

基于 Android 的可联网 LED 照明灯的设计

刘仁贵, 樊英杰, 党建林, 苏智华

(西北工业大学明德学院, 西安 710124)

摘要: 智能家居是指通过物联网技术将各种设备连接到一起, 物联网技术的迅速发展促进了智能家居的发展; 设计了一种基于 Android 的可组网的 LED 照明灯, 该 LED 照明灯以基于 Android 的智能电子设备为控制终端, 通过 socket 通信的方法实现与 LED 照明灯之间的通信, 达到对照明灯的开关以及亮度调节的目的; LED 照明灯采用 STM32F030F4P6 微处理器作为控制芯片, 选用高效率、高性能的 TPS92515 作为大功率 LED 的驱动, 并采用 USR-WIFI232-B2 作为 WIFI 通信模块; LED 照明灯的亮度通过微处理器输出的 PWM 信号进行调节, 不仅可以实现亮度从 0%~100% 的线性调节, 而且解决了普通照明灯在开灯和关灯时, 瞬间点亮和瞬间熄灭, 对人眼的冲击问题; 采用该方法设计的照明灯实用性强, 具有较好的推广价值。

关键词: 智能家居; Android; Socket; LED 照明; WIFI

Design of Network-connectable LED Lighting Lamp Based on Android

Liu Rengui, Fan Yingjie, Dang Jianlin, Su Zhihua

(Ming De College of NWPU, Xi'an 710124, China)

Abstract: Smart home refers to the connection of various devices through the Internet of Things technology. The rapid development of the Internet of Things technology has promoted the development of smart homes. An Android-based networkable LED lighting lamp is designed. The LED lighting lamp uses Android-based intelligent electronic devices as the control terminal to achieve communication with the LED lighting lamp through the socket communication method to achieve the lighting The purpose of the switch and brightness adjustment. The LED lighting uses the STM32F030F4P6 microprocessor as the control chip, the high-efficiency, high-performance TPS92515 is used as the driver for the high-power LED, and USR-WIFI232-B2 is used as the WIFI communication module. The brightness of the LED lighting lamp is adjusted by the PWM signal output by the microprocessor, which can not only achieve linear adjustment of the brightness from 0% to 100%, but also solves the problem that the ordinary lighting lamp is turned on and off instantly when the lamp is turned on and off. , Impact on human eyes. The lighting lamp designed by this method has strong practicability and has good promotion value.

Keywords: smart home; Android; Socket; LED Lighting; WIFI

0 引言

智能家居是指通过物联网技术将家庭中的音视频设备、照明系统、数字影院系统、网络家电等各种设备通过无线的方式连接到一起, 提供家用电器的远程智能控制。与普通家居相比, 智能家居不仅具有传统的居住功能, 还兼备网络通信、信息家电、设备自动化, 提供全方位的信息交互功能。

当前, 高效、节能、环保和使用寿命长的 LED 照明灯已大范围替代效率低、对环境污染大以及使用寿命短的白炽灯和荧光灯。LED 照明灯主要采用大功率 LED 灯珠作为发光光源, 大功率 LED 灯珠属于电流型半导体器件, 具有二极管的特性, 通过调节其流经的电流即可调节其发光亮度, 因此, 通过微控制器和带亮度调节功能的专用的 LED 驱动芯片, 即可方便的调节灯光的亮度。另外, 基于 Android 系统的智能电子设备, 如智能手机、PAD 等随处可

见, 成为人们生活中必不可少的一部分。因此, 将基于 Android 系统的智能电子设备和带无线通信功能的可调光 LED 照明灯结合, 将会打开产生一个新的应用空间。

本文阐述了一种带微处理器和 WIFI 通信功能的可调光、可组网的 LED 照明灯, 并提出了将多个该 LED 照明灯组成一个分布式的可调光的智能照明控制系统的方案。

1 系统结构及原理

本系统由 Android 终端以及分布在家居空间的 LED 照明灯组成, 如图 1 所示。

图 1 中, AP 为无线接入点, 是网络的中心节点, 在智能家居中, 可以采用家庭或办公室常用的无线路由器^[1]作为 AP。Android 终端和其他 LED 照明灯为 STA 站点, 所有的 STA 都以 AP 作为无线网络的中心, STA 之间的相互通信都通过 AP 转发完成。AP 通过 WLAN 与外网连接, 外网的服务终端可以通过 AP 访问到内部各个 STA 站点。

Android 终端作为 TCP Server, 其余的 LED 照明灯作为 TCP Client^[2], 作为 TCP Client 的 LED 照明灯上电后, 其内部的 WIFI 模块会主动连接指定 IP 地址和端口的 Android 终端, 连接成功后, Android 终端和 LED 照明灯之间就可以进行通信了。LED 照明灯内部的处理器就会定时发

收稿日期: 2020-02-17; 修回日期: 2020-03-16。

基金项目: 陕西省教育厅科研计划项目资助(18JK1170)。

作者简介: 刘仁贵(1983-), 男, 湖南湘乡人, 实验师, 主要从事嵌入式系统设计技术方向的研究。

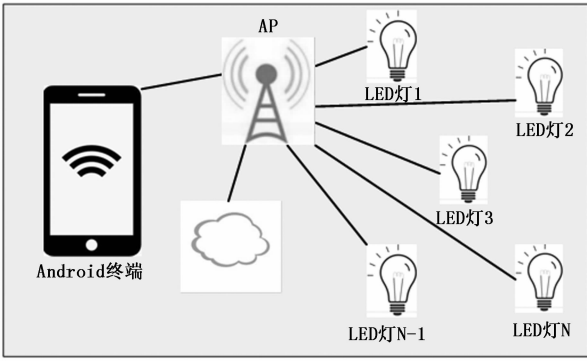


图 1 系统结构方案

送 LED 照明灯的状态信息^[3]，如开、关状态以及亮度比例等。安装在 Android 终端的 APP^[4]运行后，会通过 WIFI 网络接收到各个 LED 灯上传的数据，并显示各个照明灯的状态。同时用户通过 APP 的操作，通过 socket 通信的方式，按照自定的通信协议来控制各个灯的开、关以及调节各个照明灯的亮度。

LED 照明灯是照明系统中的功能个体，主要由 WIFI 通信模块、MCU 和 LED 驱动电路组成，如图 2 所示。

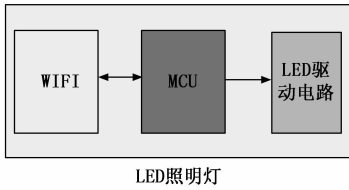


图 2 LED 照明灯原理

在 LED 照明灯的电路组成中，WIFI 电路构成 MCU 与 Android 终端的通信桥梁，MCU 是单个 LED 照明的控制中心，负责解析并执行 Android 终端发出的控制指令，MCU 选用带有 PWM 控制器的微处理器，这样很方便与带 PWM 调光功能接口的 LED 驱动芯片无缝对接。

2 硬件电路设计

2.1 LED 驱动电路

TPS92515 系列器件是内部集成了 290 mΩ 低电阻 N 沟道金属氧化物半导体场效应晶体管 (MOSFET) 的紧凑型单片开关稳压器，该系列开关稳压器利用恒定关断时间和峰值电流控制功能来运行，工作原理简单可靠。该系列器件输出高达 2A 的恒定平均电流，具有多种调光方法，适用于注重高效率、高带宽、PWM 模拟调光以及小尺寸的高亮度 LED 照明应用场景，例如汽车照明、工业照明、农业、航海和重工业照明等领域。TPS92515 系列器件具有很宽的输入电压范围，其中 TPS92515x 的输入电压范围为 5.5 V 至 42 V，TPS92515HVx 的输入电压范围为 5.5 V 至 65 V^[5]。

在 LED 照明灯中，采用 TPS92515 作为大功率 LED 的驱动电路，具体设计原理图如图 3 所示。

图 3 中，EN 信号用于控制 P 型 MOS 管 U1 (IRF6216) 的导通与截止，从而控制 TPS92515 的电源输入，达到通过 EN 信号控制 LED 照明灯的开和关的目的。TPS92515 的第 9 引脚为 PWM 调光信号输入引脚，标准的 PWM 信号的频率范围为 100 Hz 至 2 kHz，该引脚与微处理器可输出 PWM 信号^[6]的特殊引脚连接，通过固件程序改变输出 PWM 信号的占空比，来调节 LED 照明灯的亮度。

2.2 WIFI 电路

WIFI 电路作为 LED 照明灯与 Android 终端通信的桥梁，WIFI 电路采用济南有人物联网技术有限公司的 TTL 转 WIFI 模块 USR-WIFI232-B2，USR-WIFI232-B2 模组是一款一体化的 802.11 b/g/n 模组，通过该模组，传统的串口设备或 MCU 控制的设备可以很方便的接入 WIFI 无线网络，从而实现物联网络控制与管理。由于模块内部完成协议转换，因此，对于用户来说，无需关心内部具体细节，通过简单设置即可实现串口与 WIFI 之间数据的双向透传，其功能结构图如图 4 所示。

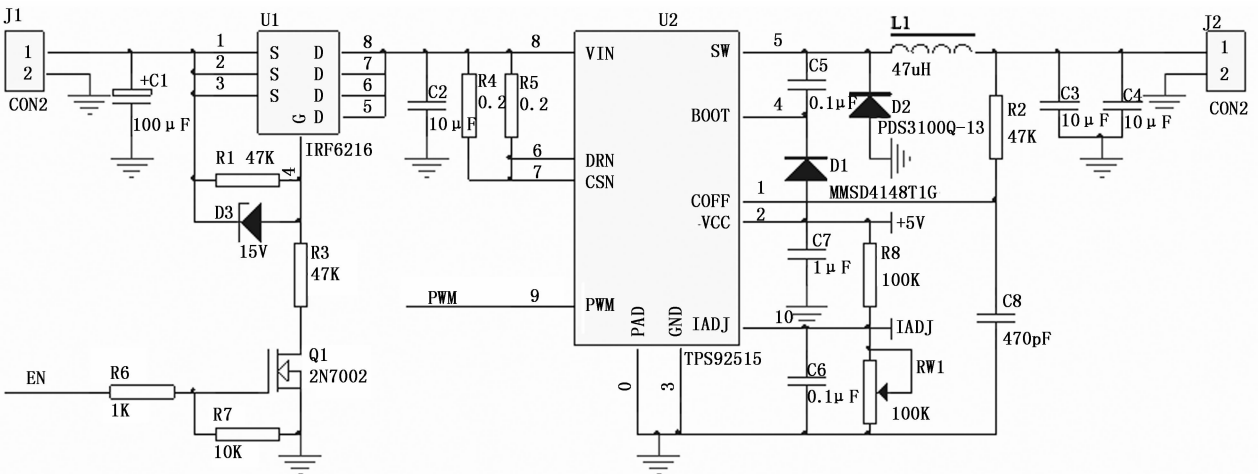


图 3 LED 驱动电路

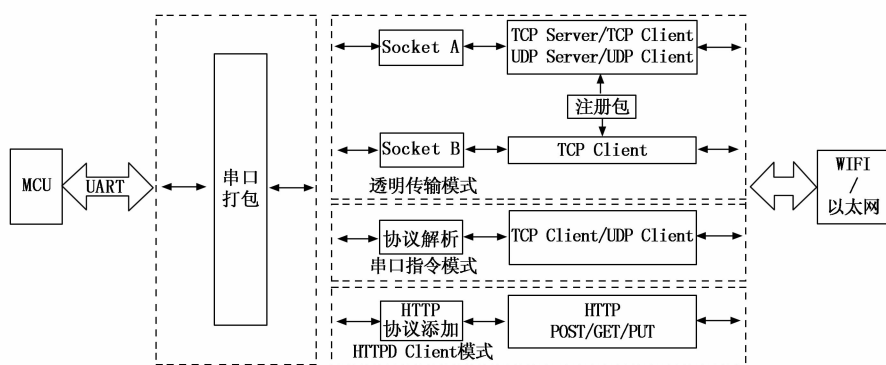


图 4 WiFi 电路功能结构图

USR-WIFI232-B2 有 5 中数据传输模式, 分别为透明传输模式、串口指令模式、GPIO 模式、Httpd client 模式和 AT 命令模式。本设计中, USR-WIFI232-B2 采用透明传输模式, 在此模式下, 所有需要收发的数据在串口与 WIFI 之间不做任何解析, 降低用户使用的复杂度。透传传输模式可以完全兼容用户原有的软件平台, 用户设备基本不用做软件改动就可以实现此功能^[7]。USR-WIFI232-B2 的电路接口如图 5 所示。

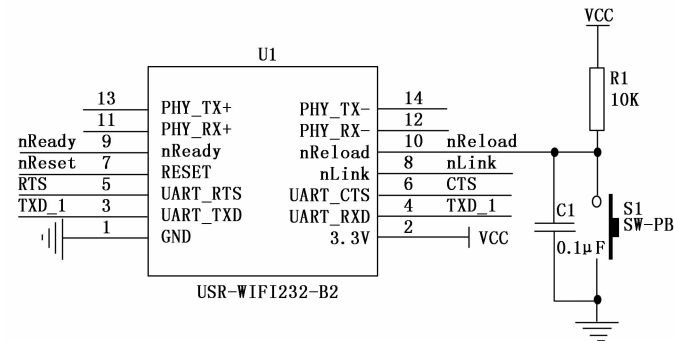


图 5 WiFi 接口电路

2.3 微处理器

智能家居调光系统的 LED 照明灯的微处理器采用 STM32F030F4P6^[8], STM32F030F4P6 是 STM32F 系列微控制器中, 成本很低的一种 32 位 ARM 处理器, 主频 48 MHz, 该系列芯片是意法半导体 (ST) 公司出品, 其内核是 Cortex-M0, 传承了 STM32 系列的重要特性, 特别适合成本敏感型应用, 该芯片集成定时器 Timer, UART 等多种外设功能, 这些外设功能的存在, 非常有利于简化电路的设计工作和降低产品的成本。

STM32F030F4P6 处理器内部有 1 个高级定时器 TIM1, 通用定时器 TIM3、TIM14、TIM15、TIM16、TIM17 和基本定时器 TIM6、TIM7, 一共 8 个定时器, 除了 TIM6 和 TIM7 以外, 其他的定时器都可以用来配置产生 PWM 信号, 其中高级定时器 TIM1 可以同时产生 6 通道的 PWM 输出, 而通用时器也能同时产生多达 4 路的 PWM 输出。PWM (脉冲宽度调制) 是利用微处理器的数字输出来对模

拟电路进行控制的一种非常有效的技术。在本设计中, 我们将 STM32F030F4P6 的第 18 引脚 PA10 与 LED 驱动电路的 PWM 控制引脚相连接, 通过底层软件配置, 将 TIM1 的 CH3 产生一路 1 kHz 的 PWM 输出, 用于调节 LED 的亮度。STM32F030F4P6 的第 17 引脚 PA9 与 LED 驱动电路的 EN 信号相连接, 用于控制驱动电路的供电。STM32F030F4P6 的第 8 引脚 PA2 (USART1_TX) 和第 9 引脚 PA3 (USART1_RX) 分别与 USR-WIFI232-B2 的第 4 和第 3 引脚连接, 用于与 WIFI 模块进行串行通信。

微处理器电路如图 6 所示。

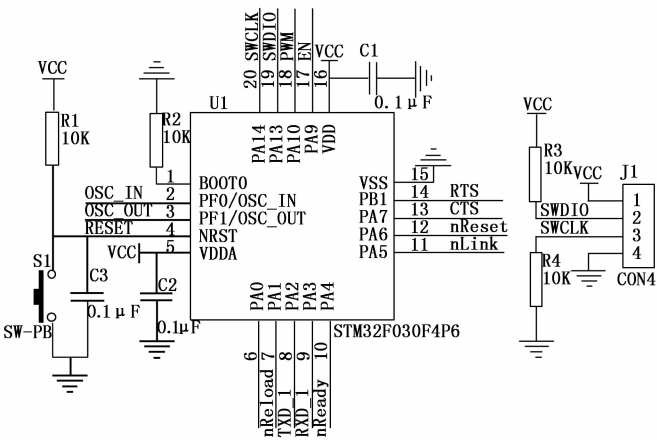


图 6 微处理器电路

3 系统软件设计

3.1 Android 软件设计

ToggleButton^[9]是 Android 中一个具有选中 and 未选中两种状态的按钮, 并可为不同的状态设置不同的显示文本, 常用于表示开、关场景中。在 Android 终端的 APP 软件设计中, LED 照明灯的开和关用 ToggleButton 来实现。在 xml 布局文件中设置 ToggleButton, 然后在 Activity 文件中初始化用于控制照明灯开关的 ToggleBuuton 控件, 并设置控件的 OnCheckedChangeListener 监听器和重写监听器的 onCheckedChanged () 方法, 通过该方法的传入参数 isChecked, 判断控件的状态。如果控件压下, 则发送开灯指令, 如果控件弹起, 则发送关灯指令。具体实现代码如下。

```
LED_Lighting.setOnCheckedChangeListener(new OnCheckedChangeListener()
{
@Override
public void onCheckedChanged(CompoundButton buttonView, boolean isChecked)
{
if(! buttonView.isPressed())
```

```

{
return;
}
else
{
if(isChecked)
{
//发送开灯指令
printWriter.print(" $ $ o# #");
    printWriter.flush();
}
else
{
printWriter.print(" $ $ p# #");
    printWriter.flush();
}
}
}
});

```

在 Android 系统中,SeekBar 继承于 ProgressBar,是增加了滑动块的扩展 ProgressBar。SeekBar 的目的是让用户可以自主拖动,用来调节进度,比如通过拖动进度条来改变音乐和视频的播放进度,这里我们也用 SeekBar 来调节 LED 照明灯的亮度。其具体实现代码如下。

```

private SeekBar mseekBarBrightness;
mseekBarBrightness.setOnSeekBarChangeListener ( new On-
SeekBarChangeListener()
{
@Override
public void onStopTrackingTouch(SeekBar seekBar)
{
//移动后放开事件
}
@Override
public void onStartTrackingTouch(SeekBar seekBar)
{
// TODO Auto-generated method stub
}
@Override
public void onProgressChanged(SeekBar seekBar, int progress,
boolean fromUser)
{
// TODO Auto-generated method stub
// 取得当前亮度
int Brightness= seekBar.getProgress();
// 当进度小于 5 时,设置成 0,关闭 LED 照明灯
if (Brightness< 5)
{
Brightness= 0;
}
// 根据当前进度改变亮度 String orderStr = "";

```

```

orderStr = String.valueOf(Brightness);
    try
    {
printWriter.print(orderStr);
        printWriter.flush();
    }
catch(Exception ex)
    {
ex.printStackTrace();
}
}
});

```

Android 终端的 APP 中,通过 ToggleButton 来打开和关闭 LED 照明灯,通过 SeekBar 来调节 LED 照明灯的亮度是外在的,处于应用层,而内在的通信,即 Android 终端与 LED 照明灯之间通信指令的传输则是通过 Socket 来实现的。Socket 即为套接字,用于描述 IP 地址和端口,是支持 TCP/IP 协议的网络通信的基本操作单元,也是应用层与 TCP/IP 协议族通信的中间软件抽象层,表现为一个封装了 TCP/IP 协议族的编程接口 (API),主要解决数据如何在网络中传输。对我们用户来说,在 Andorid 平台上进行 TCP/IP 协议开发,只需调用 Socket 按照指定的协议,向网络发出请求或者应答网络请求,即可实现通信,内部具体实现则无需关心。

在本设计中,采用基于 TCP 的 Socket 实现客户端与服务端间的双向可靠连接的实时通信,是一种 C/S 模型的通信方式。Android 终端为服务端,LED 照明灯为客户端,由于 LED 照明灯采用的是具有透明传输功能的 WIFI 模块,内部集成了 Socket 通信功能,在使用时,只要对模块进行配置成客户端,并绑定服务器的 IP 地址和端口号即可,而服务器端的程序则要另行开发。基于 TCP 的 Socket 通信模型如图 7 所示。

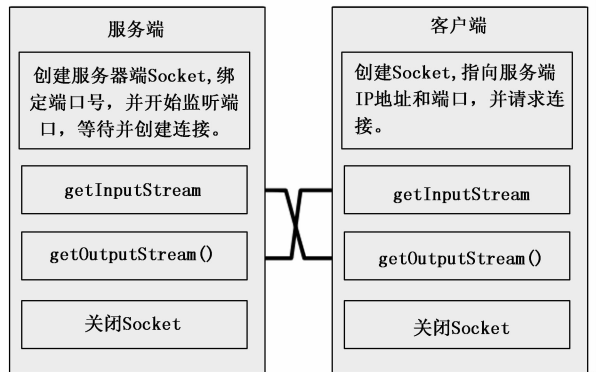


图 7 Socket 通信模型

图 7 中,服务器与客户端通信前,需要创建服务器端的 Socket,绑定端口号并开始监听端口,等待建立基于 TCP 面向连接的端对端的传输。具体实现代码如下。

```
public class LightServer
```

```

{
    public static ArrayList<Socket> socketList = new ArrayList
    <Socket>();
    public static void main(String[] args)
    throws IOException
    {
        //创建 8899 端口的服务端
        ServerSocket socket = new ServerSocket(8899);
        while(true)
        {
            //等待连接
            Socket new_socket = socket.accept();
            socketList.add(new_socket);
            //为新加入的连接创建线程
            new Thread(new ServerThread(new_socket)).start();
        }
    }
}

```

服务器与客户端一旦创建连接后, 就可以通过 `getInputStream` 和 `getOutputStream` 来接收和发送数据。`getInputStream` 用于获得读 `Socket` 的输入流, `getOutputStream` 用于获得写 `Socket` 的输出流。通过 `getInputStream` 和 `getOutputStream`, 实现 LED 照明灯亮度调节指令的发送和 LED 照明灯状态的收集, 特别是 APP 软件中途退出重新启动时, APP 根据收集 LED 照明灯状态调整显示信息, 确保 APP 显示的状态与实际的状态一致。

3.2 固件程序设计

固件程序运行于 STM32F030F4P6, 主要有两个功能, 一是用于从串口 1 接收 Android 终端通过 WIFI 模块发送的控制指令, 打开和关闭 LED 照明灯和调节亮度, 二是将照明灯的状态信息, 例如当时的实际亮度比和开、关状态等信息, 通过 WIFI 模块定时上传到 Android 终端, 使 Android 终端的显示状态与实际的一致。固件程序的流程图如图 8 所示。

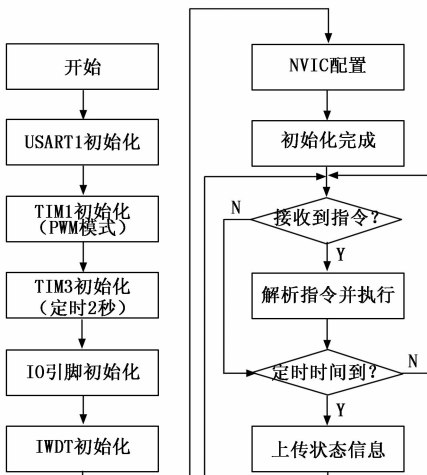


图 8 固件程序流程图

固件程序中, STM32F030F4P6 的 USART1 的收发数据均采用 DMA 的方式来完成, 其中数据发送采用 DMA1 的通道 4, 数据接收采用 DMA1 的通道 5, 并通过空闲中断的方式来响应接收到的指令, 这样可以大大提高 CPU 对外设的响应速度。通用定时器 TIM3 用来设置上传信息的 2 秒时间间隔, 每隔 2 秒, LED 照明灯上传当前的状态信息, APP 如果超时未收到照明灯发送的信息, 则认为 LED 照明灯故障或者失联, 并给出提示信息。用于调光的 PWM 信号是通过配置 STM32F030F4P6 的 TIM1 来产生 1 kHz 的 PWM 信号, 并通过库函数 `TIM_SetCompare3 (TIM1, Duty)` 或者通过修改寄存器的值 `TIM1->CCR3=Duty` 来修改 PWM 信号的占空比, 达到调节亮度的目的。Duty 为占空比, 取值范围为 0~100。

为了解决普通照明灯开灯和关灯时, 照明灯瞬间点亮和瞬间熄灭, 对人眼的冲击, 固件程序在接收到开灯指令时, 并不会将亮度直接调节到指定值, 而是先将 LED 驱动电路的电源关闭, 将输出的 PWM 信号的占空比调节到最小, 然后打开 LED 驱动电路的电源, 再将输出 PWM 信号的占空比逐步增大, 使亮度慢慢增加。在接收到关灯的指令时, 先将 PWM 信号的占空比逐步减小, 使亮度慢慢降低, 达到最小值后再关闭 LED 驱动电路的电源。

4 实验结果与分析

根据设计, 制作了原理样机, 并在功能和指标上做了测试。

在功能上, 各个 LED 照明灯上电后, 默认均处于关灯状态, 并每隔 2 秒发送状态信息, 用于连接 Android 终端。打开运行在 Android 手机上的 APP 后, LED 照明灯与 Android 终端能很快建立连接。建立连接前, APP 上 LED 照明灯的状态是灰色, 建立连接后, 状态变为绿色, 表示连接成功。连接成功后, 通过 APP 上的控件可以对 LED 照明灯进行开灯和关灯操作, 通过拖动进度条可以很灵敏地调节对应照明灯的亮度, 并且在开、关灯的时候, 亮度有明显的变化, 对人眼没有明显的刺激。另外, 对样机进行了高温环境下的老化测试, 样机能连续较长时间稳定工作。

在指标上, 主要测试了实际输出电流与 PWM 信号占空比的关系。通过对 PWM 信号的占空比与输出电流的监测, 得出 LED 驱动电路的输出电流与 PWM 信号占空比的关系如图 9 所示。

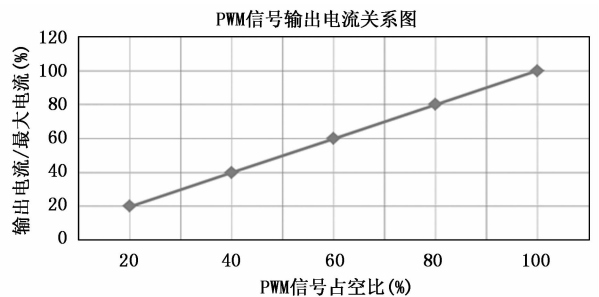


图 9 电流与占空比关系图