

二维码门禁预约管理系统

本文使用树莓派 DIV 搭建一个二维码门禁预约管理系统。起源是学校里需要对活动室进行人员管理，保护学校固定资产。原先学校的活动室安装的门禁系统是 IC 卡片式的，由于时间较长，管理不当，相当一部分的 IC 卡丢失，导致活动室被迫只能强制每天直接开放。所以，经过学校教工会商量，由学生外包开发基于二维码的门禁预约系统，而我是项目经理。

接手项目后，我将工程分为三个模块，微信模块（老师要求的）、网站预约模块、硬件控制模块。用户通过微信或浏览器进入网站，在网站上预约有关于使用某个活动室的情况，将信息保存进入数据库并提供给用户一个二维码。到了预约的时间范围之内，用户携带二维码到活动室控制前，扫描二维码，硬件控制模块读取二维码内容，如果二维码是本系统的，则到数据库中查询相关预约，如果当前时间与预约的时间符合，则开门。

在开始之前需要准备这样几个东西：树莓派开发板一个、电源箱一个（输出 12V）、电磁锁一个（12V 220MA）、继电器一个（5V 驱动）、树莓派官方摄像头一个（NoIRCamera）、装有树莓派原装系统 SD 卡一张、面包板（建议使用焊锡直接将线焊起来）、线若干。

先放上 Github 地址：<https://github.com/leafspace/CSLG-Access-Reservation-System>。

首先我们需要一个数据库，这个数据库用于保存有关于预约方面的信息。我在这个工程中使用的是 SQL Server，原因是与学校的数据库兼容。表结构如下：

```
CREATE TABLE [dbo].[Users](
    [user_id] [int] NOT NULL,
    [userName] [varchar](20) NOT NULL,
    [password] [varchar](16) NOT NULL,
    [wechat_id] [varchar](16) NULL,
    [phone_number] [varchar](13) NULL,
    [identity_number] [varchar](9) NULL,
    [is_temporary] [bit] NULL,
    [is_manager] [bit] NULL,
    [information] [varchar](max) NULL
)
CREATE TABLE [dbo].[Reservations](
    [reservation_Id] [int] NOT NULL,
    [user_id] [int] NULL,
    [room_id] [int] NULL,
    [valid] [bit] NOT NULL,
    [lock] [bit] NOT NULL,
    [year] [int] NULL,
    [month] [int] NULL,
    [day] [int] NULL,
    [start] [int] NULL,
    [finish] [int] NULL,
```

```

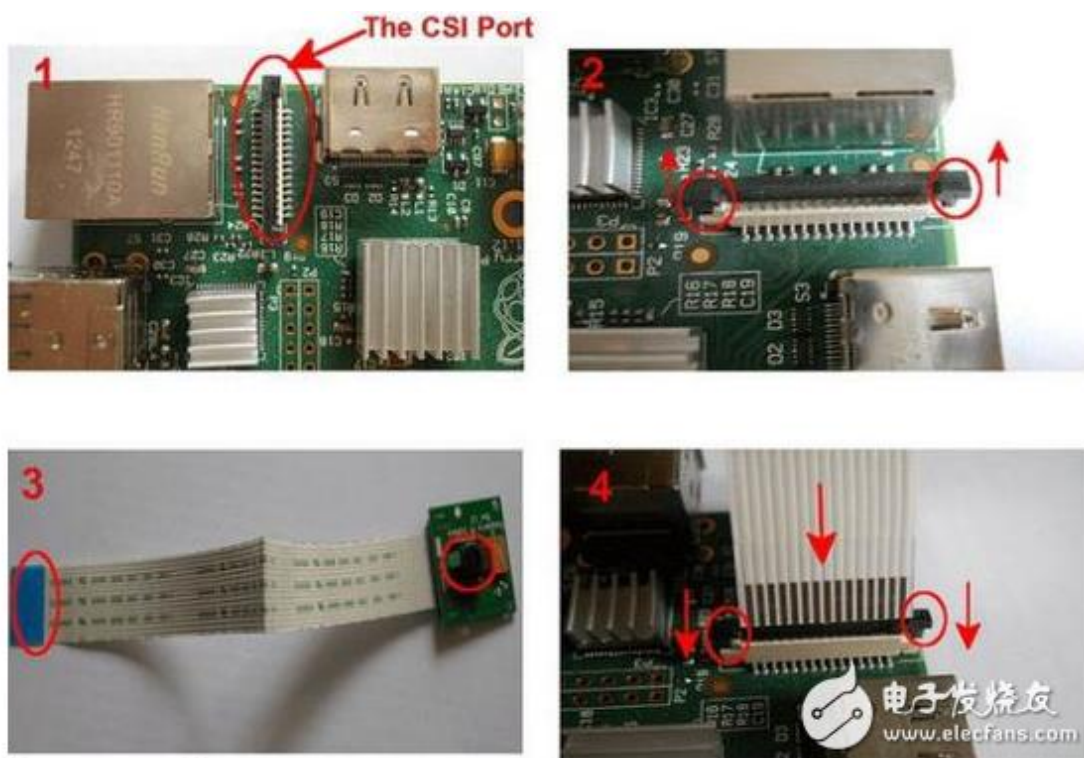
[qr_location] [varchar](50) NULL,
[information] [varchar](max) NULL
)
CREATE TABLE [dbo].[ActivityRooms](
[room_Id] [int] NOT NULL,
[information] [varchar](max) NULL
)

```

其中 Reservations 表中的 year、month、day、start、finish 就表示了用户某个预约的预约时间，(start 和 finish 都为三位数或四位数，例如开始时间 14：30，则 start 为 1430)。用户每进行一次预约就在 Reservations 表中新增一条记录。然后将其 reservation_Id 添加一段密钥生成二维码，生成二维码使用的是 zxing 库。下载地址：<http://www.oschina.net/p/zxing/>。

然后就是搭建部署预约网站，这个在此不多讲，因为当前开发方案有很多，网站烂大街了，我这边使用的是 JSP/Servlet 的形式开发的。

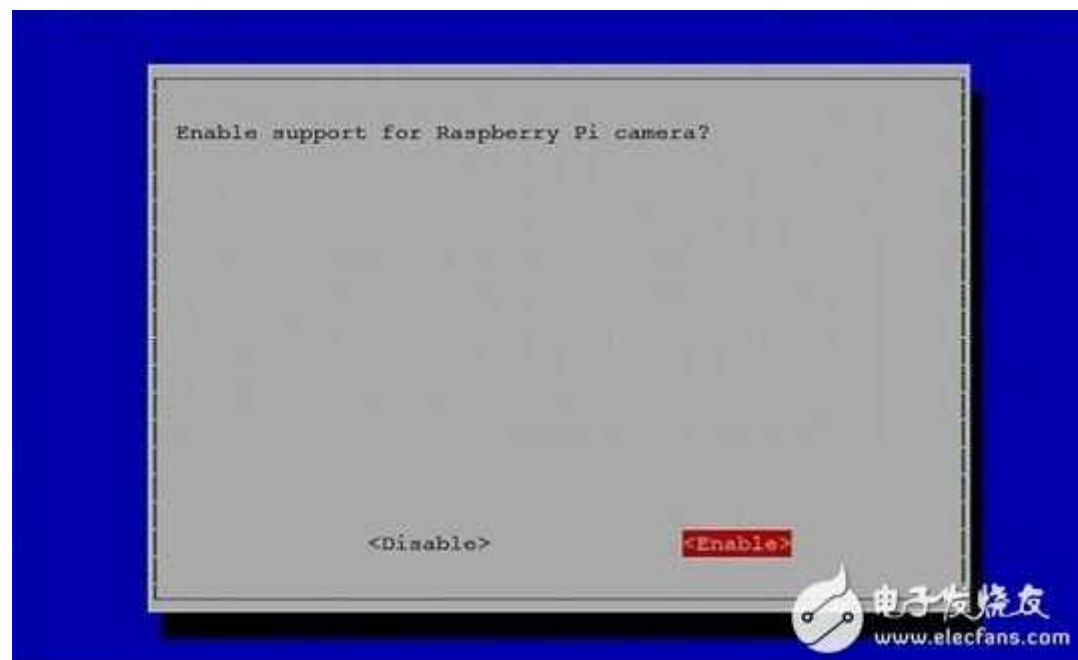
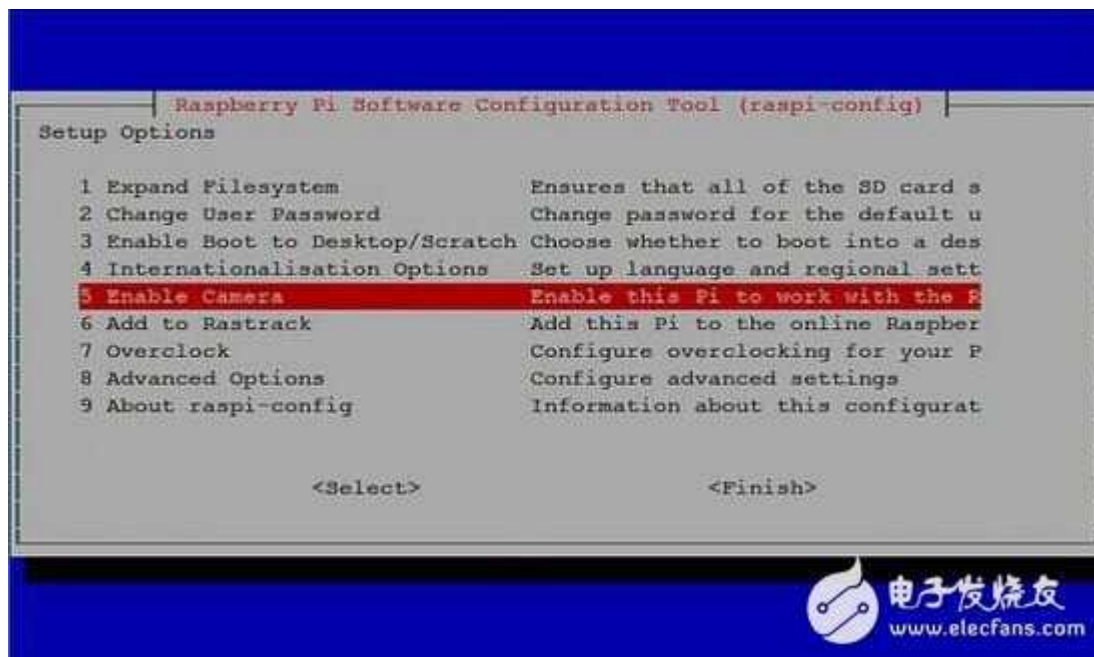
然后就是组装硬件，先将摄像头穿过亚克力板插进树莓派（蓝色部分朝网络接口）。如下图（来自网络）：



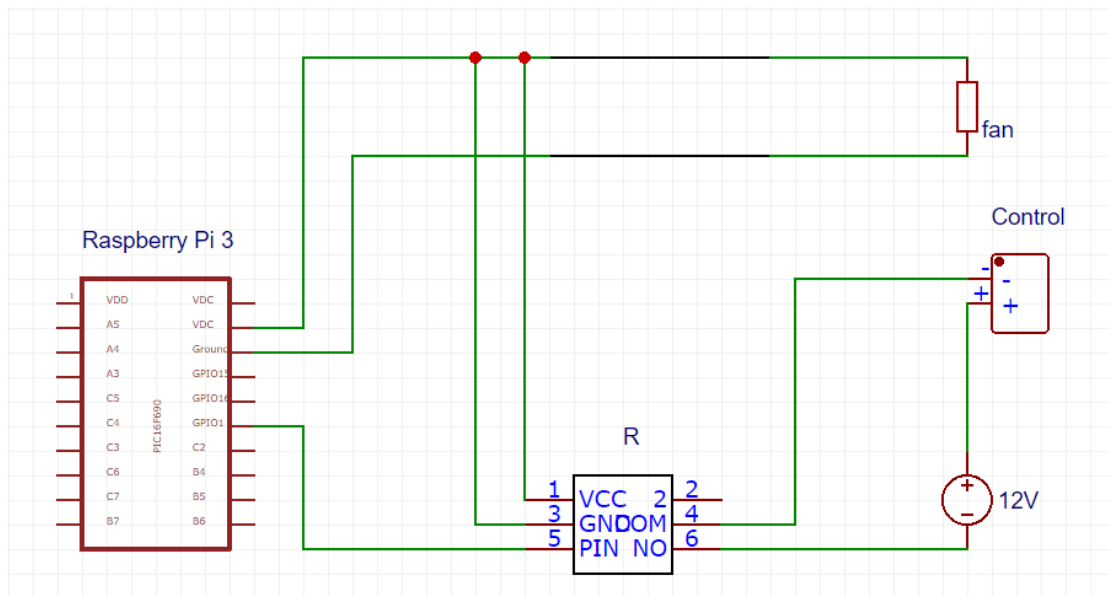
插入摄像头后还无法直接使用，需要在系统内进行配置。以树莓派原装系统为例：

- 进入命令行窗口，输入 \$sudo raspi-config
- 按动键盘上下键到 “Enable Camera”，按回车键，将其设为 Enable。
- 使用 \$sudo raspistill -o image.jpg 来拍照测试

如下图（来自网络）：



摄像头安装完成之后开始组装门禁控制系列其他硬件，先用一张原理图来说明接线结构，如下图：



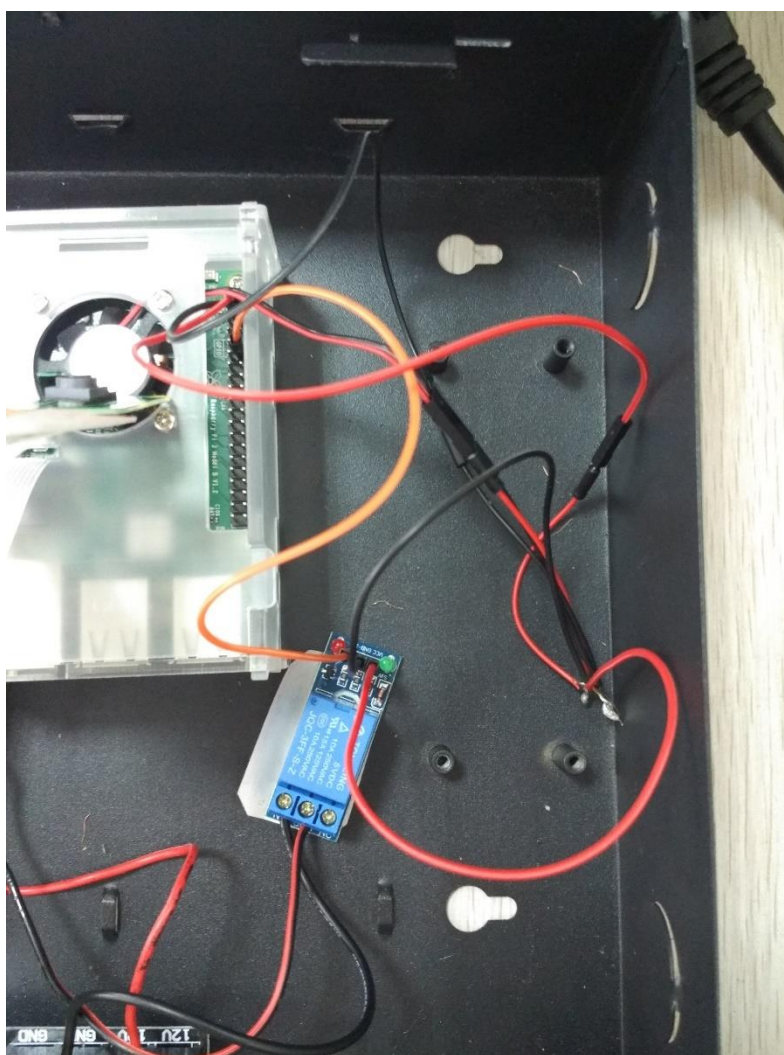
这张图上，从左至右，从上至下分别为：树莓派 3b，继电器，树莓派风扇，电磁锁，电源。

树莓派总共使用了三个引脚，电源脚、地线脚、GPIO1 号脚，其中，电源脚与地线脚同时为风扇，继电器供电。（注：图中有稍许错误，继电器的 GND 脚（3 号）应该与树莓派的地线脚相连）。GPIO1 号脚与继电器的 PIN（5 号）相连，这条线用于输出控制信号，绝对不能弄错。

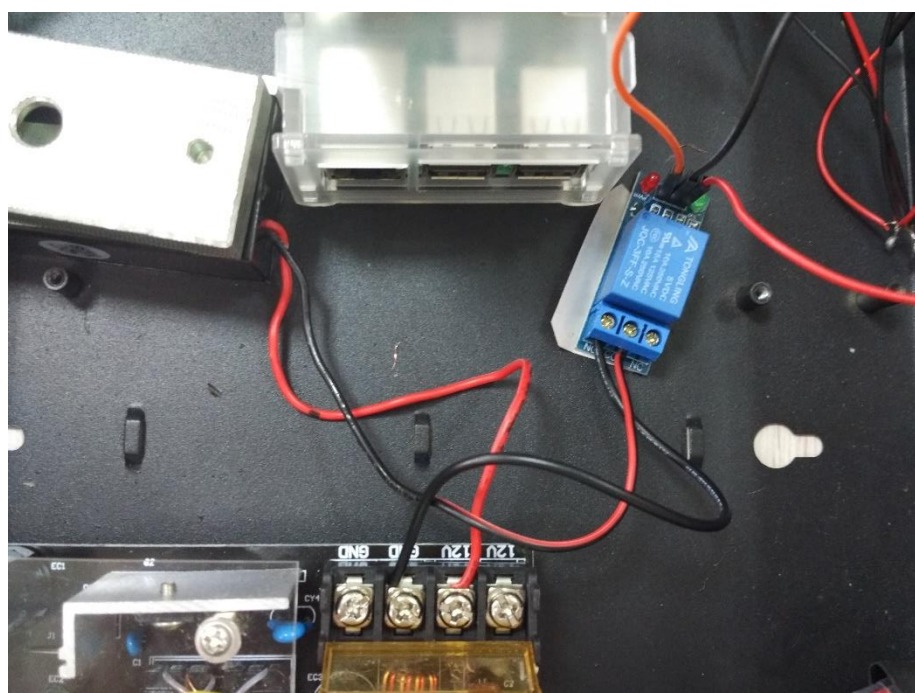
继电器使用了五个针脚（共 6 根），DOM 脚连电子锁的负极，NO 脚接交流电的负极。

电磁锁的正级与电源的正级相连。

我在实际操作的时候将由树莓派提供电源部分的线焊接起来了，所以没有用面包板。见下图：



这是继电器与树莓派相连的部分，下面这个是继电器与其余模块相连的部分：



然后就可以开始编写程序控制。

这里我是使用 java 编写的程序，使用 java 对树莓派针脚控制就需要使用 PI4J 库，下载地址：<http://pi4j.com/>。

程序逻辑如下：

1) 设置 GPIO1 号门为持续高电平：

```
final GpioController gpioController = GpioFactory.getInstance();
    final GpioPinDigitalOutput doorController =
gpioController.provisionDigitalOutputPin(RaspiPin.GPIO_01, "MyControl",
PinState.HIGH);
    doorController.setShutdownOptions(true, PinState.LOW);
```

2) 拍照，将照片保存在一个固定的文件夹下：

```
private static String path = "/home/pi/qrCode/image.jpg";
    private static String[] cmdOrder = {"sh", "-c", "raspistill -w 500 -h
500 -o " + path};

    public static boolean TakePhoto() {
        try {
            Process process = Runtime.getRuntime().exec(cmdOrder);
            process.waitFor();
            InputStream inputStream = process.getInputStream();
            BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(inputStream));
            String line = null;
            while((line = bufferedReader.readLine()) != null) {
                System.out.println("Information : (Take photo) " + line);
            }
        } catch (InterruptedException exception) {
            return false;
        } catch (IOException exception) {
            return false;
        }
        return true;
    }
```

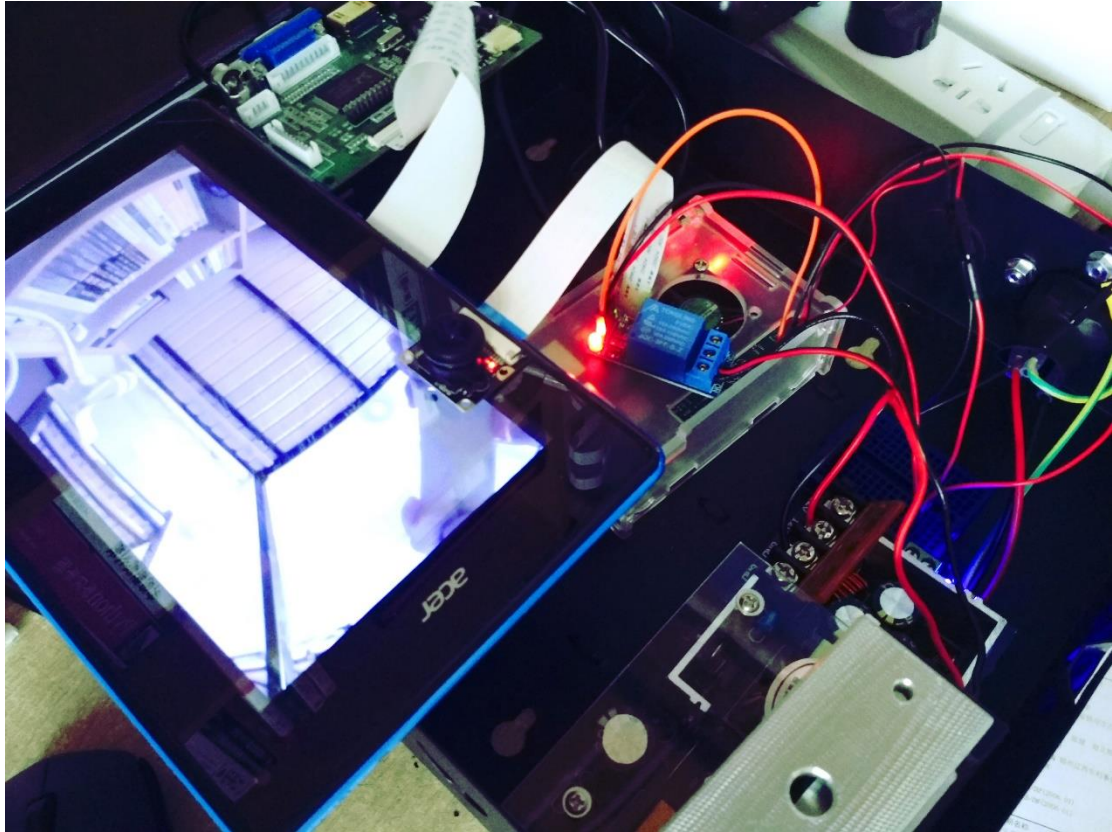
3) 使用 ZXing 库解析二维码，如果成功则跳转到 3

4) 开门，设置 GPIO1 号脚的电平为低电平（让电磁铁断电，失去磁性），系统等待 3 秒，3 秒后自动设置回高电平（关门）

```
public static void OpenDoor(GpioPinDigitalOutput doorController) {
    doorController.setState(PinState.LOW);
    try {
        Thread.sleep(3000);
```

```
    } catch (InterruptedException exception) {  
        exception.printStackTrace();  
    } finally {  
        doorController.setState(PinState.HIGH);  
    }  
}
```

最后来张部署后的全家福：



视频演示链接：

http://v.youku.com/v_show/id_XMjg3MjY2ODAxMg==.html?spm=a2h3j.8428770.3416059.1