

# 基于 Arduino 的 RFID 识别实验

转载

dasuo7626 于 2015-12-07 12:45:00 发布 2411 收藏 15

文章标签: 嵌入式

原文链接: <http://www.cnblogs.com/luoxu34/p/5025760.html>

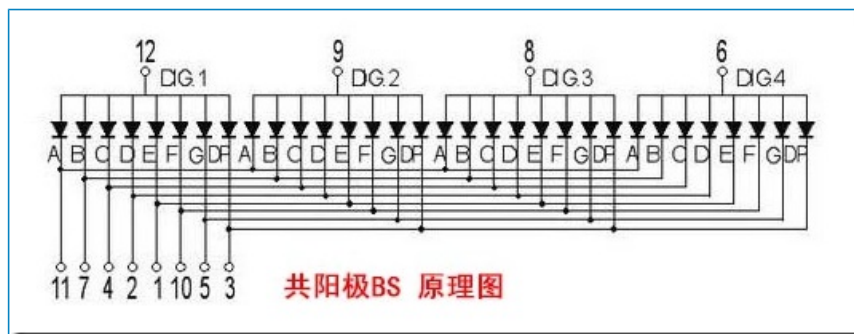
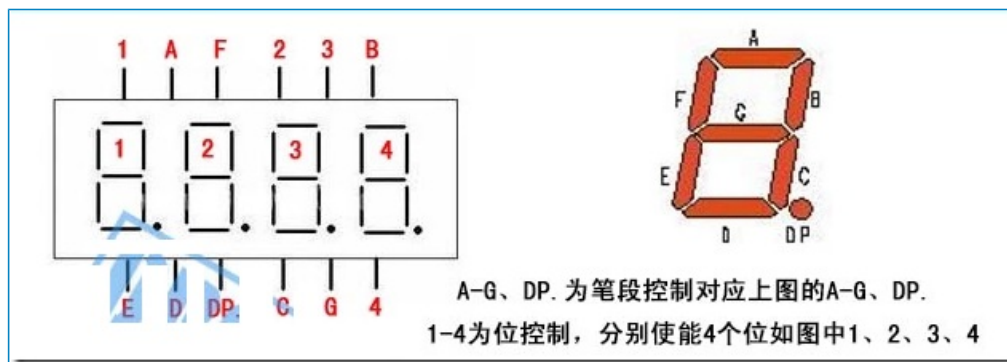
版权

餐前小菜

我买的 Arduino 是入门套件 + RFID学习套件，某宝包邮130+，买来也将近1个月了。

有的店铺需要确认收货后去向客服索取配套资料的链接。其实网上的资料都大致一样，有的卖家只是修改了文件的名字，有的甚至修改了代码却没有保留最初的源代码，真是可恶。资料不能公开分享就算了，还把源码给改了。

今天我还是试了4位数码管的实验，发现给的资料和代码跟发的数码管不是一个型号，以至于显示一堆乱码。卖家发的数码管是 **3461BS** (0.36英寸 4位LED数码管 共阳极)。不得不自己到网上找的原理图并修改代码。下边分享给大家：



四位共阳极数码管，位控制端(1-4)给高电平使能相应的位，笔段控制端(A-G、DP)给低电平可点亮。

记得要加限流电阻哦！

数码管的原理图我还是看的懂的。我所异者，是做一天和尚敲一天钟，不思进取，敷衍了事，不愿追求把工作做到极致的态度。

实验由来

2015年12月04日（周五）上午 博士的智能卡实验——RFID识别实验，基于51单片机；

我们的实验用的读写器是 RFID-RC522 模块，刚好和我买的 Arduino RFID 套件里的是同一款；

实验时候并没有完成，因为在烧写程序的时候一直烧不进去，好吧，下午在办公室的时候我们博士说有一块单片机是有问题的，❤️ TMD有那么巧吗 😒😒

中午开始我就开始阅读 Arduino 套件配套的 RFID 实验资料，打算把这个实验做完，只是我把上位机换成 Arduino 而已。

## 实验目的

---

1. 了解 RFID 的基本概念和射频卡的工作原理；
2. 熟悉 Arduino 和 RC522RFID 模块的使用；
3. 测试 RC522 模块对卡的识别。

## 实验材料

---

1. 计算机一台；
2. Arduino 一块；
3. USB-串口线一根；
4. RFID-RC522 模块一个；
5. 标准S50空白卡一张；
6. S50异形卡一个（钥匙扣形状）；
7. 直排、弯排插针各一条；
8. 杜邦线一排。

买来的射频 IC卡感应模块就像下面这样，包邮也就14块左右：



买来的时候排插并没有焊上，所以必须自己焊，虽然我没有焊接的工具，不过借着这次实验我还是焊上了。下面是我第二次焊接的成果：



### Mifare 1 s50 感应式IC卡

- ◇ 芯 片： Philips Mifare 1 S50
- ◇ 存储容量： 8Kbit ,16个分区,每分区两组密码
- ◇ 工作频率： 13.56MHZ
- ◇ 通讯速度： 106Kboud
- ◇ 读写距离： 2.5—10CM
- ◇ 读写时间： 1-2MS
- ◇ 工作温度： -20℃-85℃
- ◇ 擦写次数： >100000次
- ◇ 数据保存： >10年
- ◇ 规 格： 0.87×85.5×54/ 非标卡
- ◇ 封装材料： PVC、PET、0.13铜钱
- ◇ 封装工艺： 超声波自动植线/自动碰焊
- ◇ 制作标准： ISO 14443, ISO 10536
- ◇ 应用范围： 企业/校园一卡通、公交储值卡、高速公路收费、停车场、小区管理等

### 实验原理

## （一）RFID系统组成

RFID 技术利用无线射频方式在阅读器和射频卡之间进行非接触双向数据传输，以达到目标识别和数据交换的目的。最基本的 RFID 系统由三部分组成：

1. 标签(Tag, 即射频卡)：由耦合元件及芯片组成，标签含有内置天线，用于和射频天线间进行通信。
2. 阅读器：读取(在读写卡中还可以写入)标签信息的设备。
3. 天线：在标签和读取器间传递射频信号。

## （二）工作原理

**MCU**通过对读卡器芯片内寄存器的读写来控制读卡器芯片，读卡器芯片收到MCU发来的命令后，按照非接触式射频卡协议格式，通过天线及其匹配电路向附近发出一组固定频率的调制信号(13.56 MHz)进行寻卡，若此范围内有卡片存在，卡片内部的LC谐振电路(谐振频率与读卡器发送的电磁波频率相同)在电磁波的激励下，产生共振，在卡片内部电压泵的作用下不断为其另一端的电容充电，获得能量，当该电容电压达到2 V时，即可作为电源为卡片的其他电路提供工作电压。当有卡片处在读卡器的有效工作范围内时，**MCU**向卡片发出寻卡命令，卡片将回复卡片类型，建立卡片与读卡器的第一步联系，若同时有多张卡片在天线的工作范围内，读卡器通过启动防冲撞机制，根据卡片序列号来选定一张卡片，被选中的卡片再与读卡器进行密码校验,确保读卡器对卡片有操作权限以及卡片的合法性，而未被选中的则仍然处在闲置状态，等待下一次寻卡命令。密码验证通过之后，就可以对卡片进行读写等应用操作。

## （三）RC522模块

模块的射频读写芯片应该是飞利浦的 MF RC522，在上一篇文章我应该已经附上了英文和中文的手册。

MF RC522提供了3种接口模式：高达10 Mb/s的SPI、I2C总线模式(快速模式下能达400 kb/s，而高速模式下能达3.4 Mb/s)、最高达1228.8 kb/s的UART模式。买来的模块采用了第一种模式——**四线制SPI**，通信中的时钟信号由 Arduino 产生，MF RC522芯片设置为从机模式，接收来自 Arduino 的数据以设置寄存器，并负责射频接口通信中相关数据的收发。

**数据的传输路径为：Arduino 通过 MOSI 线将数据发到 MF RC522，MF RC522 通过 MISO 线发回至 Arduino。**

## 实验设计

---

### RC522模块各引脚功能

SDA -- 串行数据线（IIC接口时的I/O线）；在SPI接口中为NSS（从机标志管脚）；

SCK -- 连接MCU的SCK信号；

MOSI -- MCU输出，RC522接收（即主设备输出，从设备输入）；

MISO -- RC522输出，MCU接收（即从设备输出，主设备输入）；

IRQ -- 中断请求输出；

GND -- 接地；

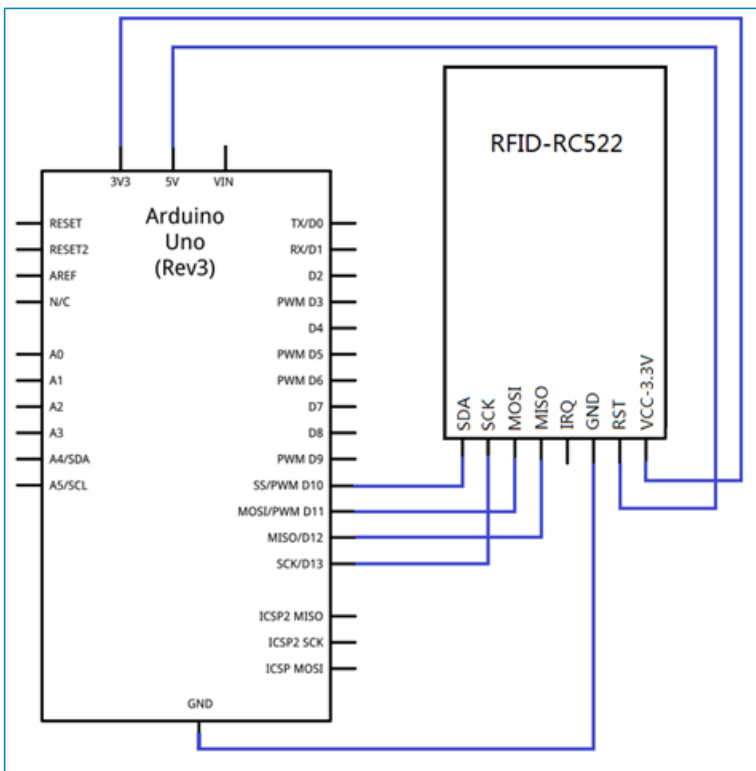
RST -- 复位;

3.3V -- VSS, 工作电压, 若使用的事5V的MCU, 注意分压。

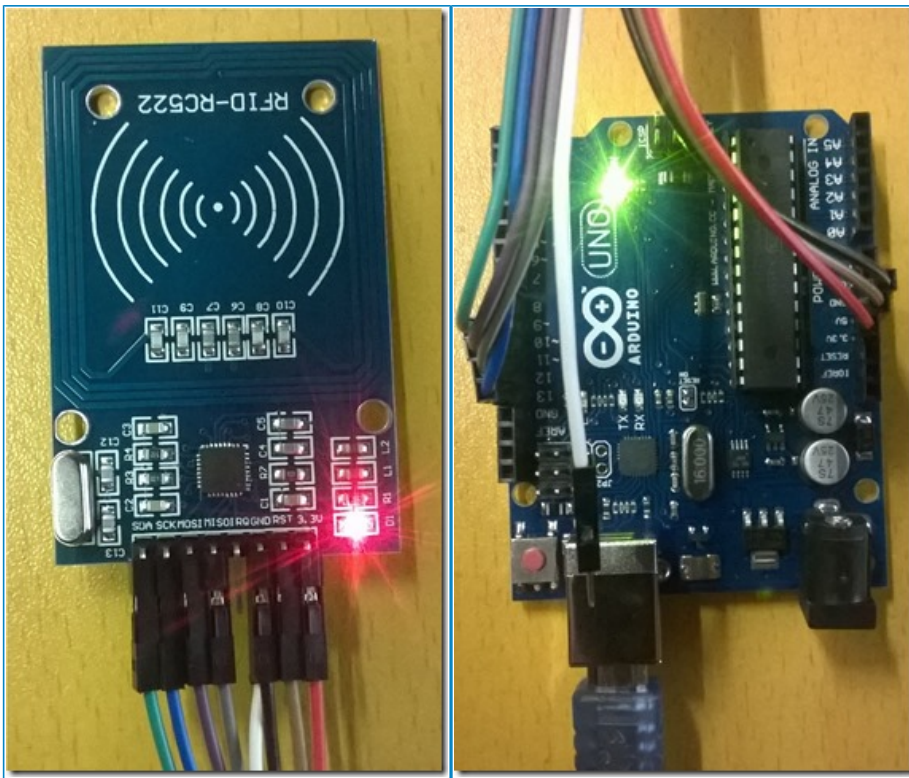
管脚对应关系:

RFID模块	Arduino UNO
VCC	3.3V
RST	5V
GND	GND
MISO	12
MOSI	11
SCK	13
NSS	10
IRQ	(不接)

电路图



实物连接图



代码的设计，我会放到下一篇文章中讲。

## 实验内容

---

- 1、实现利用IC卡控制LED灯闪烁。
- 2、实现利用七段数码管显示IC卡的类型，并记录。
- 3、实现在七段数码管上以十六进制显示IC卡的序列号，并记录。
- 4、从串口打印IC卡的类型和序列号。
- 5、测试最大的识别距离。

## 思考题

---

- 1、完成对IC卡的数据读写，需要哪几个步骤？

答：四个步骤：寻卡 → 防冲突 → 选卡 → 三次互相确认 → 读写

- 2、请分别写出实验材料中标准S50卡、S50钥匙卡的类型和序列号。

答：S50标准卡：Type 0400 和 SN FDC2E3D5

S50异形卡：Type 0400 和 SN D8163100

- 3、当多张卡在一起时，能否正确识别其卡号？并请说明其原因。

答：在这个实验中不行。在这个实验系统中，阅读器会先执行寻卡操作并返回卡的类型，之后才执行防冲突操作。因为有多张卡处在阅读器的可读范围，防冲突函数返回值为 MI\_ERR，也就不能继续执行选卡和读写的操作，也就获取不到卡的序列号。

#### 实验中的观察：

当我将两张卡同时慢慢靠近阅读器的过程中，阅读器首先识别出了卡的类型和其中的某一种卡，并在串口上打印了卡的类型和序列号。接着串口只会一直打印卡的类型而不会打印的卡的序列号。有时也会只打印卡类型而没有序列号。

#### 分析：

在慢慢靠近的过程中，阅读器得到的是不确定的信号，偶尔碰巧识别出了其中一张卡。但是之后，两张卡相互干扰，以至都不能工作。

#### 那么在多张卡的情况下，RFID阅读器能否同时处理多个标签呢？

答案是肯定的，TI 的 13.56MHz 系统每秒钟能处理大约50张标签。回顾一下上一篇文章讲到的MF卡的逻辑结构，其中有一个模块叫防冲突模块：

如果有多张Mifare 1卡片处在读写器的天线的工作范围之内时，防冲突模块的防冲突功能将被启动工作：根据卡片的序列号来选定一张卡片。被选中的卡片将直接与读写器进行数据交换，未被选择的卡片处于等待状态，准备与读写器进行通信。

既然有防冲突模块，那么为什么我们用的同样是 MF1 S50 的卡，还是无法识别呢？

原因就在 RFID 阅读器的防冲撞机制上。如果我们的读卡设备使用了防冲撞机制，那么在冲撞产生后，RFID系统会根据冲撞算法，将不同的卡片分时响应，并且已经响应的卡片将暂时被休眠(SLEEP)，未被选择的卡片处于等待状态。这点在上一篇文章中我仅仅是提到算法的名字。

**防冲撞机制**使得同时处于读写区内的多张卡的正确操作成为可能，只用算法编程，读头即可自动选取其中一张卡进行读写操作。这样既方便了操作，也提高了操作的速度。

如果在同一时间段内有多于一个的 VICC 或 PICC 同时响应，则说明发生冲撞。RFID的核心是防冲撞技术，这也是和接触式 IC 卡的主要区别。ISO 14443-3 规定了 TYPE A 和 TYPE B 的防冲撞机制。二者防冲撞机制的原理不同:前者是基于位冲撞检测协议，而 TYPE B 通过系列命令序列完成防冲撞；ISO 15693 采用轮寻机制、分时查询的方式完成防冲撞机制，在标准的第三部分有详细规定。

4、请测试 RC522 模块是否能够识别羊城通、校园一卡通等非接触式IC卡？若能识别，请写出其类型及序列号？

答：RC522 模块还能识别羊城通、校园一卡通和银行卡，发现类型都是 0400，属于 MFOne-S50 卡。

某校园一卡通

Card type: MFOne-S50

The card's number is: F68F8EAB

某羊城通

Card type: MFOne-S50

The card's number is: 95AF4B27

某银行卡

Card type: MFOne-S50

The card's number is: EF660184

第二代身份证的载波频率是13.56MHz，副载波频率为847KHz，记录的信息有身份证号/姓名/性别/居住地址/照片等。

但是RC522并不能识别它，实验的白卡是 ISO 14443 TYPE A，而第二代身份证是B型卡，协议是 ISO 15693B

## 实验结果

### led 闪烁

下次附图。。。

### 数码管显示

下次附图。。。

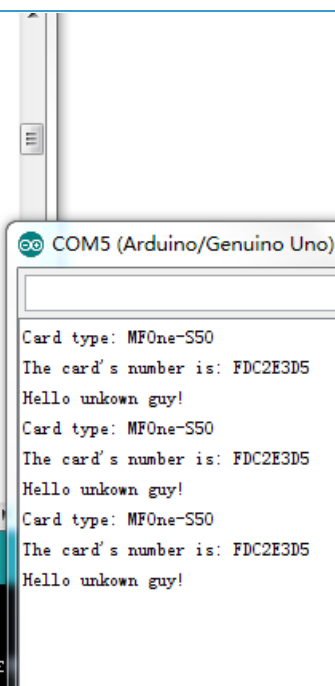
### 串口打印

```
// Search card, return card types
status = MFRC522_Request(PICC_REQIDL, str);
if (status != MI_OK)
{
    return;
}

// Show card type
ShowCardType(str);

//Prevent conflict, return the 4 bytes Serial number of the card
status = MFRC522_Anticoll(str);

// str[0..3]: serial number of the card
// str[4]: XOR checksum of the SH.
if (status == MI_OK)
```



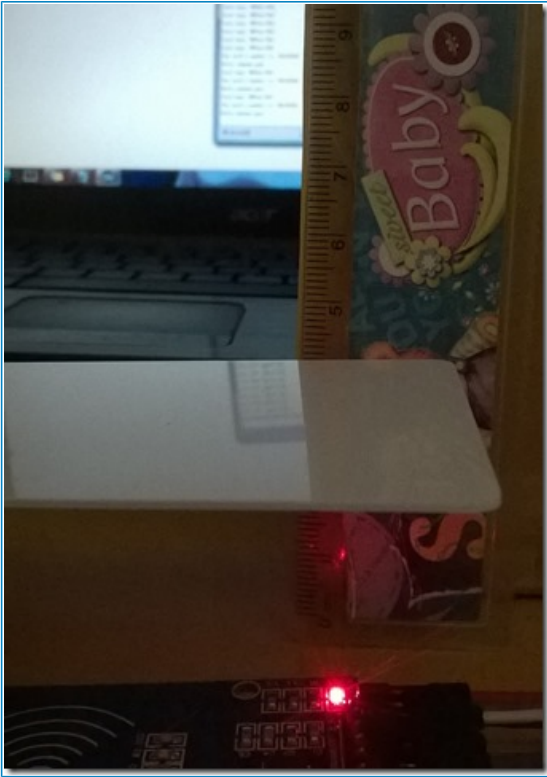
上传成功。

项目使用了 4,440 字节，占用了 (13%) 程序存储空间。最大为 32,256 字节。  
全局变量使用了342字节，(16%)的动态内存，余留1,706字节局部变量。最大为2,048字



## 识别距离

我粗略测量，发现白卡的识别需要在 **4 cm** 内，在 7 cm 的位置还是读不到卡片的。至于加大工作电流，还是不能增加通信距离。用我的校园卡和白卡一起测试，几乎都是只能识别校园卡。所以我以为校园卡的线圈比较好，但是单独识别，距离依然是 4 cm。钥匙扣的识别距离为**2.5 cm** 左右。



注意 识别的时候要先释放手上的静电，阅读器的周围不要有其他电流干扰。

## 实验总结

广州大学城的校园一卡通，能当交通卡、单车租用卡、图书馆借书卡、饭堂饭卡、学生卡等。IC 卡的识别，是对于卡片读写和一卡多用的的前提。

转载于:<https://www.cnblogs.com/luoxu34/p/5025760.html>