

实验步骤 1 内容简介

阅读实验原理，了解 zlg7290 的读写流程和 I2C 总线的使用方法
根据实验步骤完成对 zlg7290 的读写程序设计和验证

实验步骤 2 实验目的

了解 zlg7290 的控制流程
掌握使用 I2C 总线读写 zlg7290 的静态驱动程序设计方法

实验环境

硬件：装有 Linux 操作系统的开发板
软件：Ubuntu12.0, IDE, putty
* 经由北京航空航天大学计算机学院的自主开发所提供的 WEB 服务，学员现在可通过网上实验室提供的远程终端在网页浏览器中完成这些需要特殊软硬件环境的开发实践了。

实验步骤 3 ZLG7290 介绍

ZLG7290 是广州周立功单片机发展有限公司自行设计的数码管显示驱动及键盘扫描管理芯片。能够直接驱动 8 位共阴极数码管（或 64 只独立的 LED），同时还可以扫描管理多大 64 只按键。其中有 8 只按键可以作为功能键使用，就像电脑键盘上的 Ctrl, Shift、Alt 键一样。另外 ZLG7290 内部还设有连击计数器，能够使某键按下后不松手而连续有效。该芯片为工业级芯片，抗干扰能力强，在工业测控中已有大量应用。该器件通过 I2C 总线接口进行操作，ZLG7290 引脚图如图 1。

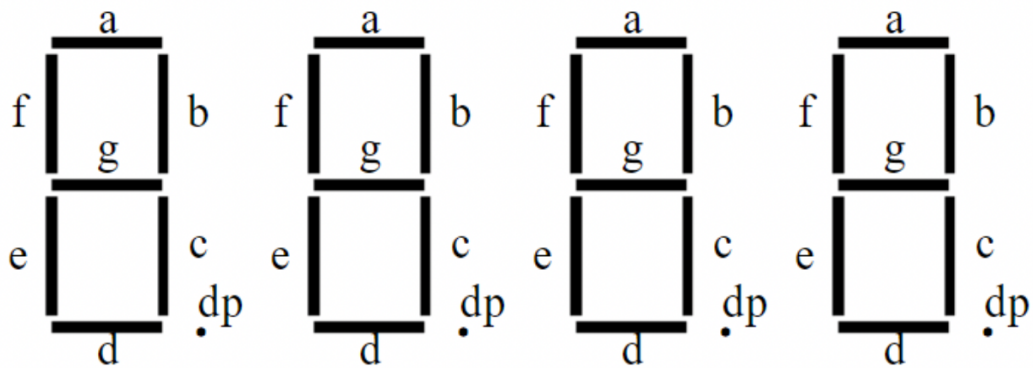
1	SC/KR2	KR1/SB	24
2	SD/KR3	KR0/SA	23
3	DIG3/KC3	KC4/DIG4	22
4	DIG2/KC2	KC5/DIG5	21
5	DIG1/KC1	SDA	20
6	DIG0/KC0	SCL	19
7	SE/KR4	OSC2	18
8	SF/KR5	OSC1	17
9	SG/KR6	VCC	16
10	DP/KR7	RST	15
11	GND	INT	14
12	DIG6/KC6	KC7/DIG7	13

下表说明了 ZLG7290 各引脚的功能

引脚序号	引脚名	功能
1	SC/KR2	数码管 c 段 / 键盘行信号 2
2	SD/KR3	数码管 d 段 / 键盘行信号 3
3	DIG3/KC3	数码管位选信号 3 / 键盘列信号 3
4	DIG2/KC2	数码管位选信号 2 / 键盘列信号 2
5	DIG1/KC1	数码管位选信号 1 / 键盘列信号 1
6	DIG0/KC0	数码管位选信号 0 / 键盘列信号 0
7	SE/KR4	数码管 e 段 / 键盘行信号 4
8	SF/KR5	数码管 f 段 / 键盘行信号 5
9	SG/KR6	数码管 g 段 / 键盘行信号 6
10	DP/KR7	数码管 dp 段 / 键盘行信号 7
11	GND	接地
12	DIG6/KC6	数码管位选信号 6 / 键盘列信号 6
13	DIG7/KC7	数码管位选信号 7 / 键盘列信号 7
14	INT	键盘中断请求信号，低电平（下降沿）有效
15	RST	复位信号，低电平有效
16	Vcc	电源，+3.3~5.5V
17	OSC1	晶振输入信号
18	OSC2	晶振输出信号
19	SCL	I2C 总线时钟信号
20	SDA	I2C 总线数据信号
21	DIG5/KC5	数码管位选信号 5 / 键盘列信号 5
22	DIG4/KC4	数码管位选信号 4 / 键盘列信号 4
23	SA/KR0	数码管 a 段 / 键盘行信号 0
24	SB/KR1	数码管 b 段 / 键盘行信号 1

数码管驱动 zlg7290.c 实现了设备文件操作控制，用户态可以调用 zlg7290_hw_write(), zlg7290_hw_read()和 zlg_led_ioctl()函数来对数码管进行读写操作。

实验步骤 4 数码管显示原理介绍



0xfc, 0x0c, 0xda, 0xf2, 0x66, 0xb6, 0xbe, 0xe0----对应显示 0-7
0xfe, 0xf6, 0xee, 0x3e, 0x9c, 0x7a, 0x9e, 0x8e----对应显示 8-F

数码管显示对应二进制位（示例）

十六进制	二进制	十进制	a	b	c	d	e	f	g	dp
0xfc	11111100	0	1	1	1	1	1	1	0	0
0x0c	00001100	1	0	0	0	0	1	1	0	0
0xda	...									
0xf2	...									

实验步骤 5 嵌入式开发 Web 在线仿真实验平台

1. 本次实验依然在实验室上课

登录嵌入式开发 Web 在线仿真实验平台地址 <http://219.224.160.133:8090>
学员登录：用户名和密码为各位学号（字母大写）

2. 正确填写模板后，【提交编译】

在线仿真实验平台

我的任务

进行实验

修改密码

关于

联络

你好, 杜东峰!

退出

编辑调试

实验说明

板卡效果

重新加载模板代码

实验服务正常。编辑你的代码。

提交编译

上传程序

```
136 }
137
138 //=====//
139 //此行为文件分隔符
140
141 /*
142  * MYZR Technology Co., Ltd
143  * http://www.myzr.com.cn
144  * Tang Bin <tangb@myzr.com.cn>
145  */
146
147 #include <linux/kernel.h>
148 #include <linux/module.h>
149 #include <linux/i2c.h>
150 #include <linux/input.h>
151 #include <linux/delay.h>
152 #include <linux/slab.h>
153 #include <linux/interrupt.h>
154 #include <linux/irq.h>
155 #include <linux/gpio.h>
156 #include <linux/fs.h>
157 #include <linux/cdev.h>
158 #include <linux/platform_device.h>
159
160 #include <linux/module.h>
161 #include <linux/cdev.h>
162 #include <linux/fs.h>
163 #include <linux/poll.h>
164 #include <linux/sched.h>
165
166 #define ZLG7290_NAME      "zlg7290"
167 #define ZLG7290_LED_NAME  "zlg7290_led"
168
169 #define REG_SYSTEM        0x00
```

```
make[1]: Entering directory '/home/fzk/onlinePlatformDir/
/home/fzk/my-imx6/03_tools/gcc-linaro-5.4.1-2017.05-i686_
/home/fzk/my-imx6/03_tools/gcc-linaro-5.4.1-2017.05-i686_
make[1]: Leaving directory '/home/fzk/onlinePlatformDir/w
make -C ../src
make[1]: Entering directory '/home/fzk/onlinePlatformDir/
make -C /home/fzk/linux-3.14.52 M=/home/fzk/onlinePlatform
make[2]: Entering directory '/home/fzk/linux-3.14.52'
CC [M] /home/fzk/onlinePlatformDir/workdir/5/3/kernel/
scripts/Makefile.build:314: recipe for target '/home/fzk/
Makefile:1278: recipe for target '_module_/home/fzk/online
make[2]: Leaving directory '/home/fzk/linux-3.14.52'
Makefile:22: recipe for target 'default' failed
make[1]: Leaving directory '/home/fzk/onlinePlatformDir/w
Makefile:4: recipe for target 'default' failed
```

未出现 failed、error 等字样说明编译成功

3. 【上传程序】

在线仿真实验平台

我的任务

进行实验

修改密码

关于

联络

你好, 杜东峰!

退出

编辑调试

实验说明

板卡效果

重新加载模板代码

实验服务正常。编辑你的代码。

提交编译

上传程序

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <fcntl.h>
5 #include <sys/stat.h>
6 #include <sys/types.h>
7 #include <sys/ioctl.h>
8 #include <time.h>
9 #include <string.h>
10
11 #define WRITE_DFRAM_IO('Z', 0)
12 #define zlglog_path  "/mnt/zlg.txt"
13 #define ledlog_path  "/led.txt"
14 //数码管计数，写入日志
15 //加入流水灯，写入日志
16 unsigned char hex_map[16] = {
17     0xfc, 0x0c, 0xda, 0xf2, 0x66, 0xb6, 0xb6, 0xe0,
18     0xfe, 0xf6, 0xee, 0x3e, 0x9c, 0x7a, 0x9e, 0x8e
19 };
20
21 unsigned char bit_map[8] = {
22     0x80, 0x40, 0x20, 0x10, 0x08, 0x04, 0x02, 0x01
23 };
24
25 int number_tester(int num)
26 {
27     if (num < 0) return 0;
28     if (num > 15) return 0;
29     return num;
30 }
31 int write_zlg7290_log(unsigned char *buf)
32 {
33     int fd;
34     fd = open(zlglog_path, O_RDWR | O_CREAT);
35     if (fd == -1)
```

```
开始将程序上传到板卡...
上传完成了 10%.....
上传完成了 20%.....
上传完成了 30%.....
上传完成了 40%.....
上传完成了 50%.....
上传完成了 60%.....
上传完成了 70%.....
上传完成了 80%.....
上传完成了 80%.....
上传完成了 100%.....
已完成，请切换到板卡页面可以运行了。
```

4. 切换到“板卡效果“页面，【运行程序】



说明：本次在线实验需要由助教查看效果

扩展实验：

- 编一个时钟程序，获取当前的年（2019），月日（1218），时分（0855），并在数码管中依次显示出来，例如：2019 1218 0855 表示 2019 年 12 月 18 日 08 时 55 分
- 数码管依次显示 "2.0.1.9."、"1.2.1.8."、"0.8.5.5."

数码管驱动 `zlg7290.c` 实现了设备文件操作控制，用户态可以调用 `zlg7290_hw_write()`，`zlg7290_hw_read()` 和 `zlg_led_ioctl()` 函数来对数码管进行读写操作，阅读 ZLG7290 数码管驱动的代码程序清单，了解其实现的具体方法。