

Arduino 单片机实验

李阳 11307110026

复旦大学物理系

June 11, 2014

成品简介

成品简介

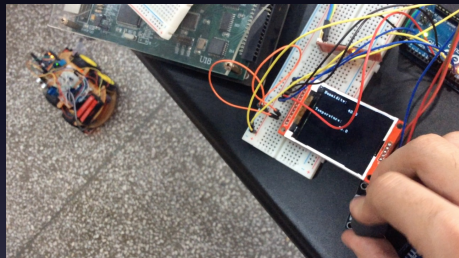


Figure : 成品实拍

成品简介

1 摇杆蓝牙遥控小车

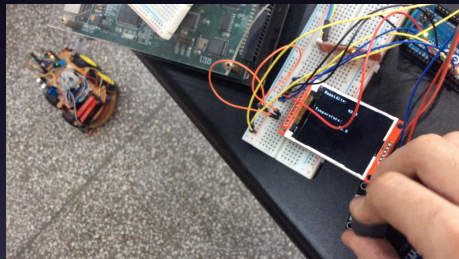


Figure : 成品实拍

成品简介

- 1 摇杆蓝牙遥控小车
- 2 正反 255 级调速

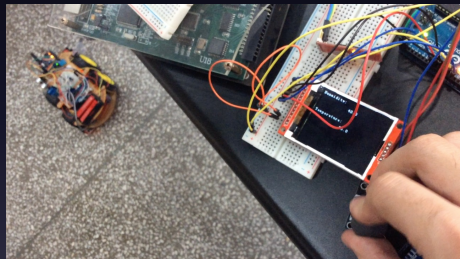


Figure : 成品实拍

成品简介

- 1 摇杆蓝牙遥控小车
- 2 正反 255 级调速
- 3 小车上可搭载各类传感器

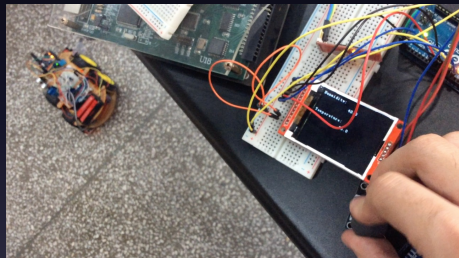


Figure : 成品实拍

成品简介

- 1 摇杆蓝牙遥控小车
- 2 正反 255 级调速
- 3 小车上可搭载各类传感器
- 4 将小车传感器数据实时传回并显示在控制端屏幕上

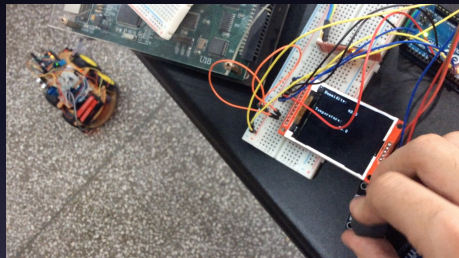


Figure : 成品实拍

Outline

Outline

1 成品简介

Outline

- 1 成品简介
- 2 Arduino 简介

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

Arduino 简介

Arduino 简介

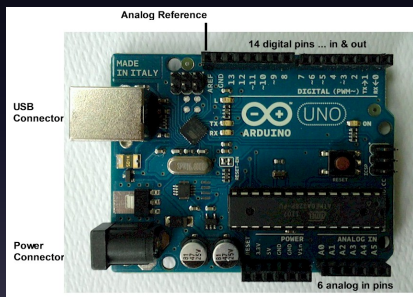


Figure : Arduino Uno

Arduino 简介

- 1 Arduino 一个开源软硬件平台, Arduino Uno 是 Arduino 旗下一系列单片机型号中的一种

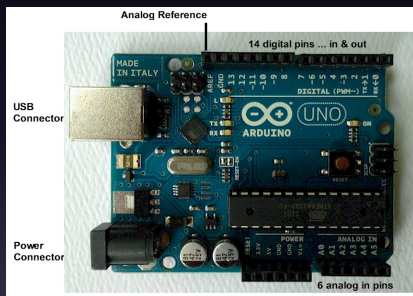


Figure : Arduino Uno

Arduino 简介

- 1 Arduino 一个开源软硬件平台, Arduino Uno 是 Arduino 旗下一系列单片机型号中的一种
- 2 通过管脚与各类电子元件相连, 起到中控作用
 - 1 +5V/+3.3V/GND
 - 2 Digital/Analog
 - 3 PWM
 - 4 TX/RX

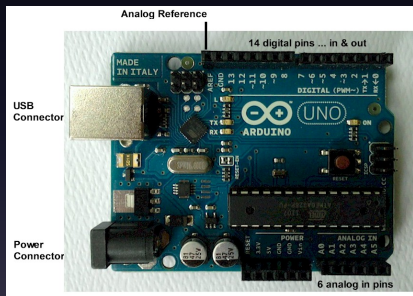
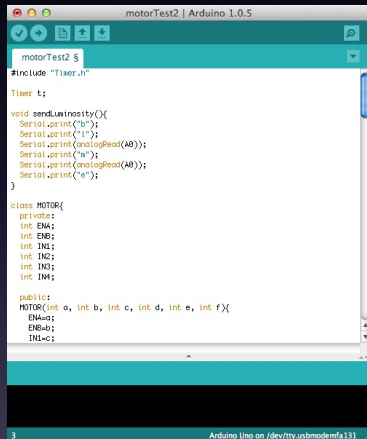


Figure : Arduino Uno

Arduino 简介

- 1 Arduino 一个开源软硬件平台，Arduino Uno 是 Arduino 旗下一系列单片机型号中的一种
- 2 通过管脚与各类电子元件相连，起到中控作用
 - 1 +5V/+3.3V/GND
 - 2 Digital/Analog
 - 3 PWM
 - 4 TX/RX
- 3 控制程序自行编写，通过 Arduino IDE 烧录到单片机中，语言：C++



```
motorTest2 | Arduino 1.0.5

#include "Timer.h"

Timer t;

void sendLuminosity(){
  Serial.println("b");
  Serial.println("1");
  Serial.println(analogRead(A0));
  Serial.println("n");
  Serial.println(analogRead(A0));
  Serial.println("e");
}

class MOTOR{
private:
  int ENA;
  int ENB;
  int IN1;
  int IN2;
  int IN3;
  int IN4;
public:
  MOTOR(int a, int b, int c, int d, int e, int f){
    ENA=a;
    ENB=b;
    IN1=c;
  }
}
```

Figure : Arduino IDE

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

原理 & 结构

原理 & 结构

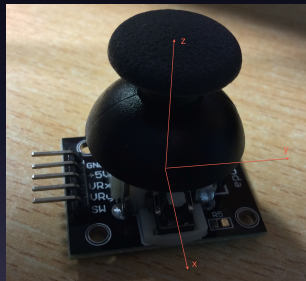


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

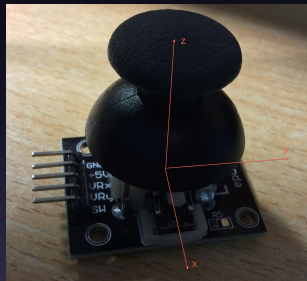


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- 1 GND：接地

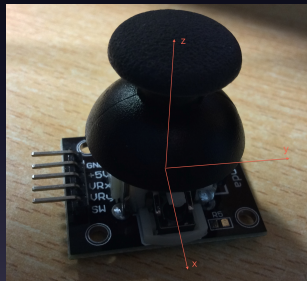


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- 1 GND：接地
- 2 +5V：正极

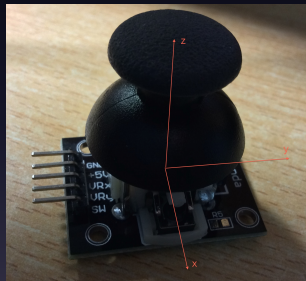


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- 1 GND：接地
- 2 +5V：正极
- 3 VRx： x 方向变阻器电压值（模拟信号，0-1023）

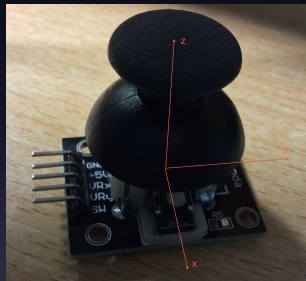


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- 1 GND: 接地
- 2 +5V: 正极
- 3 VRx: x 方向变阻器电压值（模拟信号，0-1023）
- 4 VRy: y 方向变阻器电压值（模拟信号，0-1023）

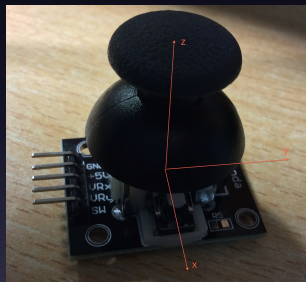


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- 1 GND: 接地
- 2 +5V: 正极
- 3 VRx: x 方向变阻器电压值（模拟信号，0-1023）
- 4 VRy: y 方向变阻器电压值（模拟信号，0-1023）
- 5 SW: z 方向按钮（数字信号，0/1）

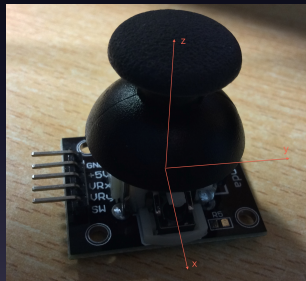


Figure : Joystick

原理 & 结构

摇杆实际上可以看做 x 、 y 两个方向分别有一个滑动变阻器加上 z 方向的一个按钮，共 5 个针脚：

- ① GND: 接地
- ② +5V: 正极
- ③ VR_x: x 方向变阻器电压值（模拟信号，0-1023）
- ④ VR_y: y 方向变阻器电压值（模拟信号，0-1023）
- ⑤ SW: z 方向按钮（数字信号，0/1）

只用到 x 、 y 方向操控， z 方向按钮没用到，SW 不接线。

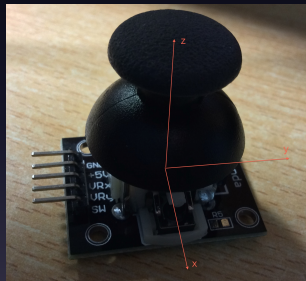


Figure : Joystick

实现 & 代码

将 GND、+5V、VRx、VRy 接在 Arduino Uno 的对应管脚上，VRx，VRy 是模拟信号输出，接 A0-A5 任二。

```
1 void setup(){
2   pinMode(A0, INPUT); //A0 连 VRx, 输入模式
3   pinMode(A1, INPUT); //A1 连 VRy, 输入模式
4 }
5
6 void loop(){
7   int x=analogRead(A0); //读取 VRx 模拟信号值
8   int y=analogRead(A1); //读取 VRy 模拟信号值
9   delay(200); //延时 200ms 执行下一次循环
10 }
```

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

L298N 直流电机驱动板

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno
上电流不够，无法驱动

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno
上电流不够，无法驱动
- 2 可控制电机正转反转，转速调节

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno 上电流不够，无法驱动
- 2 可控制电机正转反转，转速调节

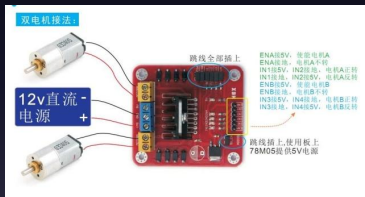


Figure : L298N 直流电机驱动板

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno 上电流不够，无法驱动
- 2 可控制电机正转反转，转速调节
- 3 分为控制模块与驱动模块，

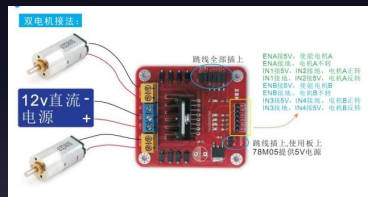


Figure : L298N 直流电机驱动板

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno 上电流不够，无法驱动
- 2 可控制电机正转反转，转速调节
- 3 分为控制模块与驱动模块，
 - 1 控制模块与单片机相连（包括供电），调控驱动模块的电流大小，方向

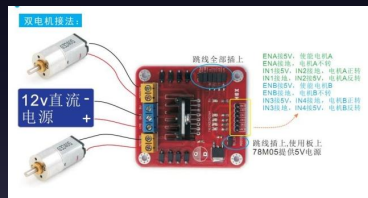


Figure : L298N 直流电机驱动板

L298N 直流电机驱动板

- 1 如果电机直接接 Arduino Uno 上电流不够，无法驱动
- 2 可控制电机正转反转，转速调节
- 3 分为控制模块与驱动模块，
 - 1 控制模块与单片机相连（包括供电），调控驱动模块的电流大小，方向
 - 2 驱动模块需另外供电以驱动电机

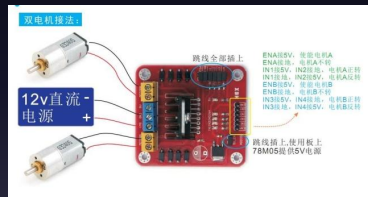


Figure : L298N 直流电机驱动板

针脚 & 接线

针脚 & 接线

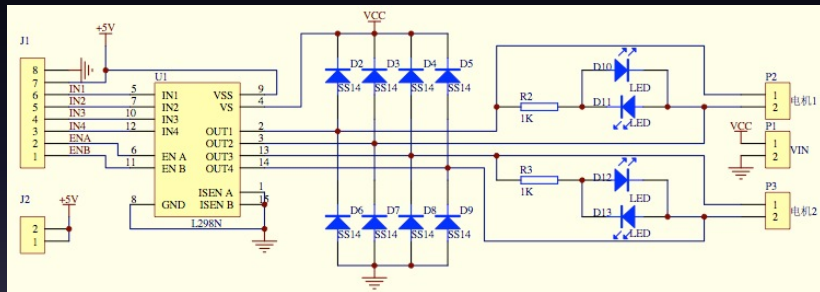


Figure : L298N 电路图

针脚 & 接线

- 1 电池接 P1 供电，左侧两电机并联至 P2，右侧两电机并联至 P3

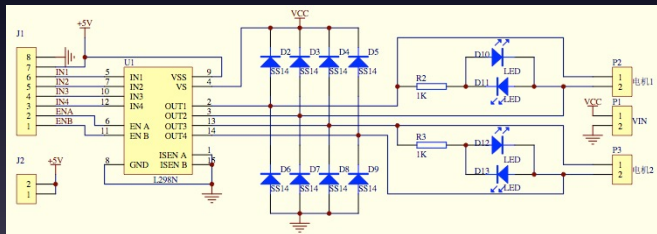


Figure : L298N 电路图

针脚 & 接线

- 1 电池接 P1 供电，左侧两电机并联至 P2，右侧两电机并联至 P3
- 2 IN1、IN2 控制左侧电机正反转接单片机的 digital pins，ENA 控制转速接 PWM pin，IN3、IN4、ENB 类似

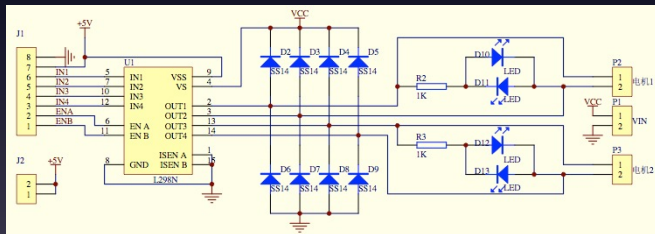


Figure : L298N 电路图

控制代码

```
1 digitalWrite(IN1, HIGH); //IN1 高电平, IN2 低电平电机正转
2 digitalWrite(IN2, LOW); //IN2 高电平, IN1 低电平电机反转
3
4 analogWrite(ENA, 255); //ENA 控制转速, 范围: 0-255
```

控制代码

```

1 digitalWrite(IN1, HIGH); //IN1 高电平, IN2 低电平电机正转
2 digitalWrite(IN2, LOW); //IN2 高电平, IN1 低电平电机反转
3
4 analogWrite(ENA, 255); //ENA 控制转速, 范围: 0-255

```

转速由摇杆的模拟信号值确定, 比如以 VR_x 为行进方向, VR_y 为左右转向, 可以使用以下函数:

$$\begin{cases} x = (VR_x - 512) \cdot \frac{255}{512} \\ y = (VR_y - 512) \cdot \frac{255}{512} \end{cases} \quad \begin{cases} LS = x + y \cdot \left(1 - \frac{x}{255}\right) \\ RS = x - y \cdot \left(1 - \frac{x}{255}\right) \end{cases}$$

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

DHT11 温湿度传感器

DHT11 温湿度传感器

- 1 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例

DHT11 温湿度传感器

- 1 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例
- 2 供电电压：3.3-5.5V DC

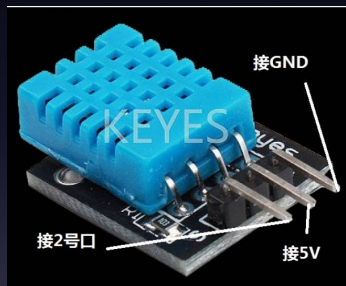


Figure : DHT11 温度湿度传感器

DHT11 温湿度传感器

- 1 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例
- 2 供电电压：3.3-5.5V DC
- 3 数字信号输出



Figure : DHT11 温度湿度传感器

DHT11 温湿度传感器

- 1 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例
- 2 供电电压：3.3-5.5V DC
- 3 数字信号输出
- 4 测量范围：湿度 20-90% RH，温度 0-50°C

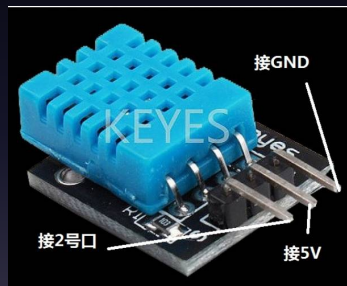


Figure : DHT11 温度湿度传感器

DHT11 温湿度传感器

- 1 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例
- 2 供电电压：3.3-5.5V DC
- 3 数字信号输出
- 4 测量范围：湿度 20-90% RH，温度 0-50°C
- 5 测量精度：湿度 $\pm 5\%$ RH，温度 $\pm 2^\circ\text{C}$



Figure : DHT11 温度湿度传感器

DHT11 温湿度传感器

- ① 小车可根据需要加装不同种类、数量的传感器。由于实验室可用传感器有限，仅以温度湿度传感器为例
- ② 供电电压：3.3-5.5V DC
- ③ 数字信号输出
- ④ 测量范围：湿度 20-90% RH，温度 0-50°C
- ⑤ 测量精度：湿度 $\pm 5\%$ RH，温度 $\pm 2^\circ\text{C}$
- ⑥ 分辨率：湿度 1% RH，温度 1°C

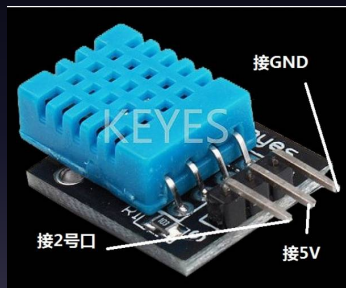


Figure : DHT11 温度湿度传感器

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

针脚 & 接线

针脚 & 接线

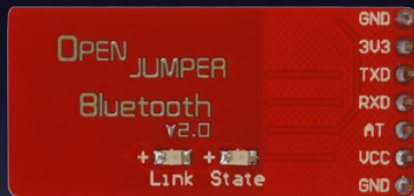
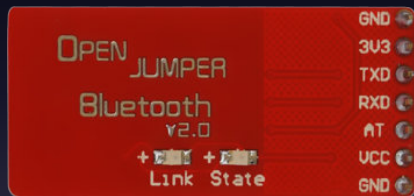


Figure : DHT11 温度湿度传感器

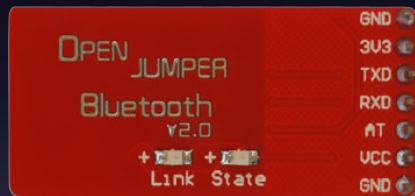
针脚 & 接线



1 GND: 接地

Figure : DHT11 温度湿度传感器

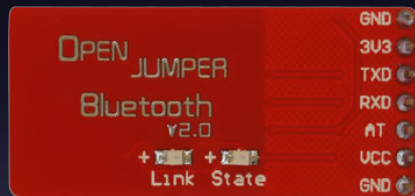
针脚 & 接线



- 1 GND: 接地
- 2 3V3: 3.3V 供电

Figure : DHT11 温度湿度传感器

针脚 & 接线



- 1 GND: 接地
- 2 3V3: 3.3V 供电
- 3 TXD: UART 接收端

Figure : DHT11 温度湿度传感器

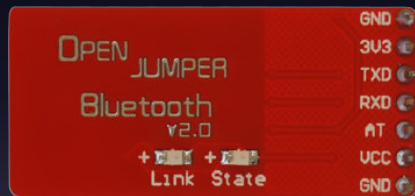
针脚 & 接线



- ① GND: 接地
- ② 3V3: 3.3V 供电
- ③ TXD: UART 接收端
- ④ RXD: UART 发送端

Figure : DHT11 温度湿度传感器

针脚 & 接线



- 1 GND: 接地
- 2 3V3: 3.3V 供电
- 3 TXD: UART 接收端
- 4 RXD: UART 发送端
- 5 AT: AT 模式控制端

Figure : DHT11 温度湿度传感器

针脚 & 接线

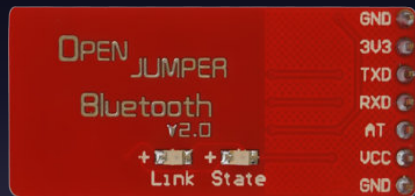


Figure : DHT11 温度湿度传感器

- ① GND: 接地
- ② 3V3: 3.3V 供电
- ③ TXD: UART 接收端
- ④ RXD: UART 发送端
- ⑤ AT: AT 模式控制端
- ⑥ VCC: 4-6V 供电输入端

工作模式

工作模式

① Master (主机)

工作模式

① Master (主机)

② Slave (从机)

工作模式

- 1 Master (主机)
 - 1 对外不可见
- 2 Slave (从机)

工作模式

- 1 Master (主机)
 - 1 对外不可见
- 2 Slave (从机)
 - 1 对外可见

工作模式

- 1 Master (主机)
 - 1 对外不可见
 - 2 主动连接
- 2 Slave (从机)
 - 1 对外可见
 - 2 被动连接

工作模式

工作模式

命令

AT+NAME=name

功能

给模块命名为 name

工作模式

命令

AT+NAME=name

AT+ROLE=1

功能

给模块命名为 name

设置主从模式，0 为从机，1 为主机

工作模式

命令

AT+NAME=name

AT+ROLE=1

AT+PSWD=0000

功能

给模块命名为 name

设置主从模式，0 为从机，1 为主机

设置配对密码为 0000

工作模式

命令

AT+NAME=name

AT+ROLE=1

AT+PSWD=0000

AT+UART=9600,0,0

功能

给模块命名为 name

设置主从模式，0 为从机，1 为主机

设置配对密码为 0000

设置波特率 9600

工作模式

命令

AT+NAME=name

AT+ROLE=1

AT+PSWD=0000

AT+UART=9600,0,0

AT+RMAAD

功能

给模块命名为 name

设置主从模式，0 为从机，1 为主机

设置配对密码为 0000

设置波特率 9600

清空配对列表

工作模式

命令

AT+NAME=name

AT+ROLE=1

AT+PSWD=0000

AT+UART=9600,0,0

AT+RMAAD

AT+BIND=2013,10,110025

功能

给模块命名为 name

设置主从模式, 0 为从机, 1 为主机

设置配对密码为 0000

设置波特率 9600

清空配对列表

设定从机的地址

工作模式

命令

AT+NAME=name

AT+ROLE=1

AT+PSWD=0000

AT+UART=9600,0,0

AT+RMAAD

AT+BIND=2013,10,110025

AT+CMODE=0

功能

给模块命名为 name

设置主从模式，0 为从机，1 为主机

设置配对密码为 0000

设置波特率 9600

清空配对列表

设定从机的地址

开启自动连接

获取从机地址

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看
- 2 实验室无此模块

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看
- 2 实验室无此模块
- 3 AT 高电平向蓝牙模块发送“AT+ADDR?” 模块会发送地址信息

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看
- 2 实验室无此模块
- 3 AT 高电平向蓝牙模块发送“AT+ADDR?” 模块会发送地址信息
- 4 Arduino 连蓝牙模块时与计算机的连接有冲突，蓝牙模块回应的地址信息计算机无法接收

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看
- 2 实验室无此模块
- 3 AT 高电平向蓝牙模块发送“AT+ADDR?” 模块会发送地址信息
- 4 Arduino 连蓝牙模块时与计算机的连接有冲突，蓝牙模块回应的地址信息计算机无法接收
- 5 但地址信息仍储存在 Arduino 缓冲区

获取从机地址

- 1 一般是将蓝牙接在一个 USB 模块连接电脑用专用软件查看
- 2 实验室无此模块
- 3 AT 高电平向蓝牙模块发送“AT+ADDR?” 模块会发送地址信息
- 4 Arduino 连蓝牙模块时与计算机的连接有冲突，蓝牙模块回应的地址信息计算机无法接收
- 5 但地址信息仍储存在 Arduino 缓冲区
- 6 可将该信息输出在显示屏上

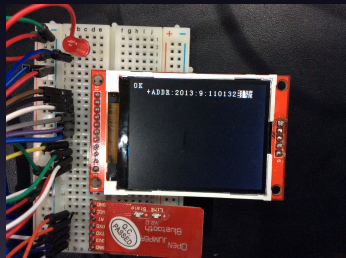


Figure : 获取蓝牙模块地址

数据包

数据包

数据包示例

```
bl155r155e
```


数据包

数据包示例

bl155r155e

- 1 数据包传输过程中可能全部丢失或部分丢失

数据包

数据包示例

bl155r155e

- 1 数据包传输过程中可能全部丢失或部分丢失
- 2 以 b 作为数据包其实， e 作为数据包结束

数据包

数据包示例

bl155r155e

- 1 数据包传输过程中可能全部丢失或部分丢失
- 2 以 b 作为数据包其实， e 作为数据包结束
- 3 l 之后的 int 控制左侧电机转速， r 同理

数据包

数据包示例

bl155r155e

- 1 数据包传输过程中可能全部丢失或部分丢失
- 2 以 b 作为数据包开始， e 作为数据包结束
- 3 l 之后的 int 控制左侧电机转速， r 同理
- 4 字符串处理

Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

ILI9225 介绍

ILI9225 介绍

1 176×220, RGB

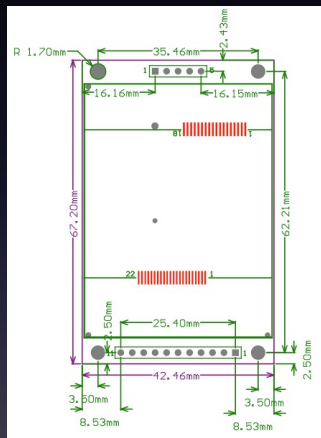


Figure : 模块尺寸图

ILI9225 介绍

- 1 176×220, RGB
- 2 有 SD 卡电路及接口

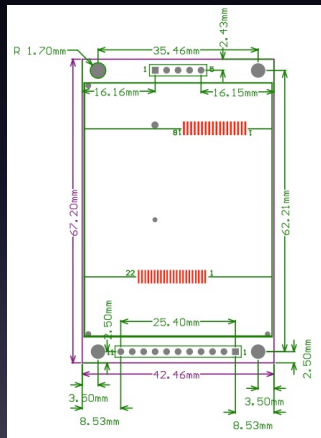


Figure : 模块尺寸图

ILI9225 介绍

- 1 176×220, RGB
- 2 有 SD 卡电路及接口
- 3 刷新速率过慢

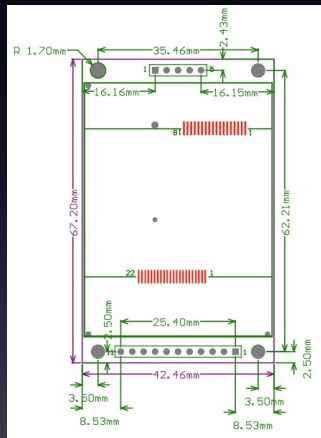


Figure : 模块尺寸图

针脚 & 接线

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地
4	NC	悬空

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地
4	NC	悬空
5	NC	悬空

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地
4	NC	悬空
5	NC	悬空
6	LED	背光亮度控制, 可 PWM 调节

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地
4	NC	悬空
5	NC	悬空
6	LED	背光亮度控制, 可 PWM 调节
7	CLK	串行 SPI 时钟信号

针脚 & 接线

No.	针脚	功能
1	VCC	正极, 5V 或 3.3V
2	GND	接地
3	GND	接地
4	NC	悬空
5	NC	悬空
6	LED	背光亮度控制, 可 PWM 调节
7	CLK	串行 SPI 时钟信号
8	SDI	串行 SPI 数据输入信号
9	RS	命令 (RS=0)/参数 (RS=1) 选择
10	RST	液晶复位信号
11	CS	液晶片选信号

SPI

SPI

1 MOSI: 主机向设备发送数据

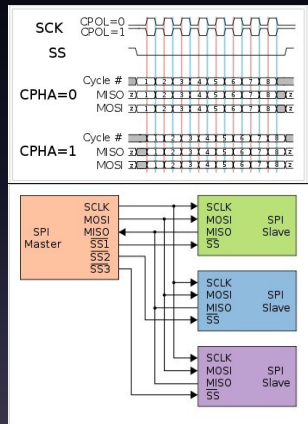


Figure : 时序图 & 多外围设备接法

SPI

- 1 MOSI: 主机向设备发送数据
- 2 MISO: 设备向主机发送数据

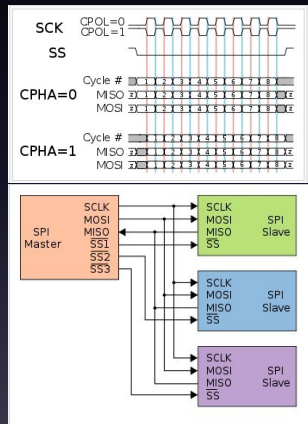


Figure : 时序图 & 多外围设备接法

SPI

- 1 MOSI: 主机向设备发送数据
- 2 MISO: 设备向主机发送数据
- 3 SCK: 时钟信号，作为数据传输校准

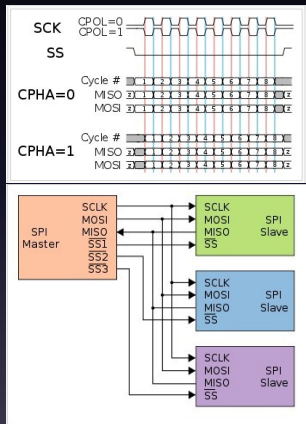


Figure : 时序图 & 多外围设备接法

SPI

- 1 MOSI: 主机向设备发送数据
- 2 MISO: 设备向主机发送数据
- 3 SCK: 时钟信号, 作为数据传输校准
- 4 SS: 设备选择

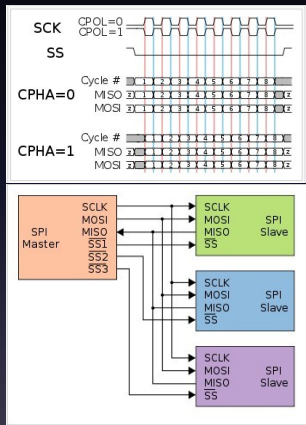


Figure : 时序图 & 多外围设备接法

软件

```
1 include "UTFT.h"//调用 UTFT 库
2 extern uint8_t SmallFont[];//ASCII 字体
3 UTFT myGLCD(QD220A,10,9,13,12,11);
4 //创建 UTFT 对象
5 //构造函数参数依次为显示器型号, RST, CS, CLK, SDI, RS
6 void setup(){
7     myGLCD.InitLCD();//屏幕初始化
8     myGLCD.setFont(SmallFont);//设置字体大小
9 }
10 void loop(){
11     myGLCD.setColor(255, 255, 255);//设置颜色
12     myGLCD.print("Hello World!", CENTER, 62);
13     //显示 "Hello World", x 坐标, y 坐标
14 }
```


Outline

- 1 成品简介
- 2 Arduino 简介
- 3 具体配置介绍
 - 1 摇杆模块
 - 2 电机模块
 - 3 传感器模块
 - 4 蓝牙模块
 - 5 显示模块
- 4 可改进之处

可改进之处

可改进之处

- 1 Arduino Uno 性能不足易死机

可改进之处

- 1 Arduino Uno 性能不足易死机
 - 1 接受发送并行，换为四个单片机

可改进之处

- ① Arduino Uno 性能不足易死机
 - ① 接受发送并行，换为四个单片机
 - ② ARM 单片机，X86 单片机

可改进之处

- 1 Arduino Uno 性能不足易死机
 - 1 接受发送并行，换为四个单片机
 - 2 ARM 单片机，X86 单片机
- 2 UTFT 库非硬件 SPI

可改进之处

- 1 Arduino Uno 性能不足易死机
 - 1 接受发送并行，换为四个单片机
 - 2 ARM 单片机，X86 单片机
- 2 UTFT 库非硬件 SPI
- 3 SD 卡