



# 开源硬件入门

陈 新

电子信息工程学院

创新实验室

2018.9

# 上次回顾

课内讲授：

- Arduino UNO



- Arduino IDE



- SolidWorks



- **数字IO，模拟IO，强电和弱电**

课后完成：

- 提前安装arduino IDE和Solidworks
- 准备一个测量长度的工具
- 完善各组的小设计方案



# 本次要点

- 方案指导：对各组方案进行分析指导，确定可行的方案



- 完成2个实验：
- 1 ) Github+Arduino IDE
- 2 ) Arduino应用程序+3D建模工具

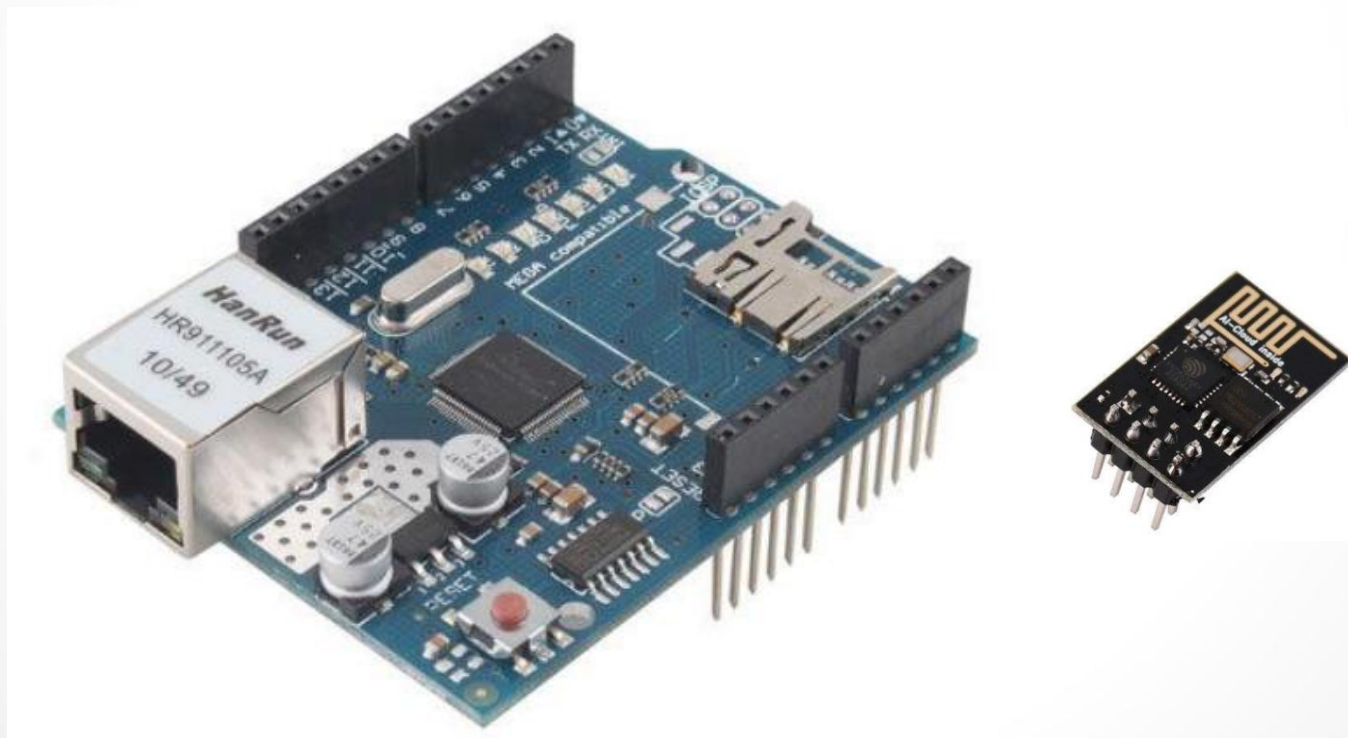
# 方案指导



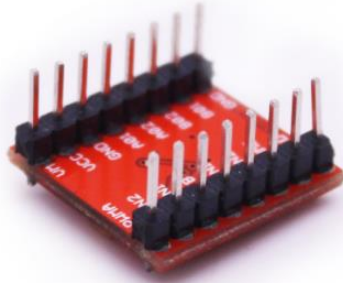
时间：20-30分钟

# 网络扩展模块

- W5100 ( 有线 )
- ESP8266 ( 无线 )



# 电机驱动TB6612



TB6612驱动模块是双驱动，也就是可以驱动两个电机

下面分别是控制两个电机的IO口

STBY口接单片机的IO口置0：电机全部停止，置1：使能驱动模块，通过AIN1 AIN2，BIN1，BIN2 来控制正反转

VM 接15V以内电源

VCC 接5V电源

GND 接电源负极

## 驱动1路

PWMA 接单片机的PWM口

真值表：

AIN1	0	0	1
AIN2	0	1	0
	停止	正转	反转

AO1 接1电机的正极

AO2 接1电机的负极

## 驱动2路

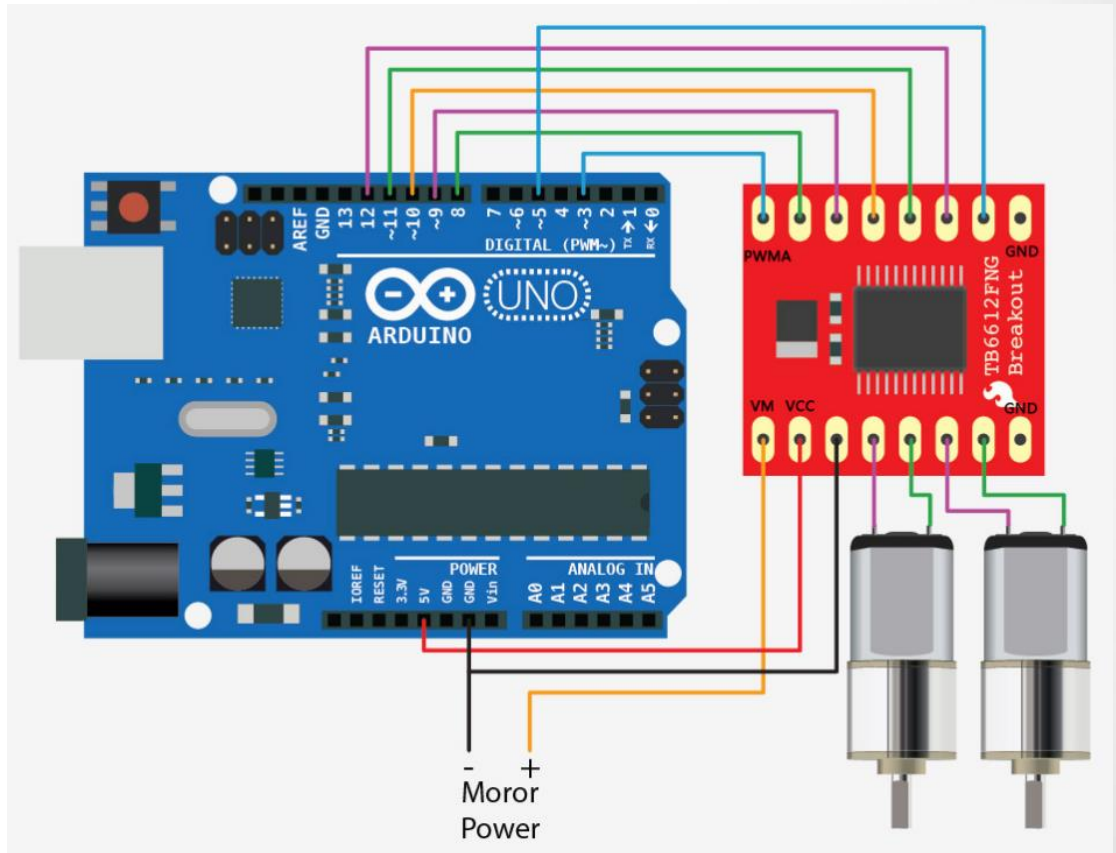
PWMB 接单片机的PWM口

真值表：

BIN1	0	0	1
BIN2	0	1	0
	停止	正转	反转

BO1 接2电机的正极

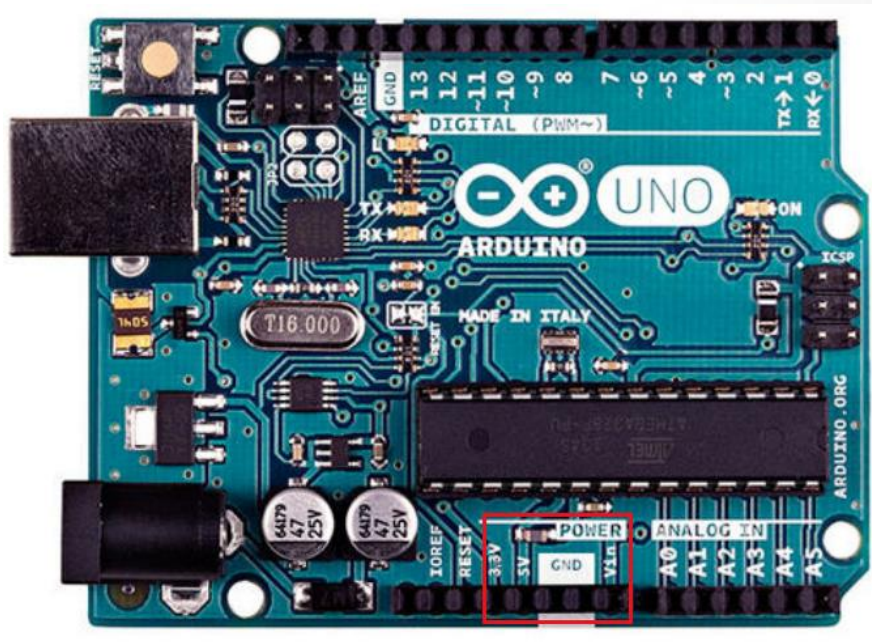
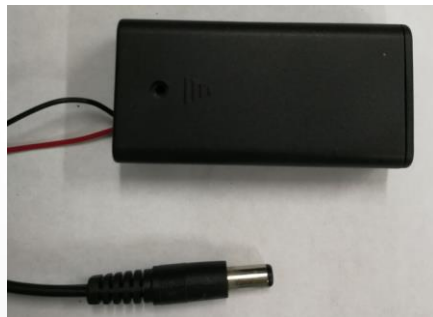
BO2 接2电机的负极





# 供电电源和设备电源

- 电机（或其他强电设备）的电源必须从UNO板的Vin和Gnd用杜邦线引出（**前提是使用了8.4伏电池盒**）
- 其他弱电设备可以使用UNO的5V和Gnd，或3.3V和Gnd



# RFID模块MFRC522





# 电磁铁



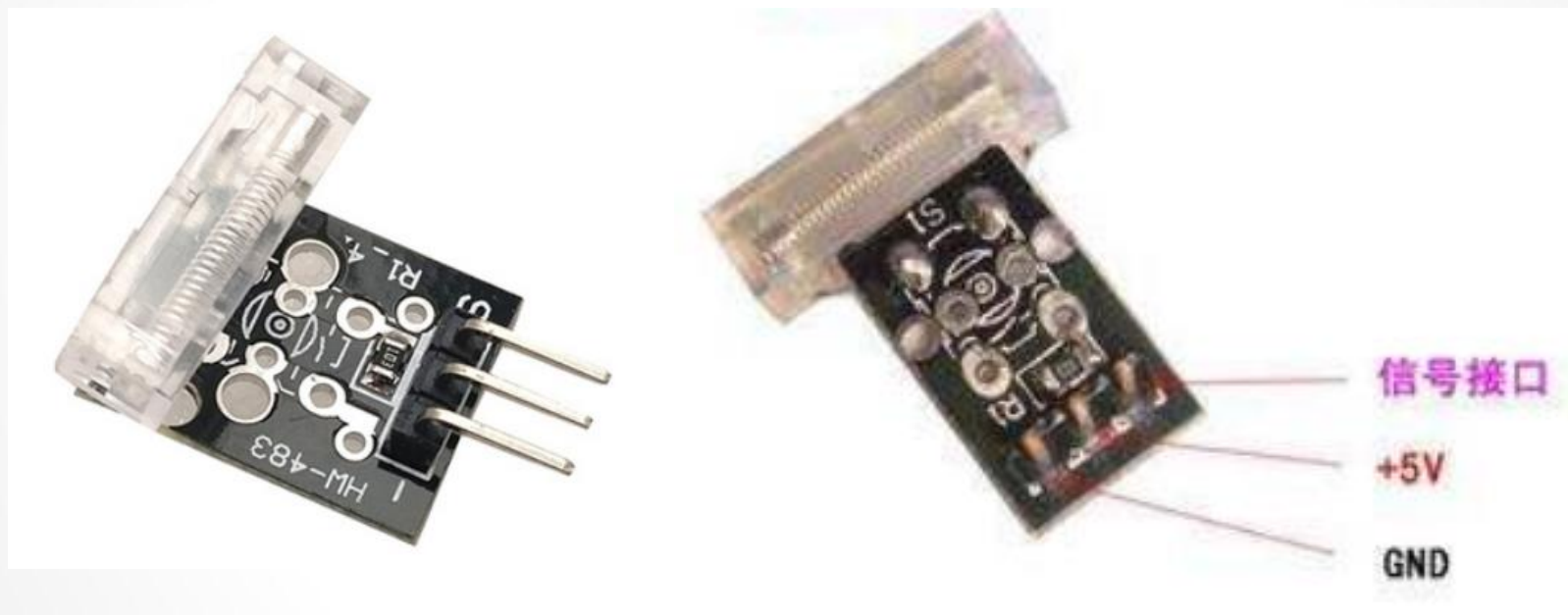
# 微动开关



# 时钟模块



# 敲击传感器



# 方案指导



时间：20-30分钟



# 实验课用到的教具-电子套件





# 开源项目分享



open source  
hardware



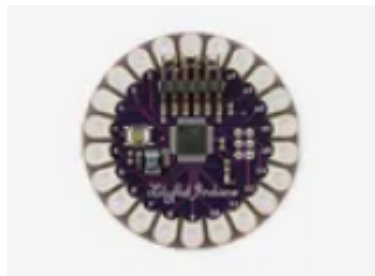
# LED舞会礼服-整体效果



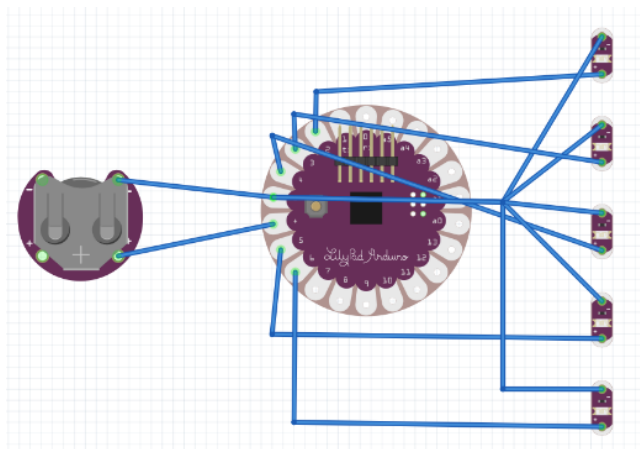
# LED舞会礼服-硬件设计

## 硬件清单

- LilyPad
- LED
- 锂电池

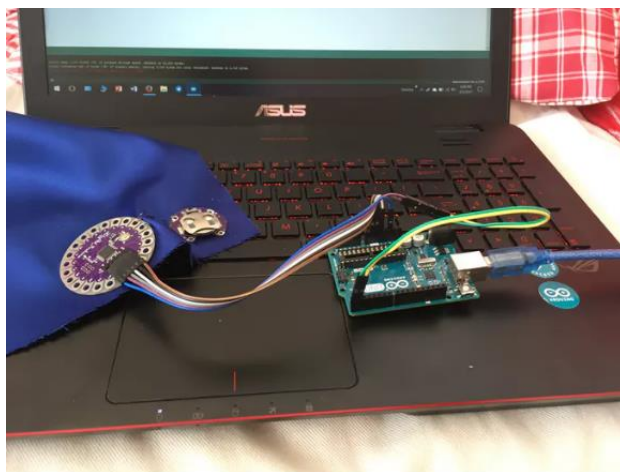


## 硬件连接示意图



# LED舞会礼服-其他资料

- 其他资料（照片，视频，硬件设计，软件设计等）



- 项目链接：  
<https://create.arduino.cc/projecthub/Maddy/starry-night-prom-2eb206>


# 实验1

- 任务单
  - 1 ) Github上传分组信息，并将github的链接发到课程平台
  - 2 ) 安装Arduino IDE和Arduino UNO的驱动
  - 3 ) 编写程序，显示PC和Arduino的串口通信。PC发送 “I am XXX” ， Arduino回复 “OK” ， 其中XXX用各组的队名代替
- 提交方式：将串口通信成功的截图，以及Arduino的源代码上传到课程平台

# Arduino IDE

- Integrated Development Environment
- <https://www.arduino.cc/en/Main/Software>

Download the Arduino IDE




### ARDUINO 1.8.6

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

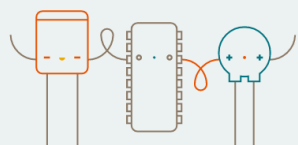
**Windows** Installer, for Windows XP and up  
**Windows** ZIP file for non admin install

**Windows app** Requires Win 8.1 or 10

Get 

## Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **26,038,610** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3

\$5

\$10

\$25

\$50

OTHER

JUST DOWNLOAD

CONTRIBUTE & DOWNLOAD

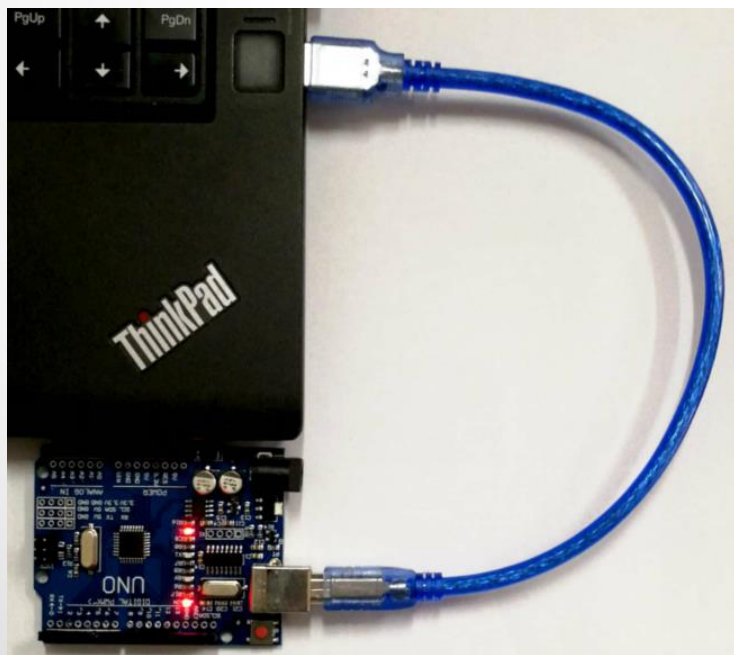
• 北京交通大学 电子信息工程学院 大学生创新活动中心

2018-09-21 •



# 连接和安装UNO

- 下载线一头接电脑，一头接控制板
- 直接运行CH340驱动文件夹里面的CH341SER
- 点击安装，安装完成后可以在IDE中查看



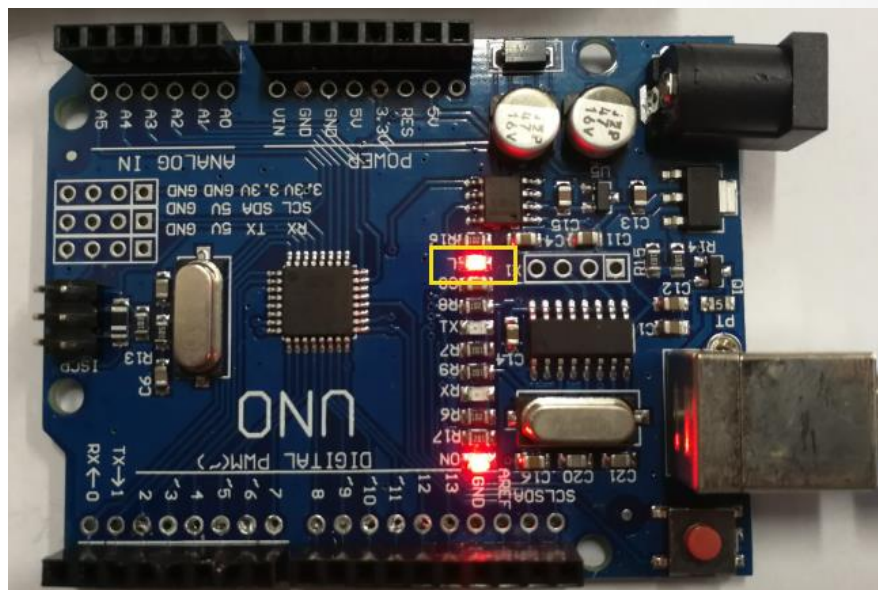
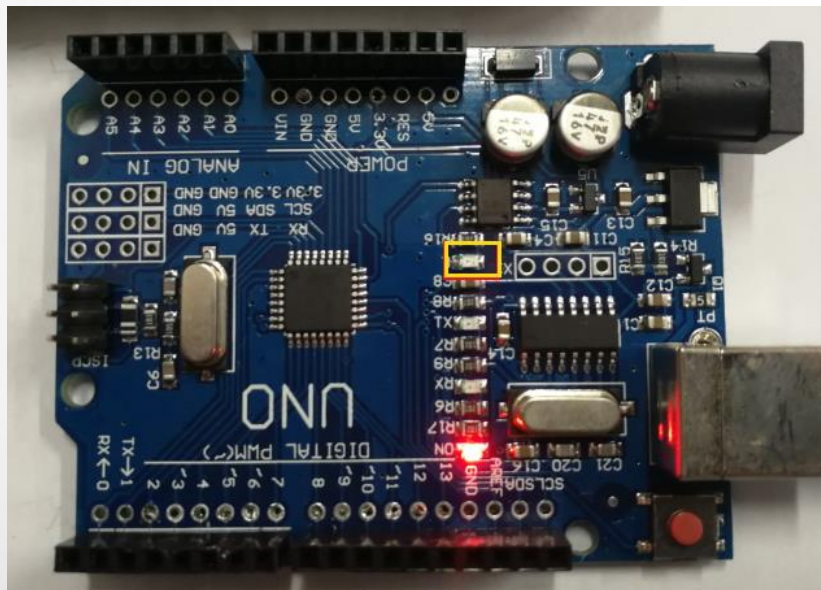
sketch\_sep11a | Arduino 1.6.7

文件 编辑 项目 工具 帮助



# 闪灯程序

- 依次点击文件——示例——01.Basic——Blink，打开闪灯例程
- 点击上传，观察板上灯的闪烁



# IDE实验：串口通信

- PC发送 “I am XXX” ， Arduino回复 “OK”

```
String comdata = "";  
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
}  
void loop() {  
  // put your main code here, to run repeatedly:  
  while (Serial.available() > 0) {  
    comdata += char(Serial.read());  
    delay(2);  
  }  
  if (comdata.length() > 0 && comdata == "I am alben") {  
    Serial.println(comdata);  
  }  
  comdata = "";  
}
```

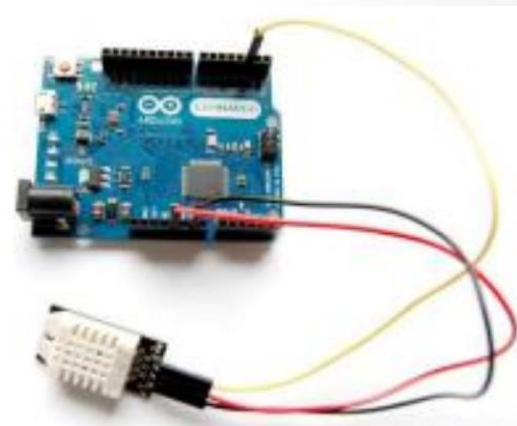
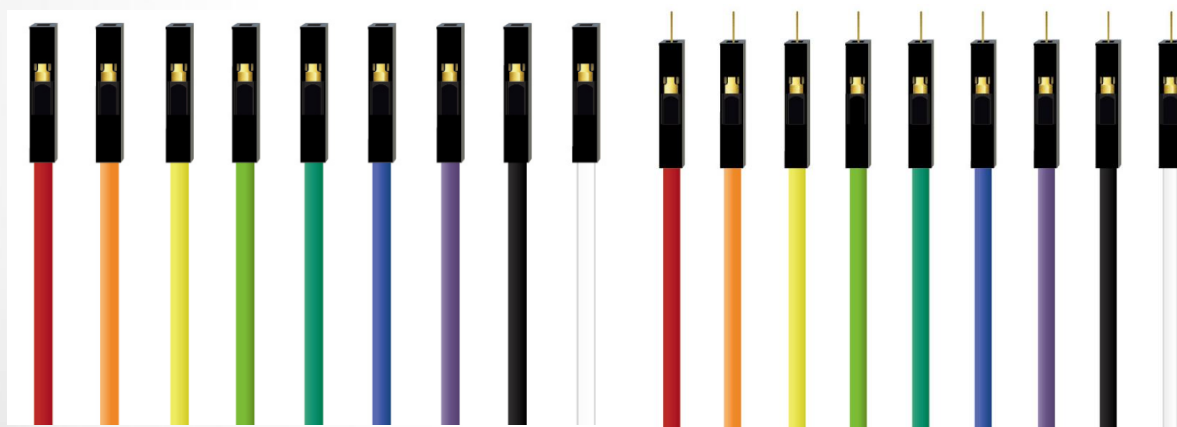


# 实验2

- 任务单
  - 1) 实现继电器控制直流电机转动
  - 2) 实现控制舵机（伺服电机）角度
  - 3) 用solidworks打开“电机连接件.STEP”进行多角度旋转
  - 4) 测量积木零件尺寸和电路模块尺寸，用solidworks设计一个自己小设计中会用到的连接件
- 提交方式：将任务1和任务2的硬件照片，任务3的三个不同角度的截图，任务4的零件截图，以及Arduino的源代码上传到课程平台。任务4的设计文件（step格式）和零件连接文字说明（TXT格式）传到github，

# 硬件连接-杜邦线

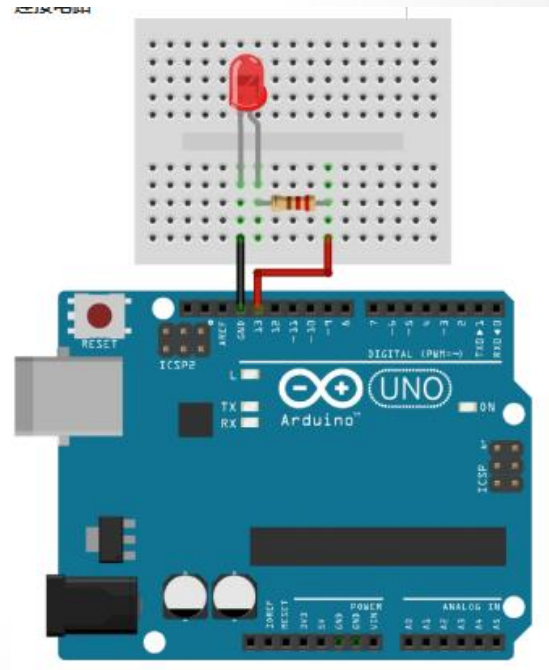
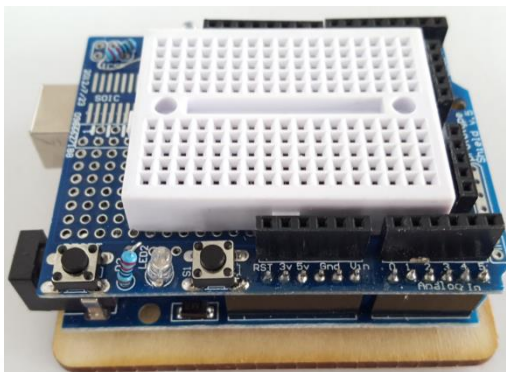
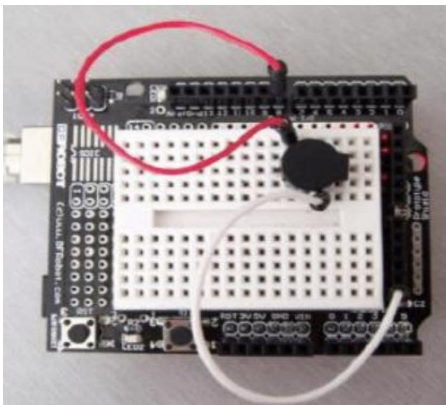
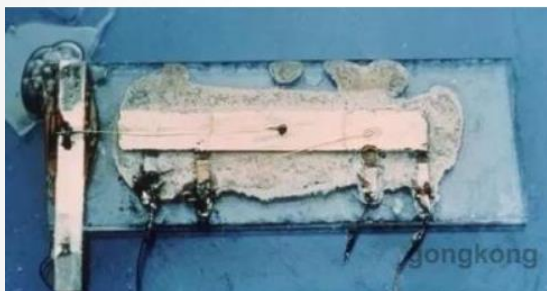
- 美国杜邦公司最先生产
- 电子行业杜邦线可用于实验板的引脚扩展  
无需焊接，可以快速进行电路试验。





# 硬件连接-面包板

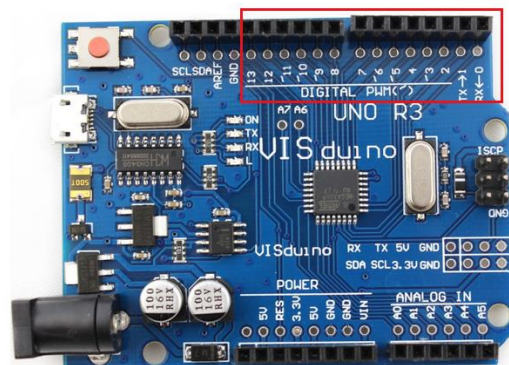
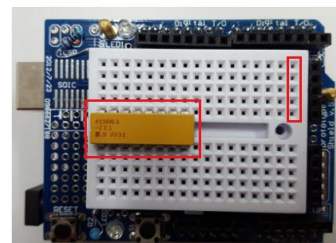
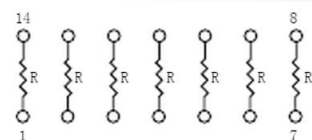
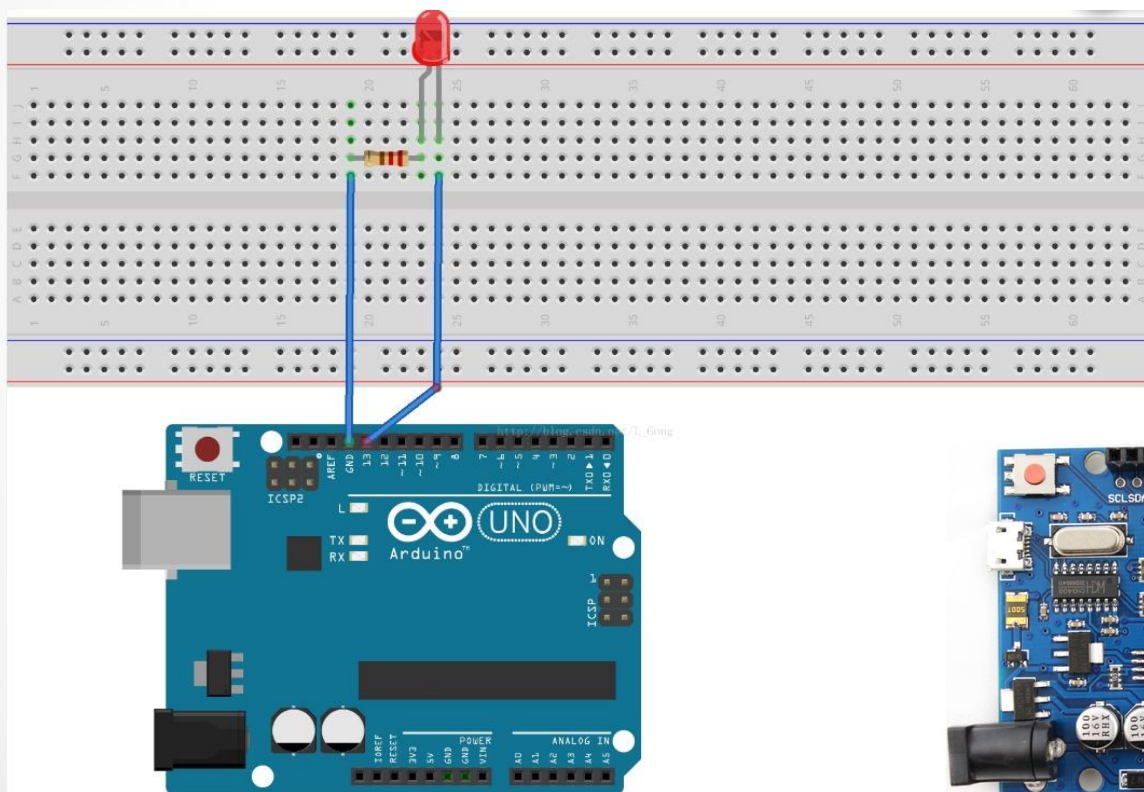
- 真空管电路的年代，当时电路元器件大都体积较大，人们通常通过螺丝和钉子将他们固定在一块切面包用的木板上进行连接。





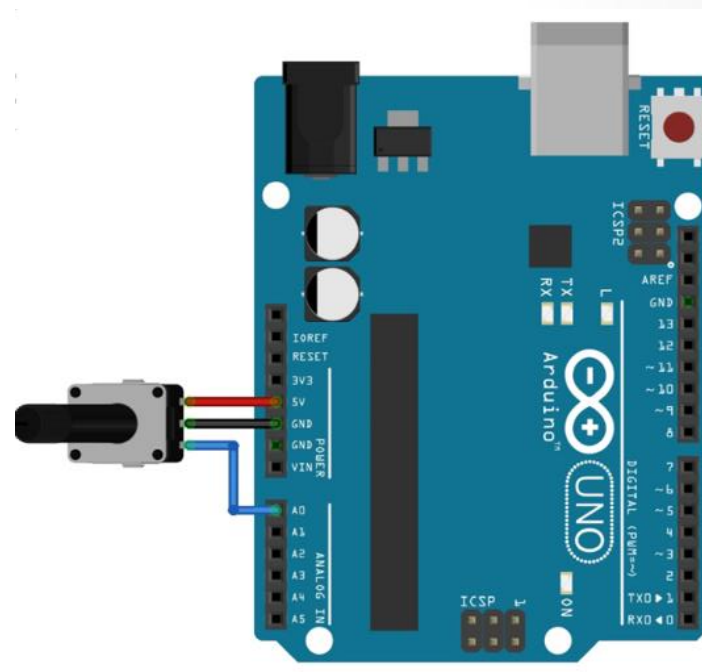
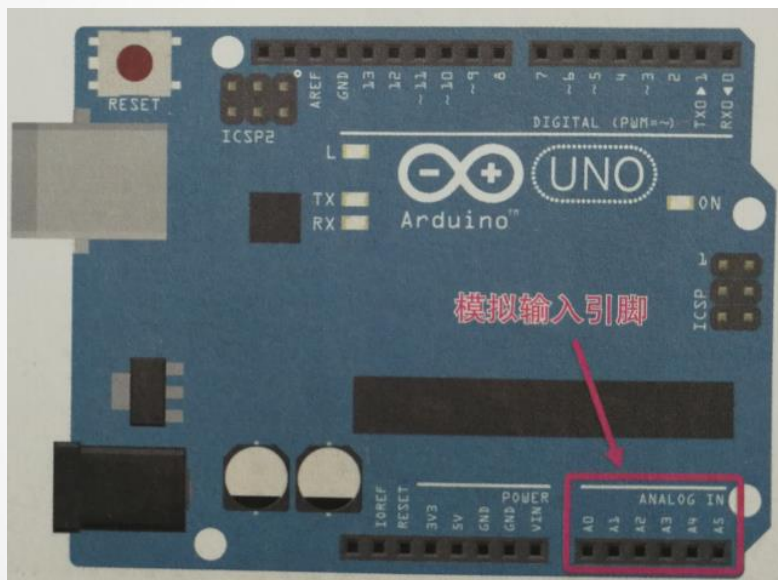
# 数字输入输出口

- 0和1两种状态，间断的变化



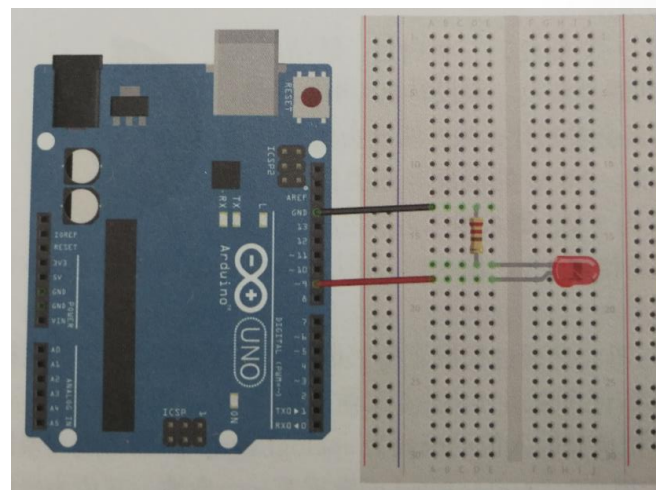
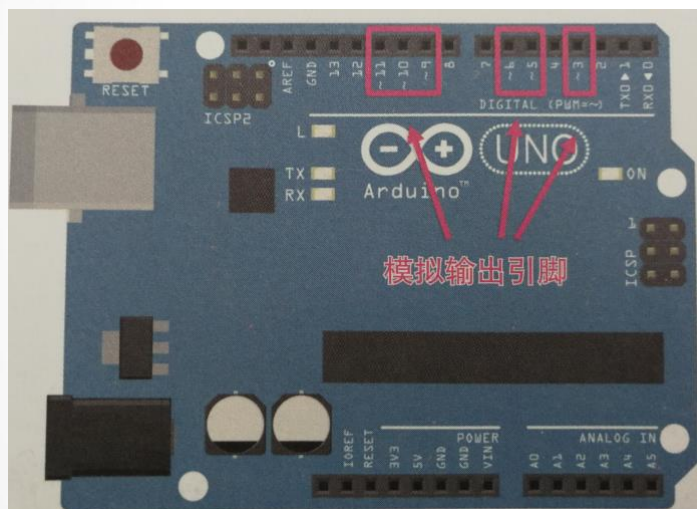
# 模拟输入口

- 读取连续不间断的信号，如：读取电压



# 模拟输出口

- 连续不间断的输出，如：呼吸灯，控制伺服电机（舵机）



# 继电器

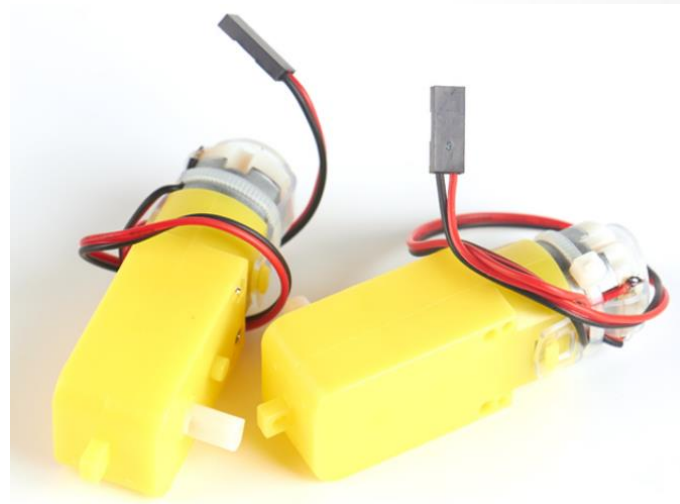




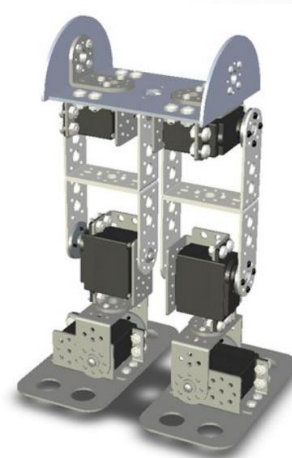
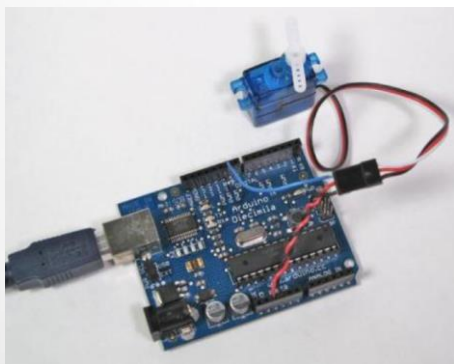
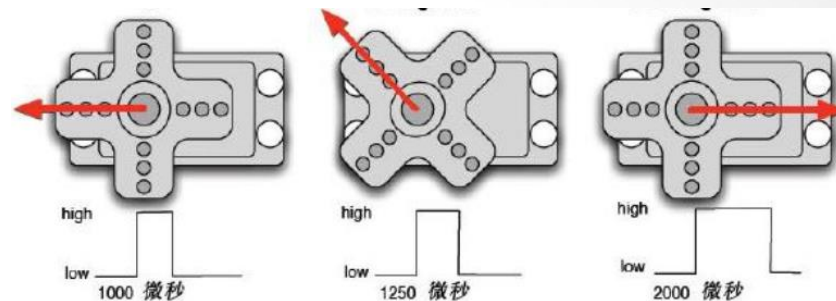
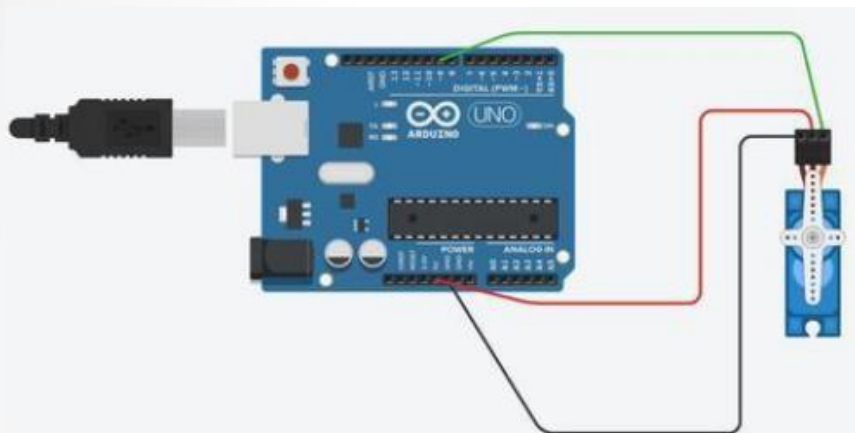


# 直流电机

- 可以提供2种型号：单轴和双轴



# 伺服电机（舵机）







# 库函数示例

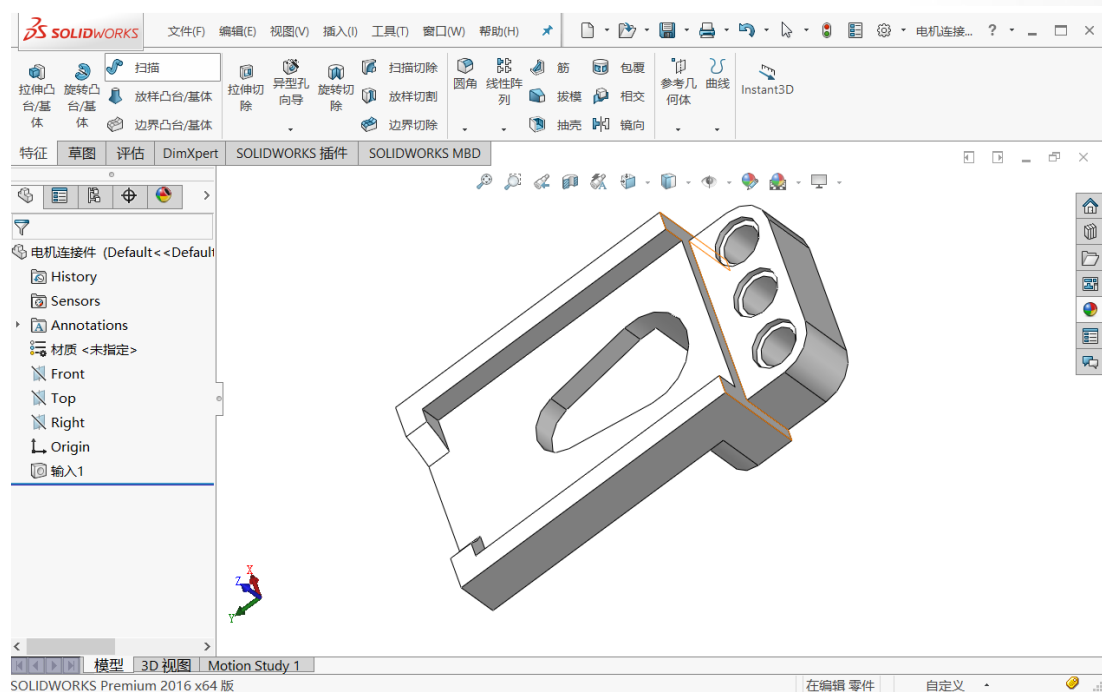
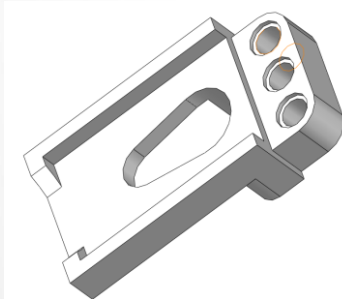
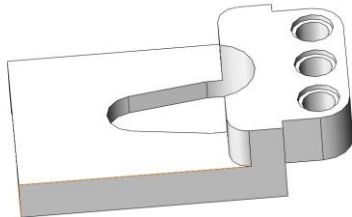
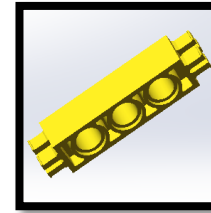
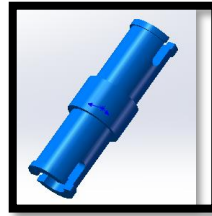
```
void setup() {  
    // put your setup code here, to run once:  
    Serial.begin(9600);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    Serial.println("Hello world");  
    delay(1000);  
}
```

```
void setup() {  
    // initialize digital pin 13 as an output.  
    pinMode(13, OUTPUT);  
}
```

```
// the loop function runs over and over again forever  
void loop() {  
    digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)  
    delay(1000);           // wait for a second  
    digitalWrite(13, LOW);  // turn the LED off by making the voltage LOW  
    delay(1000);           // wait for a second  
}
```

```
#include <Servo.h>  
  
Servo myservo; // create servo object to control a servo  
// twelve servo objects can be created on most boards  
  
int pos = 0;    // variable to store the servo position  
  
void setup() {  
    myservo.attach(9); // attaches the servo on pin 9 to the servo object  
}  
  
void loop() {  
    for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees  
        // in steps of 1 degree  
        myservo.write(pos);              // tell servo to go to position in variable 'pos'  
        delay(15);                       // waits 15ms for the servo to reach the position  
    }  
    for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees  
        myservo.write(pos);              // tell servo to go to position in variable 'pos'  
        delay(15);                       // waits 15ms for the servo to reach the position  
    }  
}
```

# Solidworks



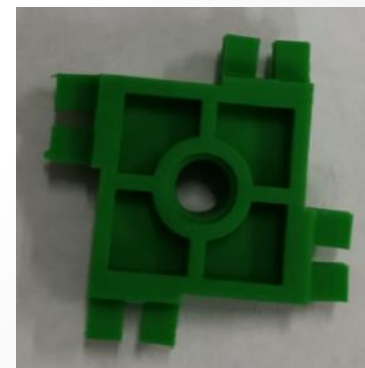
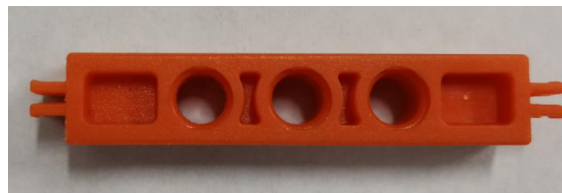


# 电子和积木结合

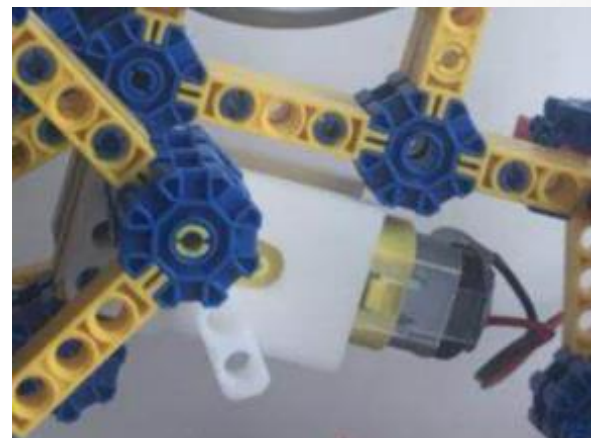
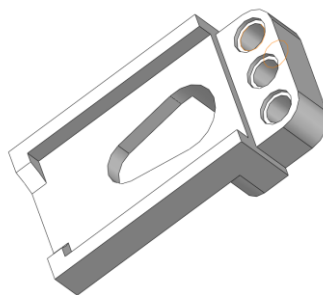
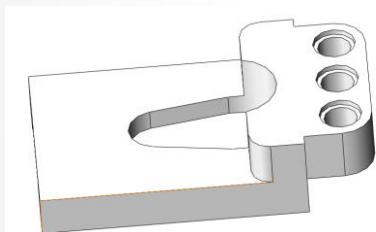
- 原有零件（4种）



- 新增零件（4种）

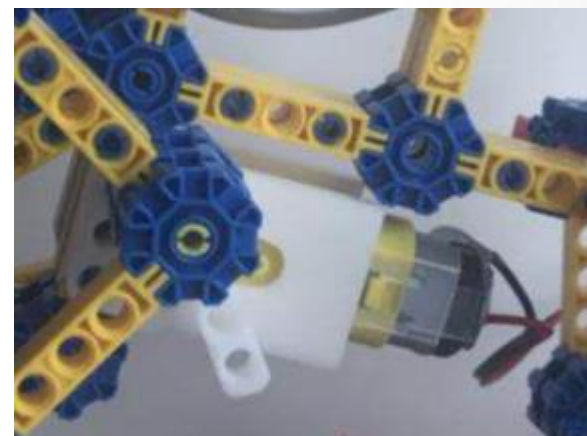
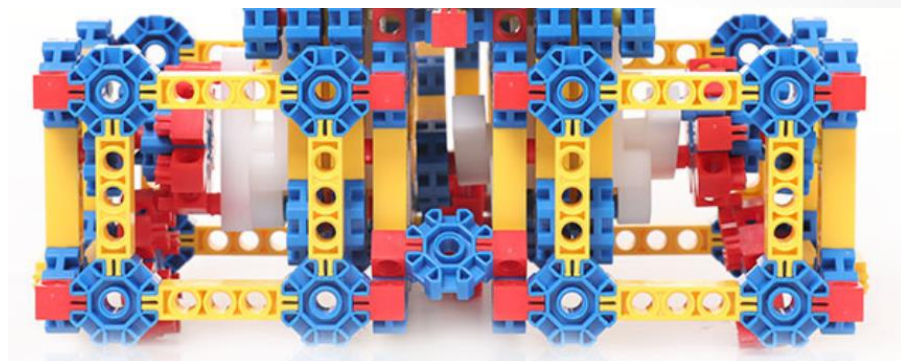


# 电子和积木结合



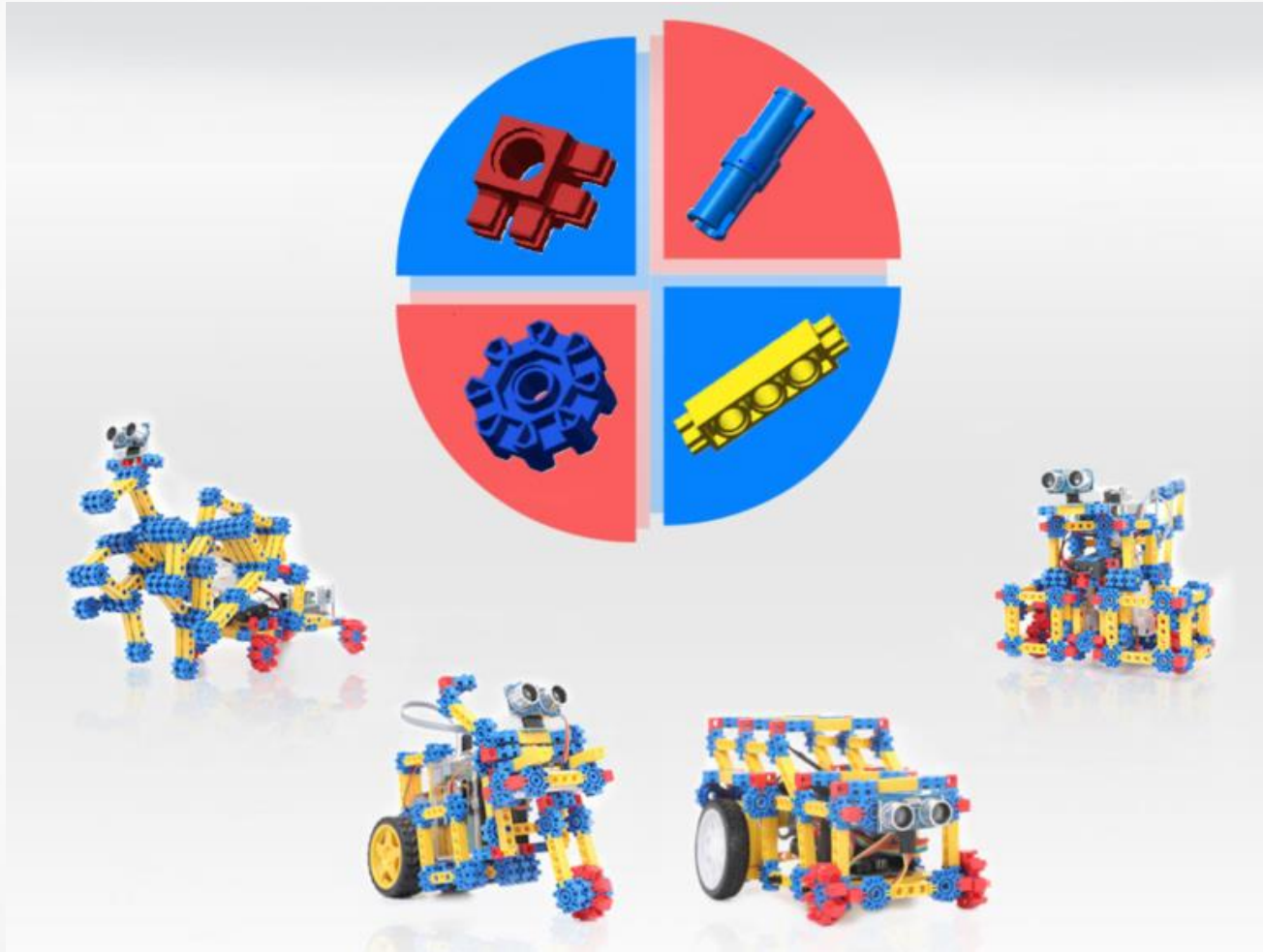


# 电子和积木结合





# 电子和积木结合





# 课后工作

- 10月9号前在课程平台和github提交前两次实验
- 提前完成各组方案的硬件连接
- 优化百变几何积木套件在各组小设计中的应用
- 实现电子和积木的完美结合
- 每组查找一个自己感兴趣的国外内的开源硬件案例，做个2-3页的PPT，下次课和大家分享

