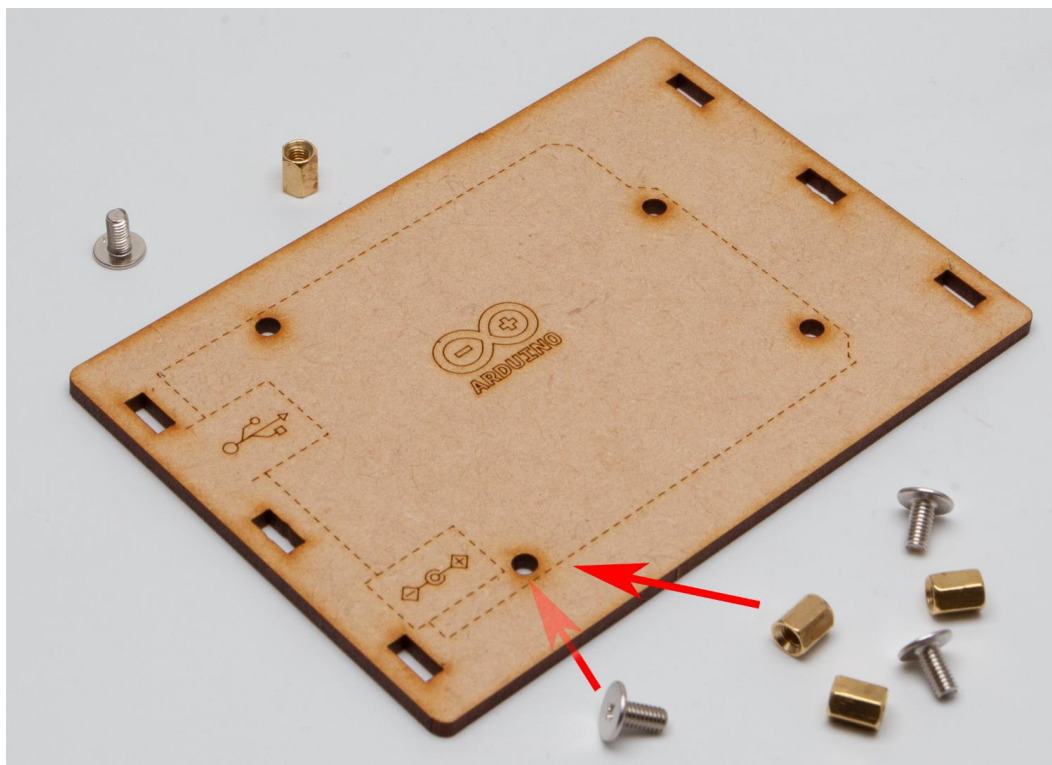
附件包内有多种型号的螺丝，请务必区分清楚再动工。

---

## 2.开始组装

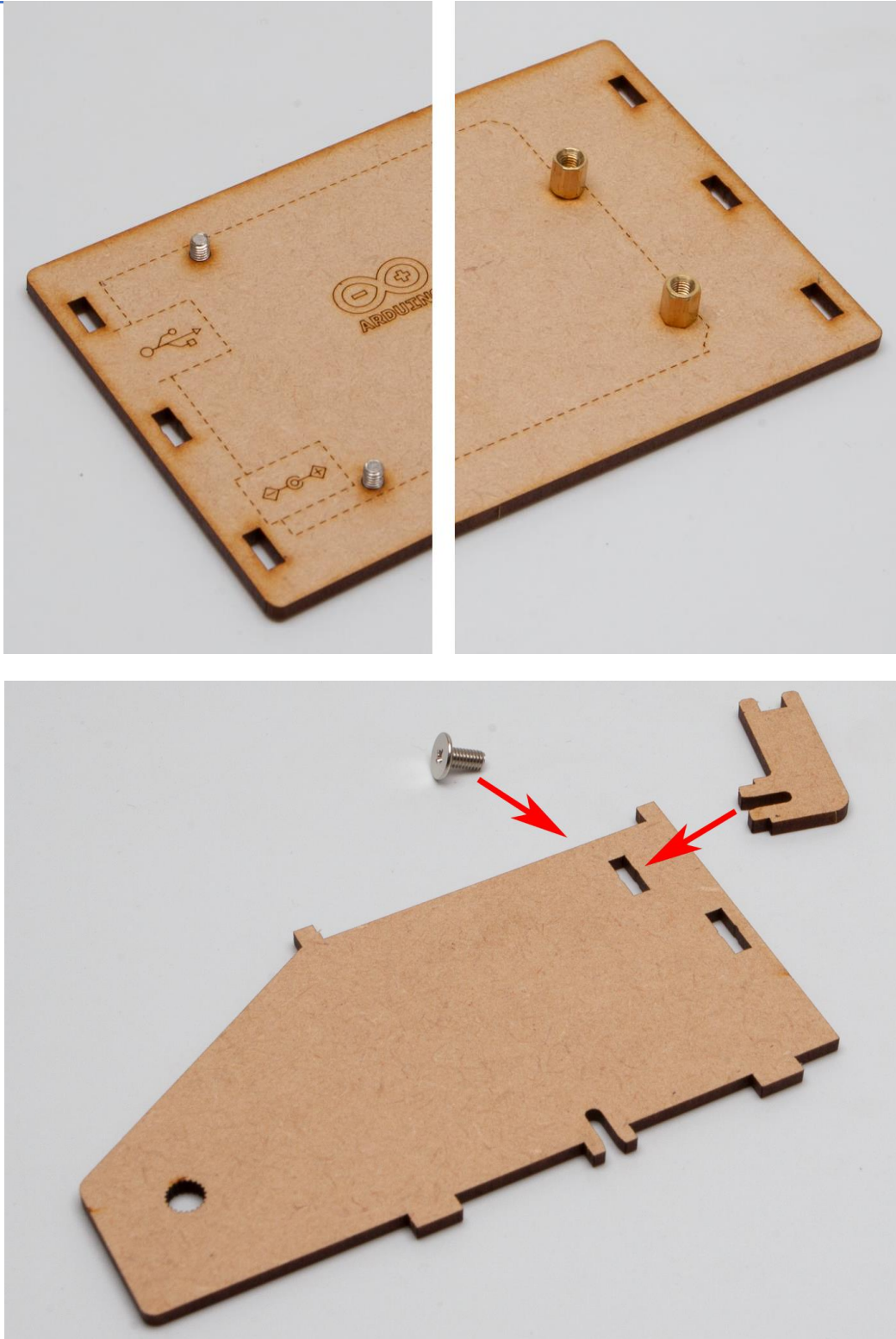


将底座从基板上拆下来。

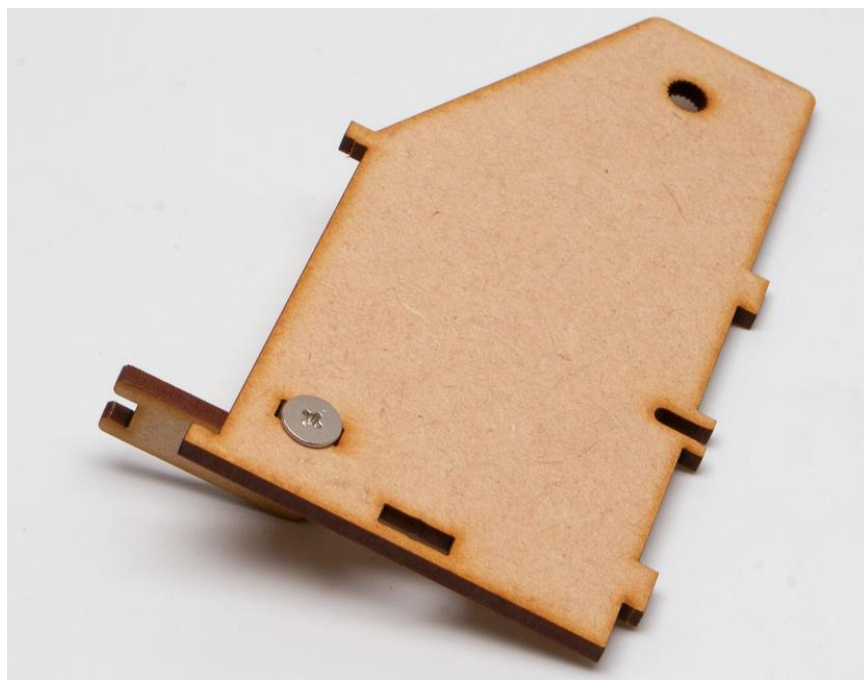
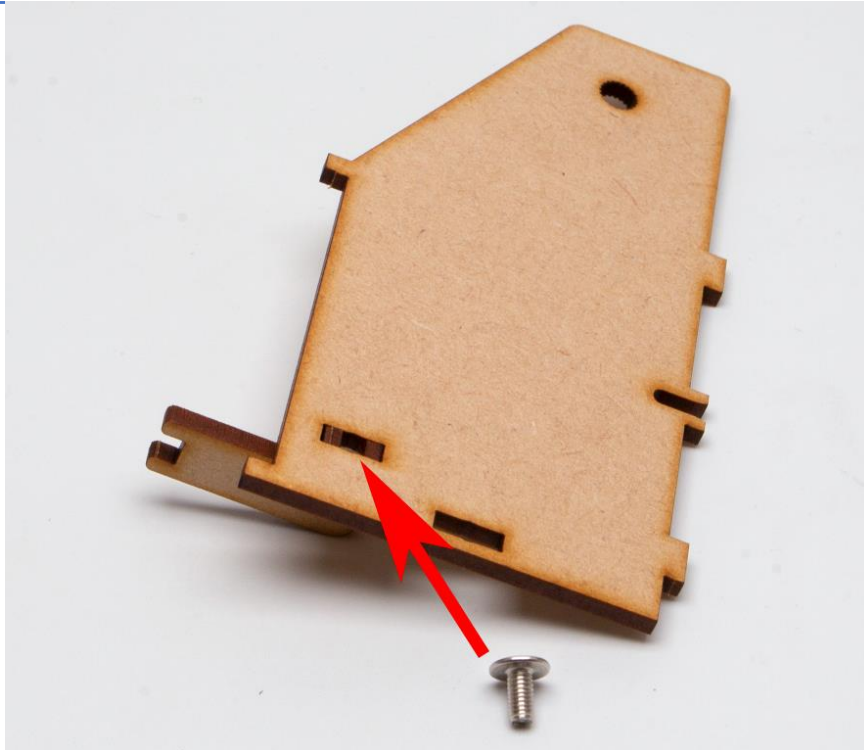


用大帽螺丝从下面固定铜柱

---



在右侧的立板上安装笔擦支架，并用大帽螺丝固定

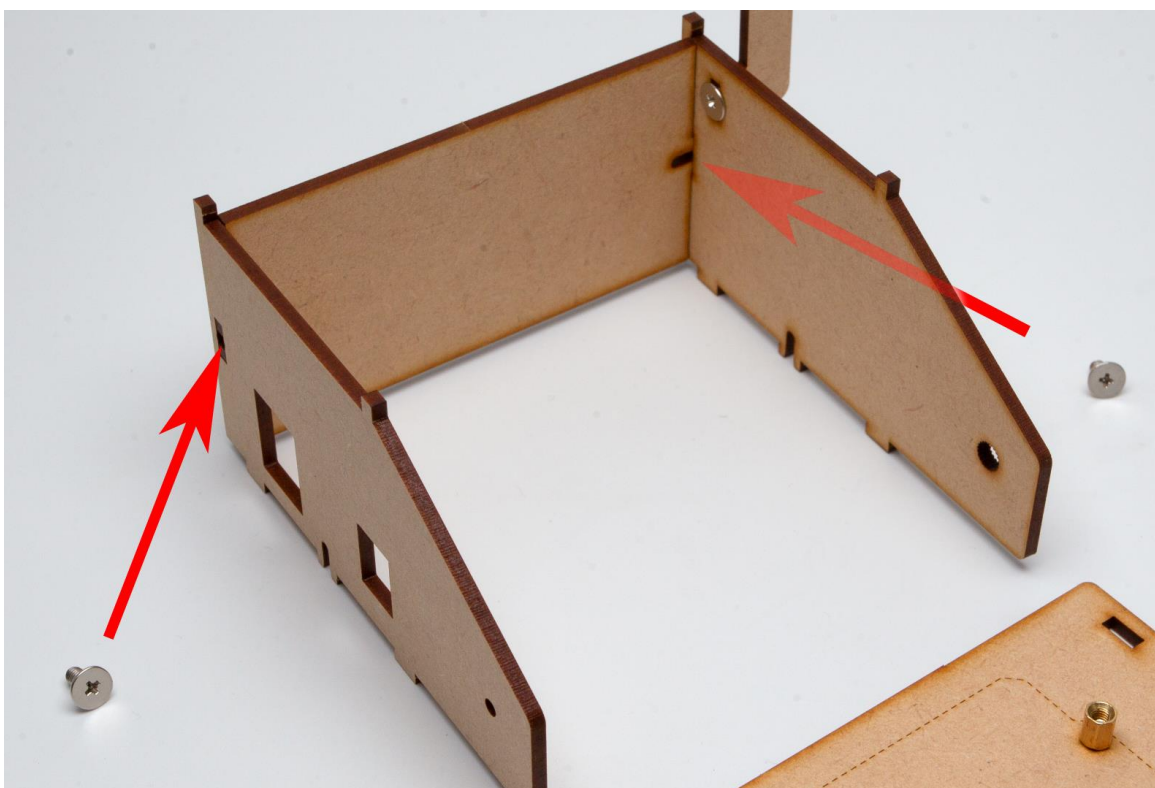


笔架需要先固定好，否则后面会妨碍操作

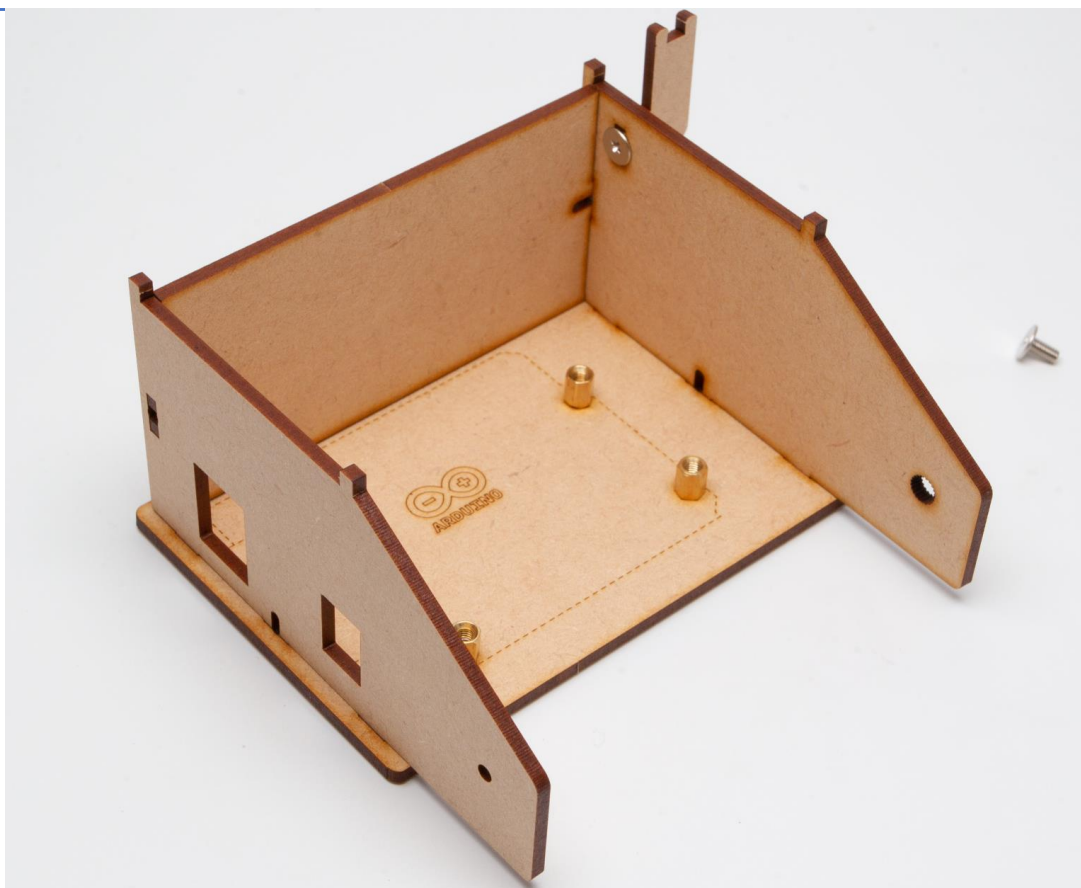




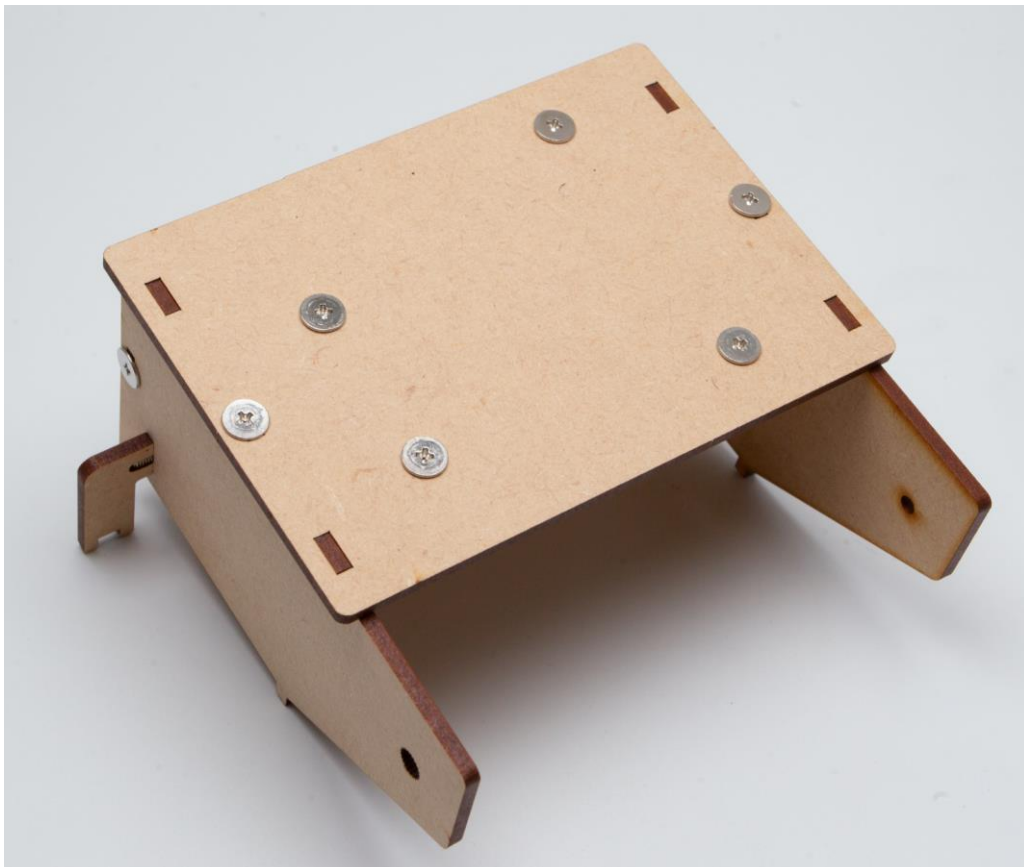
组装主体结构



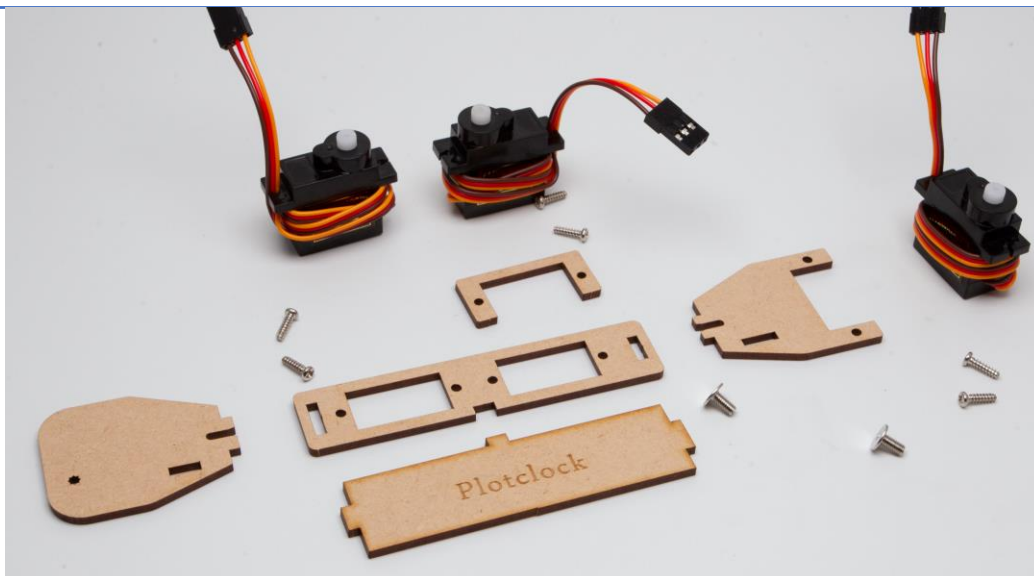
先组装 3 块立板，再整体插到底座板上



立板两侧，底板下面各需要 2 个大帽螺丝固定







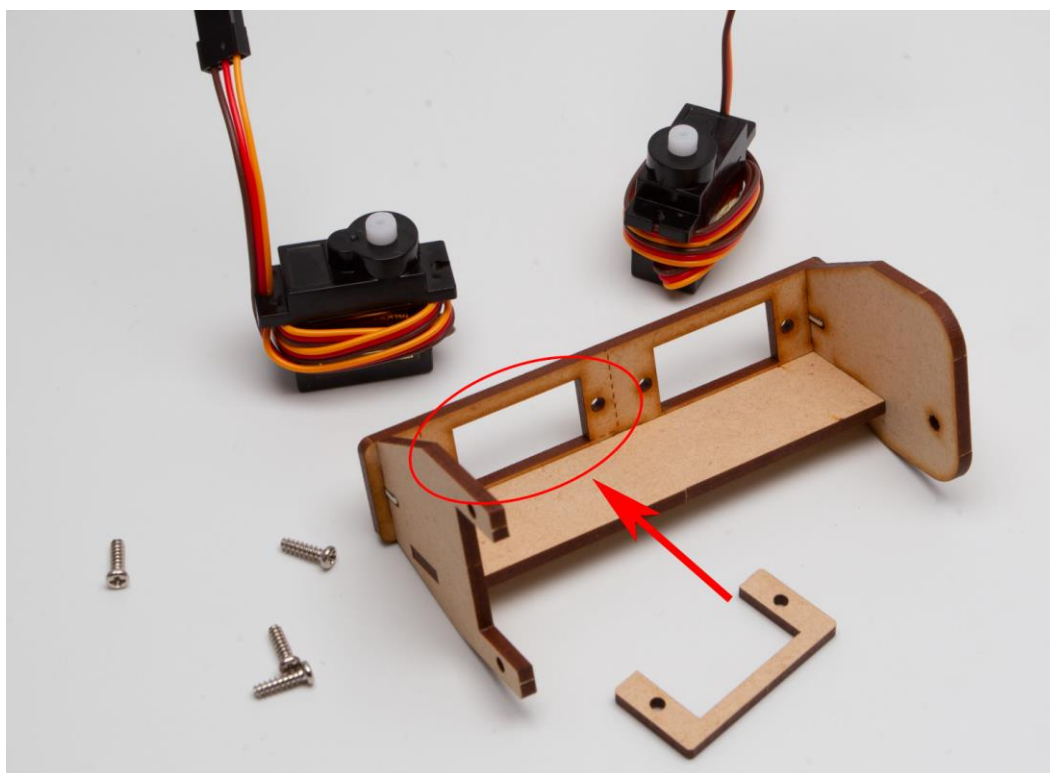
组装摆臂的龙门架架构。零件的榫卯结构都是一一对应的，为防错每个尺寸都不同。如果插不上或者很松，那是零件错了，请仔细检查。



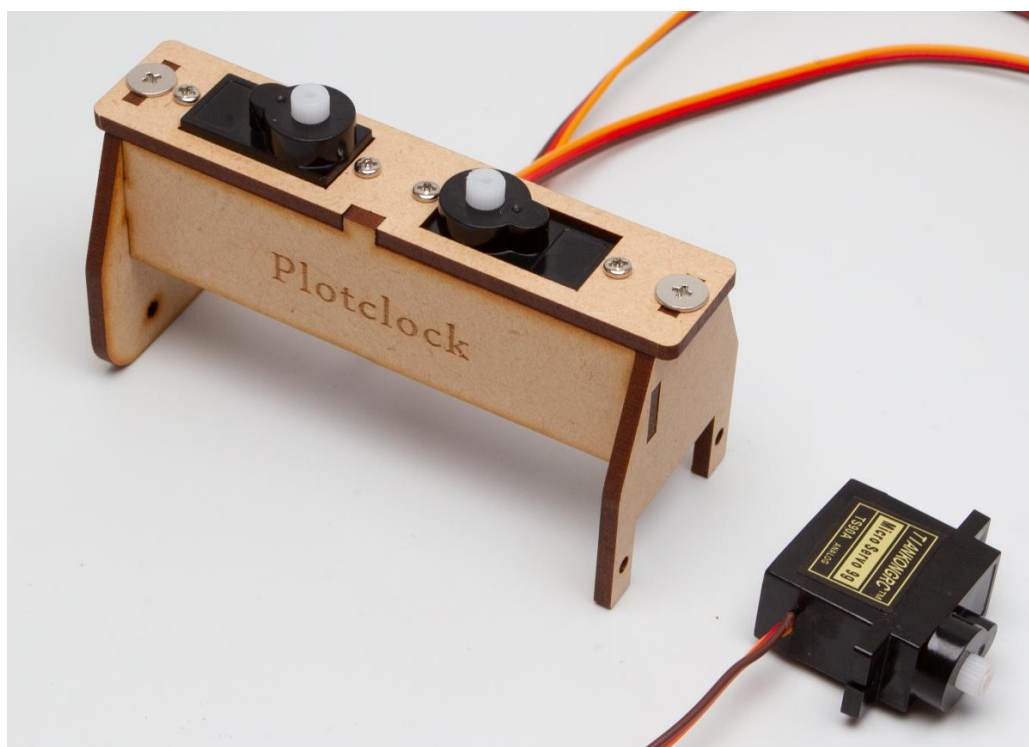
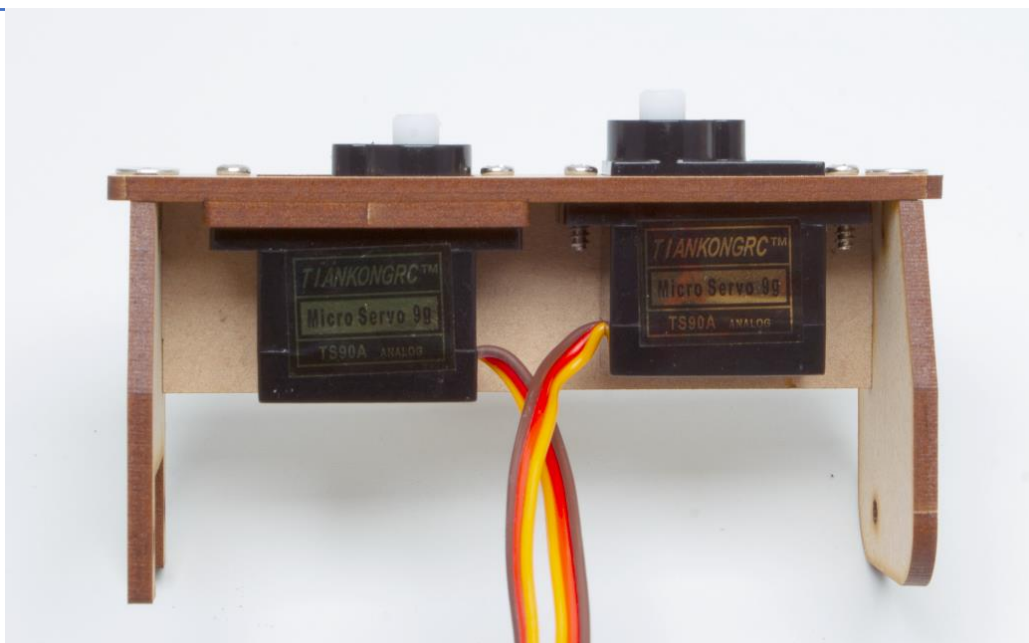
先安装横梁和左右立板



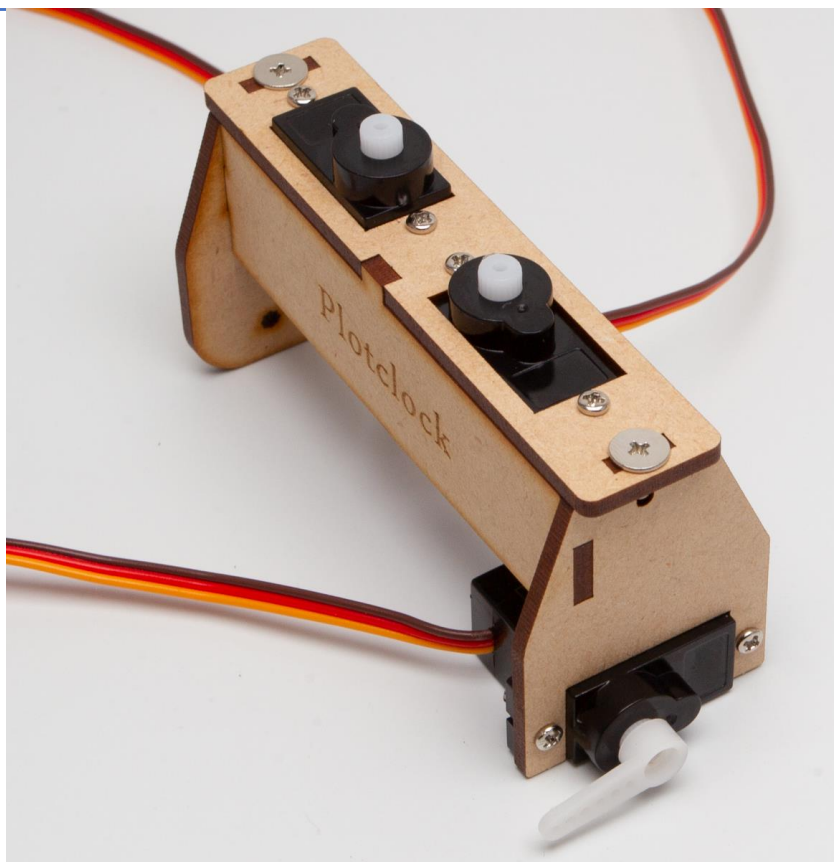
再安装上面板，并用 2 个大帽螺丝固定



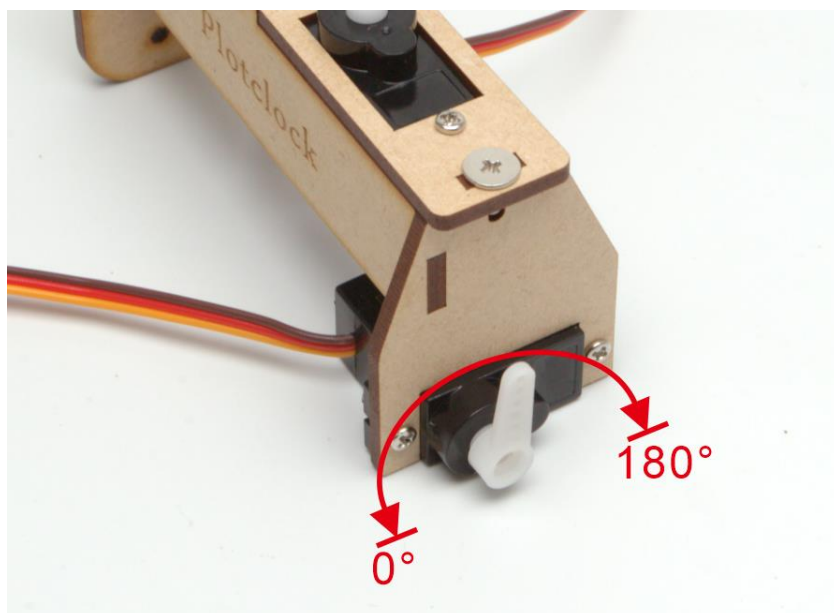
安装两个舵机，注意舵机垫片要放在有标记的一侧，两个舵机高度不同



安装侧面的抬笔舵机，注意舵机的方向，立板内侧有舵机主轴位置的标记。



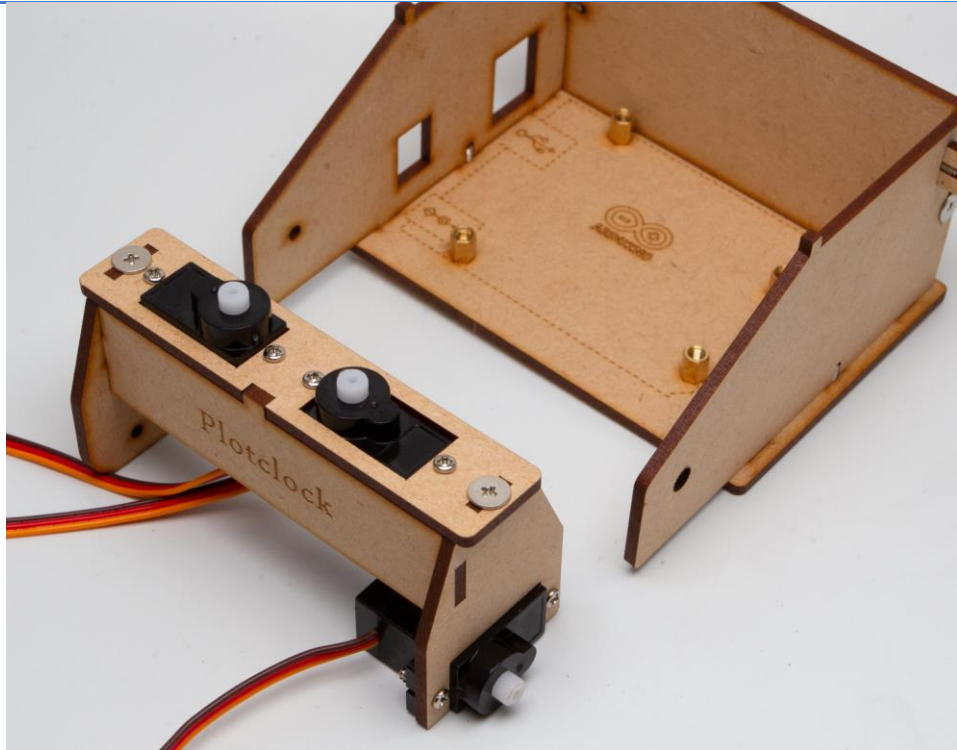
安装一只舵机摆臂，大致调节一下舵机主轴的位置。



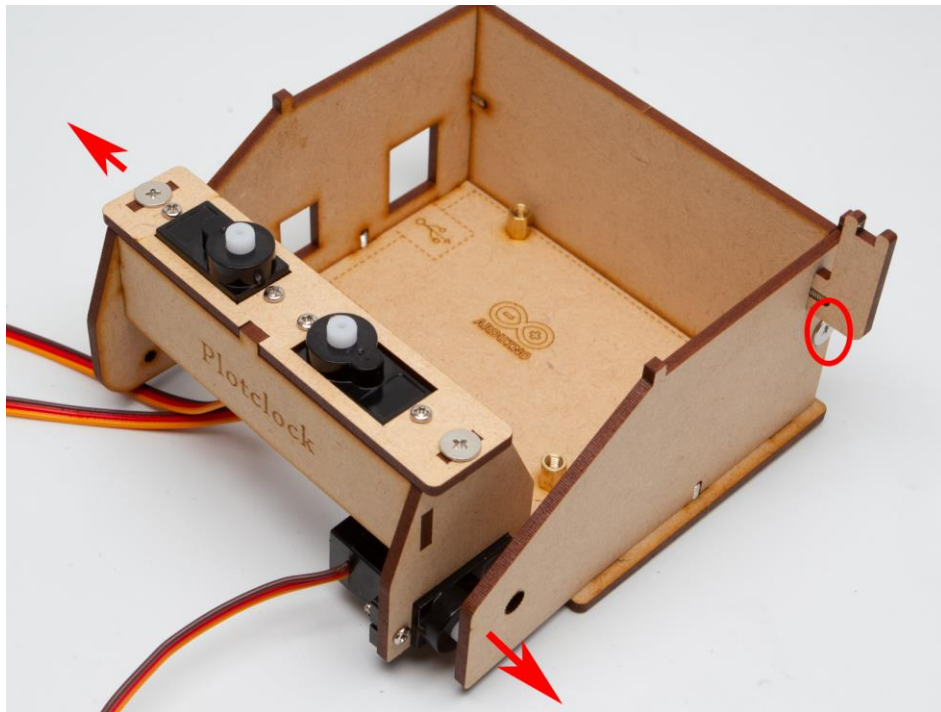
让舵机主轴停留在大致中心的位置，方便调节程序。舵机主轴只能转动 180 度左右，如果转不动不要用力猛掰，可以通电让舵机自由活动后，断电再掰动主轴。不可通电掰主轴。

---





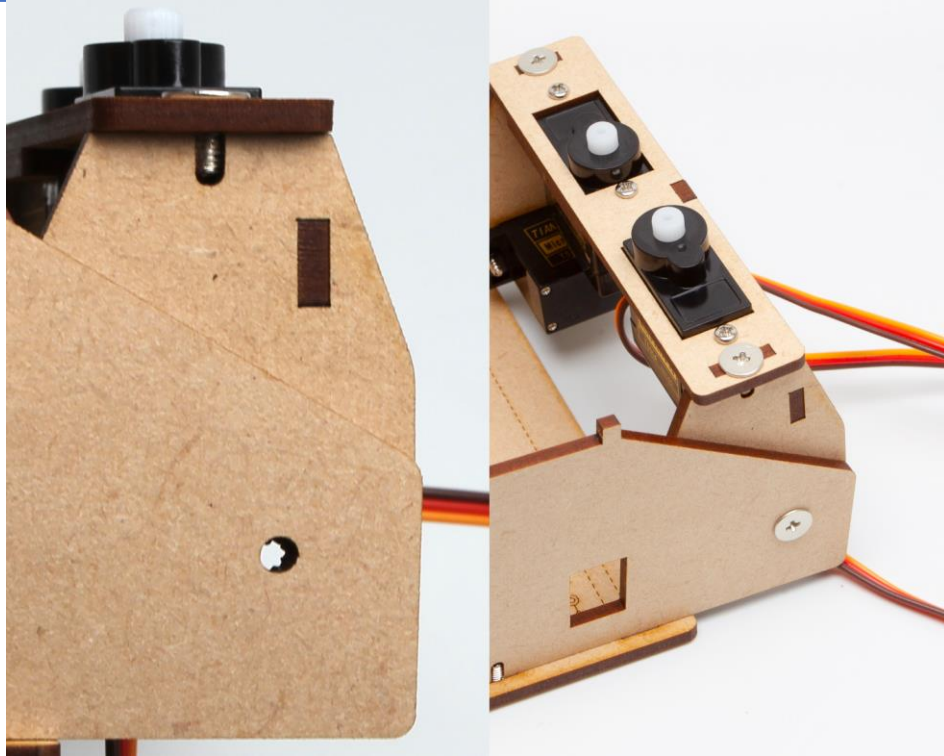
摘掉舵机摆臂（尽量保持主轴不要再转动），将摆臂龙门装入底座。



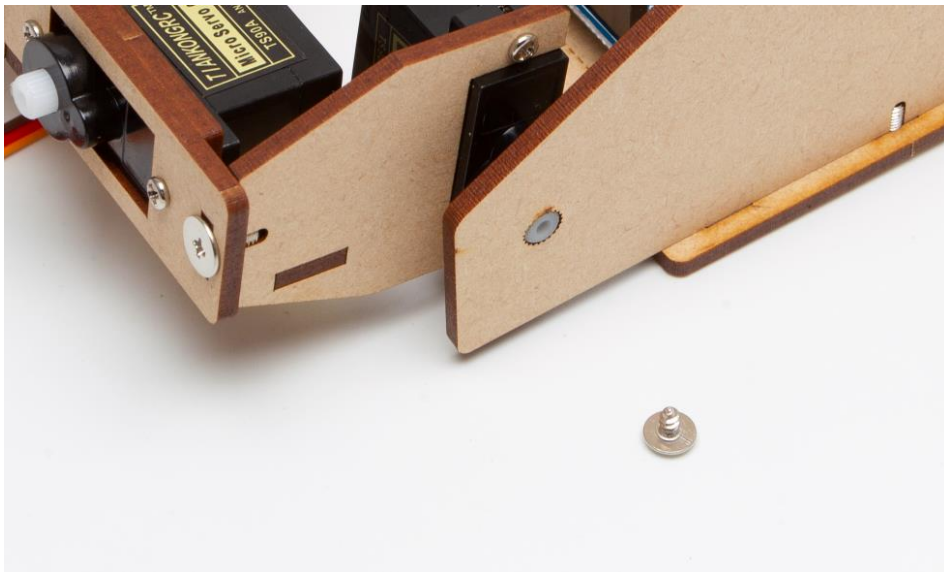
将舵机主轴插入立板的孔中。在插入之前，立板会轻微变形，如果操作有困难可以先拆下笔架旁边的立板固定螺丝，防止变形过大损坏零件。

---





左侧螺丝孔对准后，用大帽螺丝固定，但不能完全拧紧，让龙门架可以自由活动。



右侧用粗牙带帽螺丝固定在舵机主轴上，此处可以拧紧。

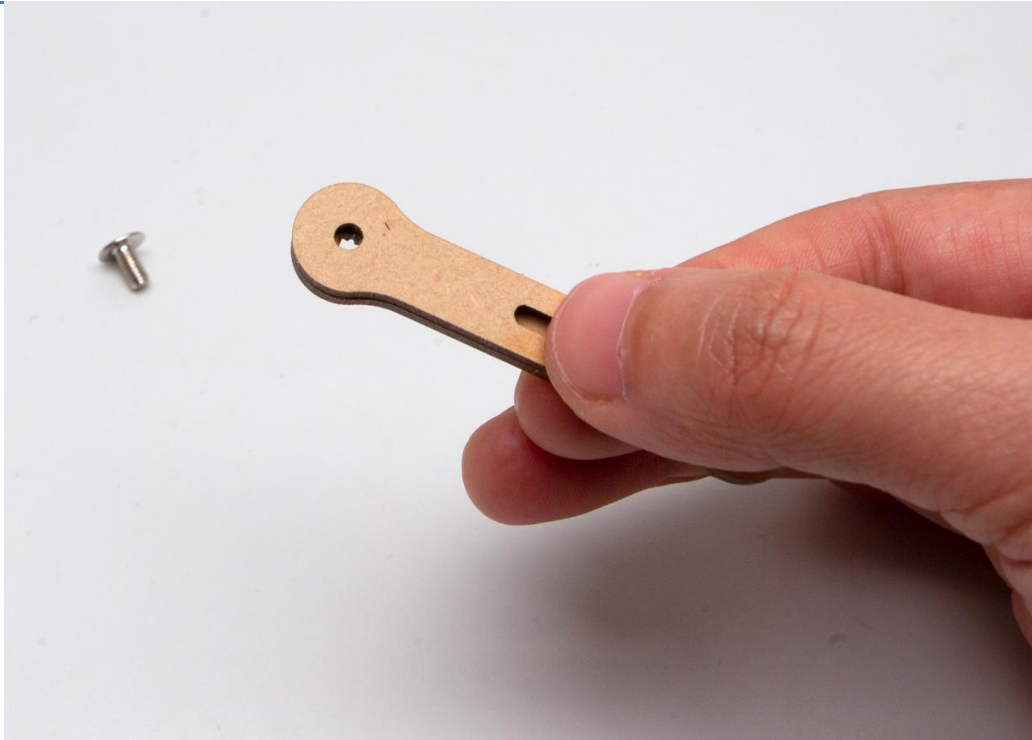


组装写字摆臂，每个零件都不同，且有固定位置，安装时要将有字的一面朝下。



注意各零件的位置关系。L 在上，插笔的长臂在下，R 在下。

---



拧螺丝之前，先将两个孔对齐，固定住两零件，再拧螺丝。

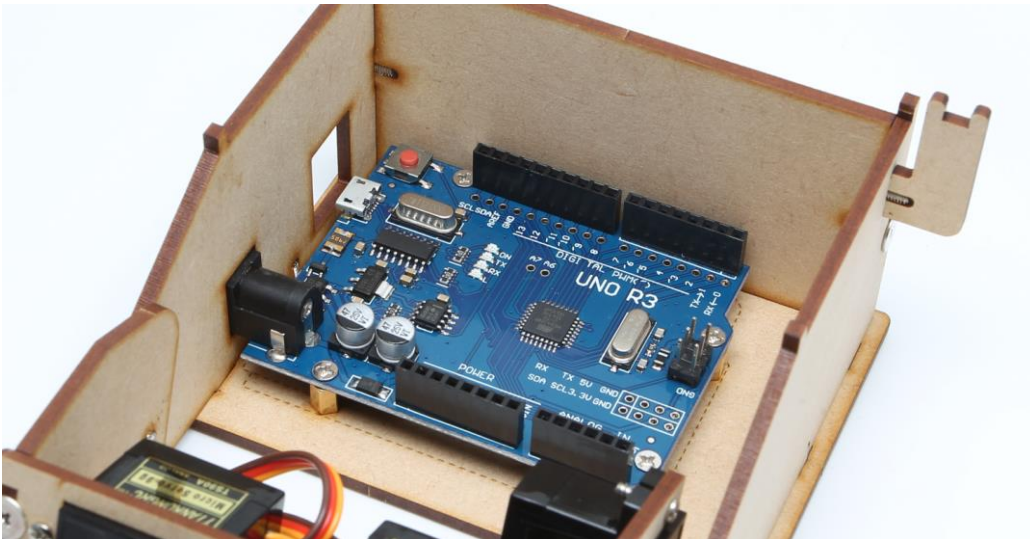


此处螺丝必须要垂直拧入，**切记不要拧歪。不要完全拧紧，让零件可以自由活动。**先将短臂和长臂固定，最后再将两个长臂固定在一起。

---

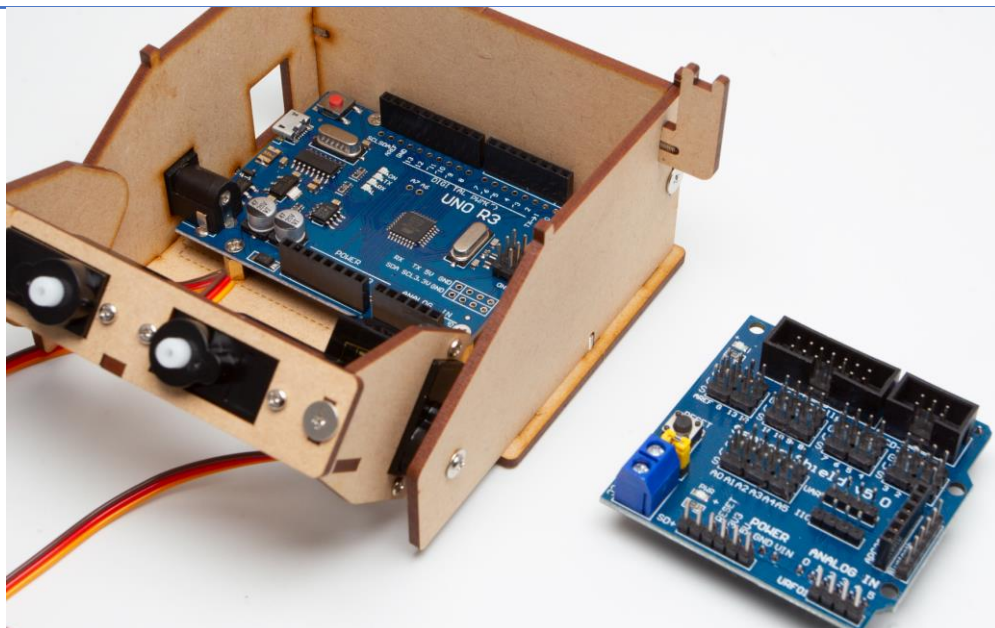


组装好的摆臂，先不要着急安装到舵机上。需要先用程序确定舵机的位置再安装摆臂。

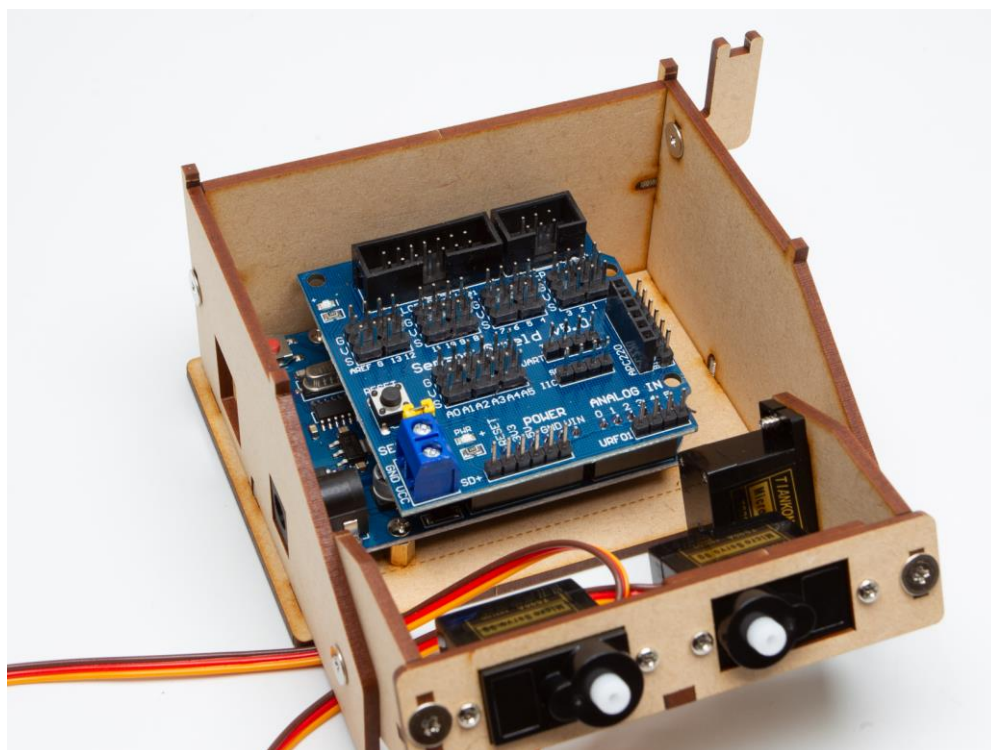


用最小的螺丝固定主板。





安装盾板



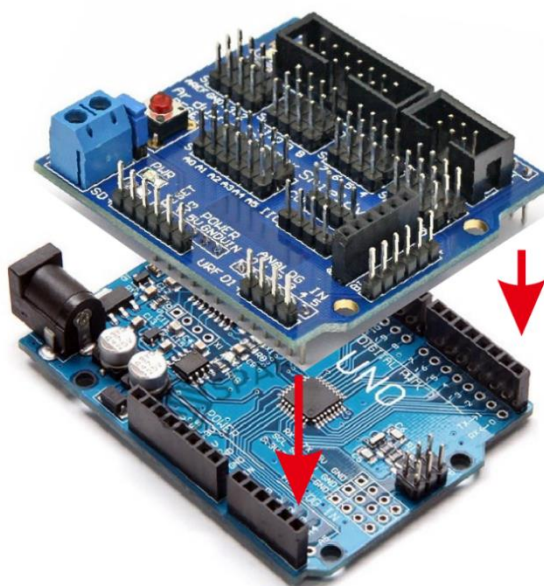
注意盾板的针脚和 Arduino 主板都一一对应，不要有遗漏或错位。

---

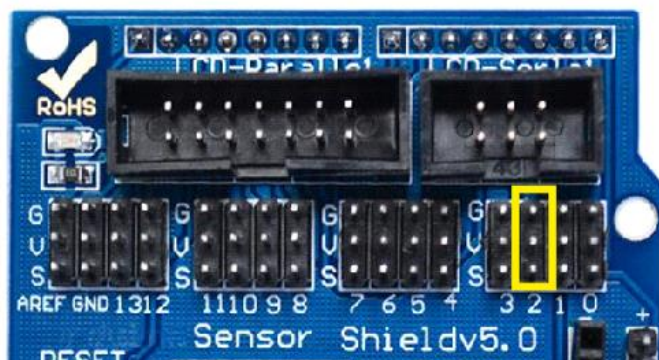


### 3. 接线

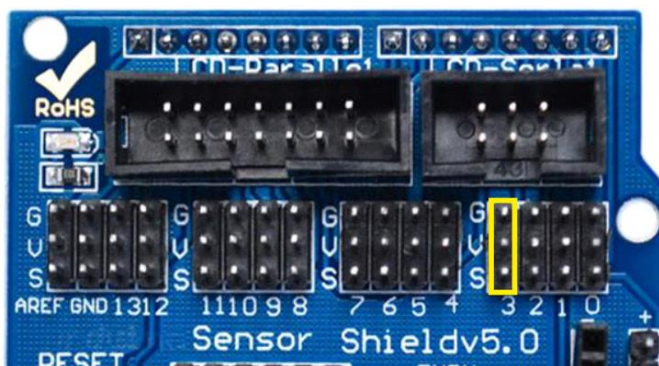
盾板 (Shield、扩展板) 插于 Arduino UNO 之上, 盾板针脚对准 UNO 的排线孔即可。盾板的作用类似接线板, 可以方便的接插舵机等设备。盾板针脚的号码和 UNO 的端口一一对应。每个端口都有一个 GND (简写 G)、VCC (V) 和 S 信号的接头。无需面包板接线非常方便。



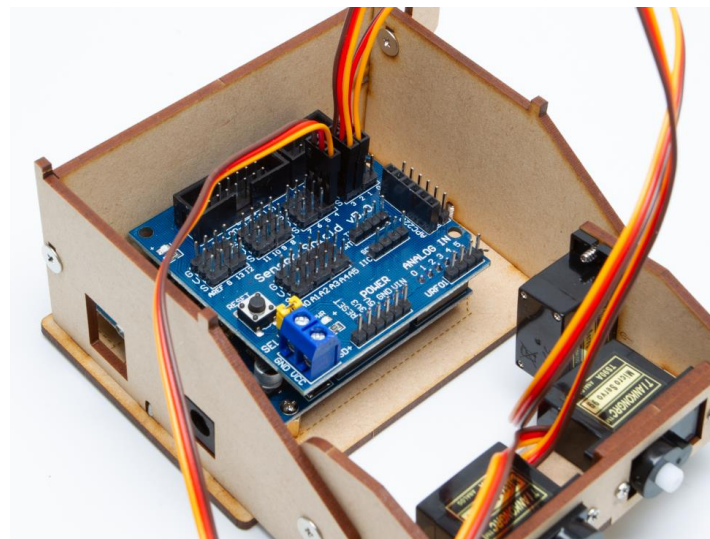
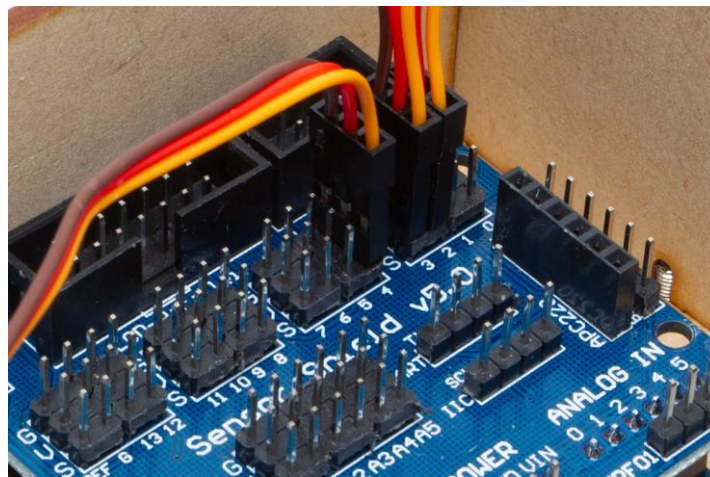
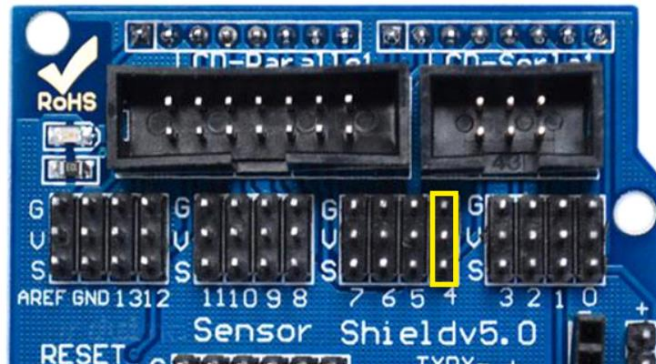
- 抬笔架的舵机 2# (舵机线是固定颜色: 棕色在 GND, 红色 VCC, 黄色是信号)



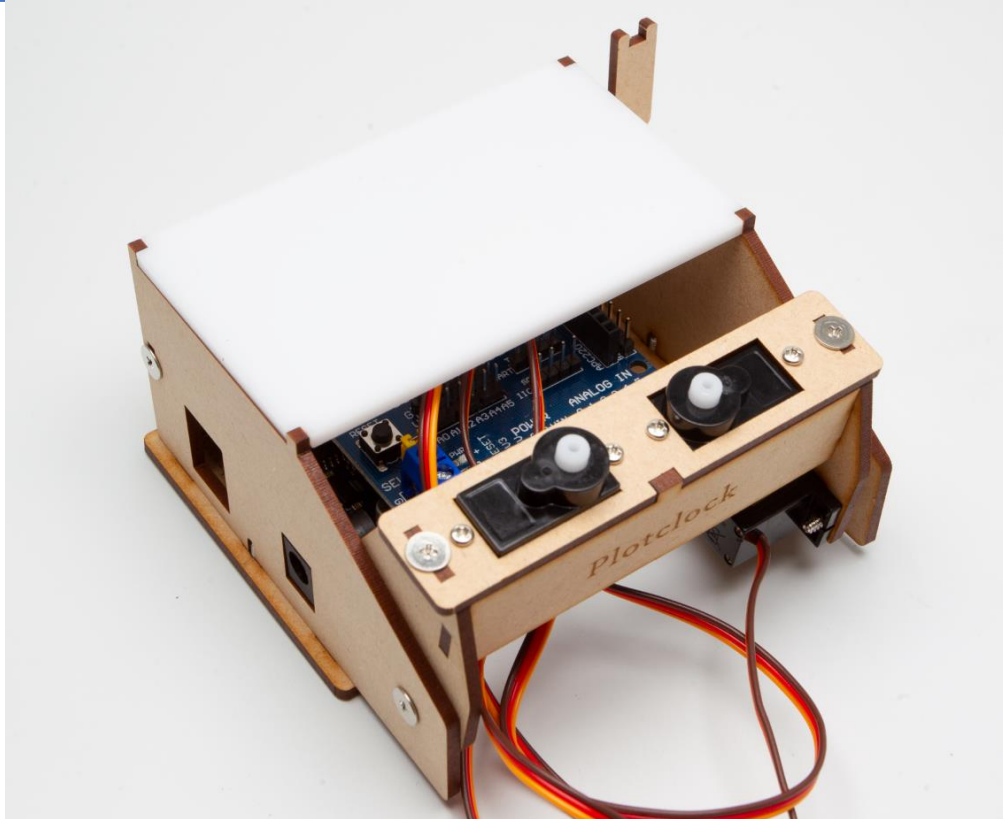
- 左测舵机接 3#



- 右测舵机接 4#



接线后，可以将白色的写字板盖好。



安装好写字板后，先不要安装笔擦支架和摆臂

## 调试程序

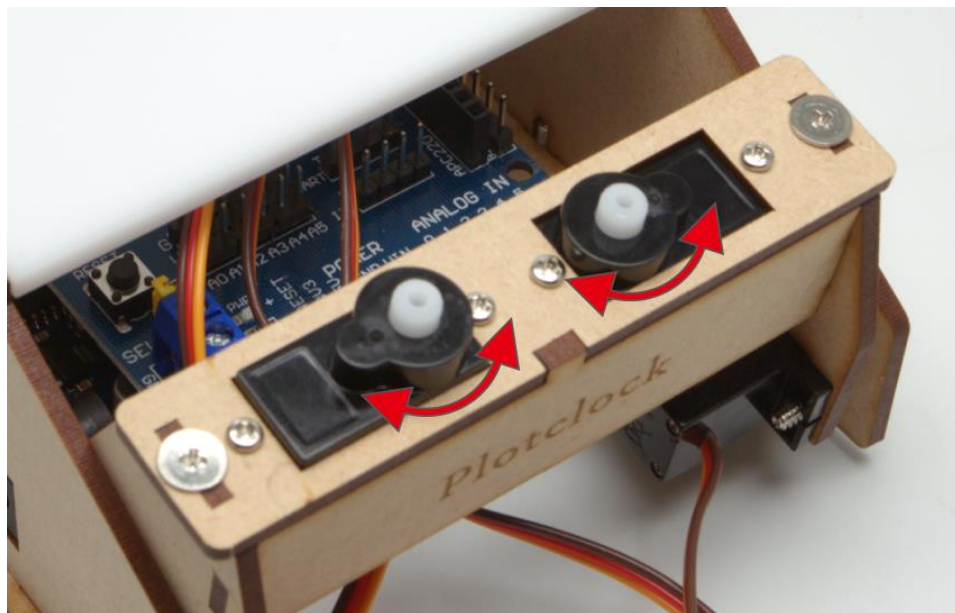
接下来刷程序。如果你还不会刷程序，甚至不会用 Arduino，那么请你先阅读 1 《**Arduino IDE 下载安装教程**》,2 《**Arduino 程序运行教程**》

先不要安装摆臂和笔擦支架，先刷主程序，注意最开始的行一下主程序，第 11 行 `#define CALIBRATION` 是否被正确定义，如果没有请复制到程序的最前面。

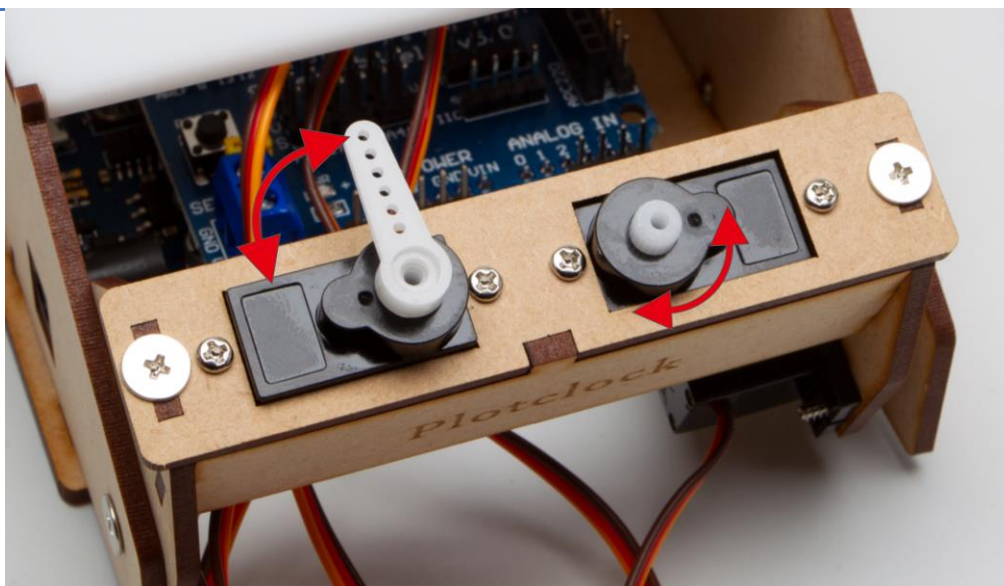




编译并上传程序到 Arduino。如果代码被正确上传，左右两个摆臂舵机将会往复旋转 90°。一直会重复此动作。



如果看不太清楚舵机的运动情况，可以插上一个舵机摆臂用来观察



摆臂插的位置不同，旋转的指向可能和照片中的不同，但是 2 个舵机的动作都是同时顺时针 90°，再逆时针 90°往复旋转。

我们要在舵机停顿的时候将 Arduino 电源断开，并安装上写字的摆臂。如果觉得停顿时间不够，可以修改代码中第 104 和 107 行 delay 的参数，1000=1 秒。

```
plotclock | Arduino 1.8.13
文件 编辑 项目 工具 帮助

plotclock

void loop()
{
  int i = 0;
  #ifdef CALIBRATION //条件编译，当定义了CALIBRATION 则编译下面99~117行 调节摆臂的代码，否则编译更下面正常运行的

    //舵机摆在90°的区域内往复运动，调节舵机参数让摆臂平行于x轴和y轴

    drawTo(6.1, 30.7); //box版
    delay(1000);

    drawTo(83.3, 26.7); //让笔尖运行到此位置，正常情况两摆臂将垂直或平行于xy轴，根据运行情况，调节舵机参数让摆臂接近
    delay(1000); //*/

    /* //如需调节抬臂高度，可以去掉注释，编译运行下面代码。
    //每个角度停留1秒钟，观察角度位置，调整LIFT0~2 三个参数
    lift(2); //高抬笔架
    delay(1000);
    lift(1); //写字时抬臂动作
    delay(1000);
    lift(0); //落笔写字 让摆臂龙门架刚好垂直，此位置最佳
    delay(1000);
    */
  }
```

修改 delay 的参数 2000 就是暂停 2 秒。类推

记录一下停机前舵机是顺时针旋转的还是逆时针的，两个安装方式不同。以下步骤 2 选 1 即可

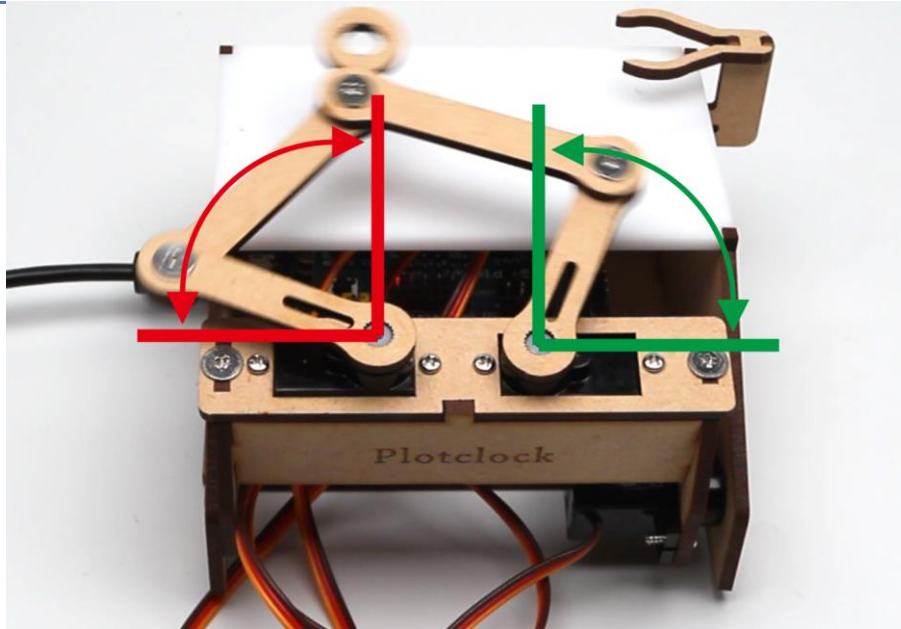




逆时针停机的位置，将左侧摆臂尽量平行于支架，右摆臂尽量垂直于支架。（因为摆臂要和舵机主轴的花键对应，所以一定会完全垂直或水平，接近即可后期用舵机参数修正）



如果是顺时针后停机，则左臂垂直，右臂水平。



安装好摆臂后，通电运行程序，观察两摆臂大致在红绿区间运行即可。并用粗牙带帽螺丝在舵机主轴上锁紧摆臂。

```
plotclock | Arduino 1.8.13
文件 编辑 项目 工具 帮助

plotclock

//+++++ 以下参数需要配合舵机摆臂运行位置进行修正，否则写字可能是歪的，甚至无法正常写字 +++++

//1.先调节0、180度的位置。调节到位后，再调节90度位置
//左右悬臂舵机的 0或180度位置，，数字增加，左侧舵机逆时针调整，右侧舵机顺时针调整
//【此数值可能需要调节】
#define SERVOLEFTNULL 2000 //数值减小，顺时针旋转，加大则逆时针旋转 默认2000
//【此数值可能需要调节】
#define SERVORIGHTNULL 1000 //数值减小，顺时针旋转，加大则逆时针旋转 默认1000

//2.调节到位0、180，再调节下面参数
//左右悬臂舵机的90度位置，，数字增加，左侧舵机顺时针调整，右侧舵机逆时针调整
//【此数值可能需要调节】
#define SERVOFAKTORLEFT 650 //数值加大，顺时针旋转，减小则逆时针旋转 默认650
//【此数值可能需要调节】
#define SERVOFAKTORRIGHT 650 //数值减小，顺时针旋转，加大则逆时针旋转 默认650

//升举舵机的3个角度
//【此数值可能需要调节】
#define LIPT0 1350 //落笔写字
#define LIPT1 1250 //写字时抬臂动作 一般H1:650 小100左右
```

修改程序中最开始的 4 个参数，让摆臂的各个动作趋近于垂直或水平。

先调节前 2 个，左右摆臂达到水平位置后，再调节后 2 个垂直状态的参数。参数增减 10 左右，大约 20 可以有 5° 左右的变化。一般修改参数不超过±200。每次修改参数，需要重新编译并上传，才能看到变化。



```
plotclock | Arduino 1.8.13
文件 编辑 项目 工具 帮助

plotclock

void loop()
{

int i = 0;
#ifdef CALIBRATION //条件编译, 当定义了CALIBRATION 则编译下面99~117行 调节摆臂的代码, 否则编译更下面正常运行的

//舵机摆在90°的区域内往复运动, 调节舵机参数让摆臂平行于x轴和y轴

drawTo(6.1, 30.7); //box版
delay(1000);

drawTo(83.3, 26.7); //让笔尖运行到此位置, 正常情况两摆臂将垂直或平行于xy轴, 根据运行情况, 调节舵机参数让摆臂接近
delay(1000); //*/

/*//每个角度停留1秒钟, 观察角度位置, 调整LIFT0~2 三个参数
lift(2); //高抬笔架
delay(1000);
lift(1); //写字时抬臂动作
delay(1000);
lift(0); //落笔写字 让摆臂龙门架刚好垂直, 此位置最佳
delay(1000);
//*/
#else
```

当摆臂角度达到可以修改代码，调节抬笔舵机的角度。



```
plotclock | Arduino 1.8.13
文件 编辑 项目 工具 帮助

plotclock

//2.调节到0、180, 再调节下面参数
//左右悬臂舵机的90度位置, 数字增加, 左侧舵机顺时针调整, 右侧舵机逆时针调整
//【此数值可能需要调节】
#define SERVOFAKTORLEFT 650 //数值加大, 顺时针旋转, 减小则逆时针旋转 默认650
//【此数值可能需要调节】
#define SERVOFAKTORRIGHT 650 //数值减小, 顺时针旋转, 加大则逆时针旋转 默认650

//升举舵机的3个角度
//【此数值可能需要调节】
#define LIFT0 1350 //落笔写字
#define LIFT1 1250 //写字时抬臂动作 一般比lift0 小100左右
#define LIFT2 1100 //高抬笔架
//【此数值可能需要调节】
```

三个参数要求写字状态，摆臂和写字板刚好平行，写字抬笔状态一般比写字状态小 100，高抬笔用来插笔擦，大约比抬笔小 150 左右。此 3 参数需要根据书写情况，再次微调。先保证摆臂和写字板平行即可。

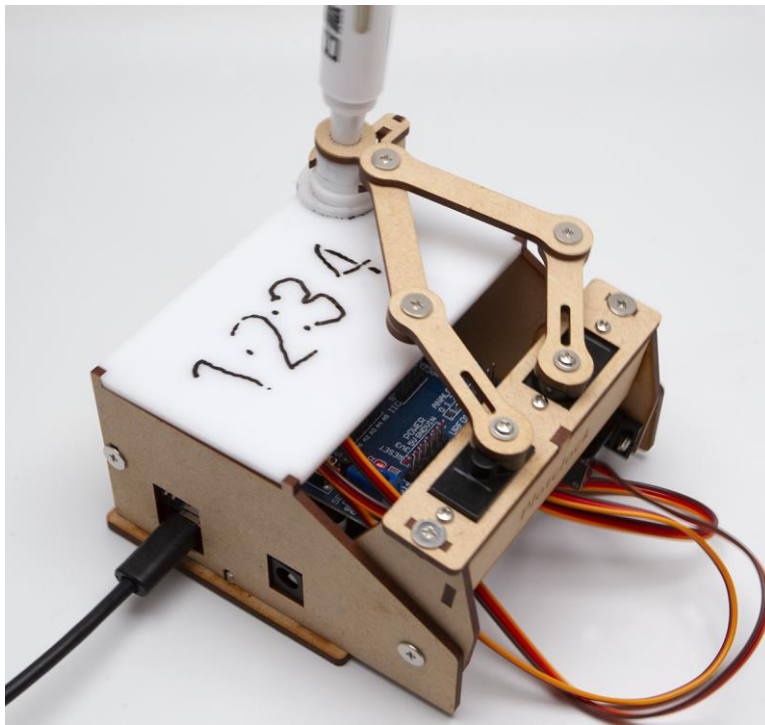
```
plotclock | Arduino 1.8.13
文件 编辑 项目 工具 帮助

plotclock $
//淘宝『有名称的店铺』https://somebodys.taobao.com/
//更新日期 2021/03/05
//Plotclock 小贱钟
//本程序对应商品 https://item.taobao.com/item.htm?spm=a1z10.3-c.w4002-1266050966.63.5c807027U2V1rh&id=
//web版程序连接: https://create.arduino.cc/editor/wjd76/abcc4e7f-508f-45e2-8284-db8d60c20181/preview
//Github版链接: https://github.com/shihaipeng03/Plotclock

// 调试完后可以删除以下两行代码 或者注释掉 调试模式，改为正常运行模式
//#define CALIBRATION // 调试模式（调试摆臂的位置）删除或注释掉此行进入 正常运行模式（正常书写时间）

//+++++ 以下参数需要配合舵机摆臂运行位置进行修正，否则写字可能是歪的，甚至无法正常写字 +++++
```

删除或注释掉 `#define CALIBRATION`，重新编译程序，进入正常书写程序



先不要安装笔和笔擦支架，观察摆臂的运行状态，如果无太大偏差再安装笔和笔擦等。首次运行要注意观察随时手动断电，防止笔和支架互相“打架”。

正式程序可能会遇到如下几个问题，也都可以通过修改参数来修正。

无法将笔插入笔擦的位置。因为高度影响，可以修改抬笔的参数 `#define LIFT2` 越小抬的越高。位置原因，修改 `int rubberx=82,rubbery=48;` 两个参数，x, y 单位是 mm。一般增减 1,2mm 即可。

书写时死机然后重启，可能是笔下压幅度太大，舵机无法旋转 to 该位置，堵转造成电流过大，保护性重启。（少量短时间死机并不会造成 Arduino 主板和舵机的损坏，但不要长时间频繁的过载

---

保护，尽快修改参数让舵机可以正常工作）修改写字位置的舵机参数 `#define LIFT0` 数字越大写字的压力越大，太小了写字断断续续不美观，太大了舵机压力过大，Arduino 会保护重启无法正常工作，适度即可。

写字的过程抬笔或落笔都会抖的很严重，造成字的笔画歪歪扭扭。除了适当调节抬笔高度的参数 `#define LIFT1`，还可以修改 `#define LIFTSPEED` 参数，单位是微妙，1000 微妙=1 毫秒。越大越慢越稳。

起始时间的调节在 `setup` 中，`settime(12,34,0)`; 3 个参数分别为，小时，分钟，秒，开机后从这个时间开始计时。每次 Arduino 重启都会从这里开始，不能延续计时。如果有需求，可以自行增加时钟模块或其他计时设备对时。

## 集中出现的问题

1 问：如果同步网络上的时间

答：Arduino 在没有网络模块的情况下，无法连接网络同步时间。现在用的是模拟计时器，所以每次通电都是从 12:34 开始。可以在 `setup` 中修改这个初始值。

2 问：为什么我的字写的这么歪歪扭扭

答：9 克舵机的精度本来就一般，再加上本身字体就不是非常规整，所以即使调节的再准，写的也都一般。可以适当调节速度参数，让舵机运行的更平稳，会有所提高。

3 问：如何书写中文？

答：本程序只有数字的代码，并没有其他字符，可以自行添加。示例程序中有个画天气符号的例子，属于提高内容，可以自行研究。

---

## 了解更多

欢迎到我们的店铺 <https://somebodys.taobao.com>

---