

# 浅度课设·裸C·群组点名源代码模版使用说明



主要涉及的源文件路径为“XX#NK2023CurriculumDesign-C-Group-Order-template/source/tester-key-with-led.c”

## 一、发问与展示

充分考虑到代码模板阅读过程中可能存在的问题，以及小组代码编程合作中对出现的问题能够进行及时有效的沟通，本着“自我→小组→结伴组→助教”的自治管理问题解决原则，代码前半部分中通过三栏化的模式进行问题的精准定位和有效聚焦。具体如下：

“三栏”主要分为以下三部分：

1. 编程主体部分：由所有编程部分的代码及注释构成，**需要填充的代码也在这一部分。**

```
/******编程主体部分******/
#include <main.h>
#include <s5p6818/reg-sys.h>
#include <s5p6818/reg-id.h>

static void gpio30_interrupt_func(void *);
static void gpio31_interrupt_func(void *);
/******hardware control card start******/
//decription: control leds, init and so on
enum key_status{STAGE_CHANGE, ORDER} key_st;

void led_all_off()
{
    led_set_status(LED_NAME_LED1, LED_STATUS_OFF);
    led_set_status(LED_NAME_LED2, LED_STATUS_OFF);
    led_set_status(LED_NAME_LED3, LED_STATUS_OFF);
    led_set_status(LED_NAME_LED4, LED_STATUS_OFF);
}

void led_all_on()
{
    led_set_status(LED_NAME_LED1, LED_STATUS_ON);
    led_set_status(LED_NAME_LED2, LED_STATUS_ON);
    led_set_status(LED_NAME_LED3, LED_STATUS_ON);
    led_set_status(LED_NAME_LED4, LED_STATUS_ON);
}

void init_stat()
{
    key_st = STAGE_CHANGE;
}
```

// 定义两种按键功能：“阶段切换”和“点名”

// LED灯全关  
// led\_set\_status用来设置LED灯的状态，第一  
// 将第1个LED灯熄灭(OFF)  
// 将第2个LED灯熄灭(OFF)  
// 将第3个LED灯熄灭(OFF)  
// 将第4个LED灯熄灭(OFF)

// LED灯全开  
// led\_set\_status用来设置LED灯的状态，第一  
// 将第1个LED灯打开(ON)  
// 将第2个LED灯打开(ON)  
// 将第3个LED灯打开(ON)  
// 将第4个LED灯打开(ON)


// 初始化系统状态

// 按键功能初始化为“阶段切换”  
// 群组名为灯全关标志

2. 认知发问部分：**在理解和实现编程主体部分过程中如有提出问题，则在此部分按照规范格式进行认知发问。**具体格式如下：

由有序的四元组构成，四元组包括：

- 【问题定位】：在编程主体部分中的问题定位标记
- 【问题描述】：对该定位标记的问题展示进行描述
- 【解决状态】：经过个人、小组、组间等途径解决问题后，问题的解决状态：已解决/未解决
- 【解决策略】：对该问题的解决策略（若未解决则填无）

 注意：为了保证四元组的层次性和可读性，不同的四元组之间应该用一个空行进行分割。

下面通过示例进行说明：

如对第621-623行的注释和代码部分有疑问，应按照问题定位标记的顺序添加新的问题定位标记（Q1、Q2、Q3……），示例以Q0标记，标记后创建四元组并添加具体的问题描述阐释问题。

C++

```
1 //通过按键中断方式进行群组点名任务
2 //Q0
3 int tester_key_led_beep(int argc, char * argv[])
```

之后在组员、组间充分交流后完成【解决状态】和【解决策略】部分的填写，从而实现完整的四元组。另外，可以附加上个人的信息，作为个人的贡献。具体示例如下：

```
/******认知发问部分*****
*
*【问题定位（示例）】Q0(NK2023-01#-Le)
*【问题描述（示例）】tester_key_led_beep函数在哪里被调用的？
*【解决状态（示例）】已解决(NK2023-01#-R)
*【解决策略（示例）】在main.c的main函数中被调用
*
*【问题定位】Q1(..)
*【问题表述】
*【解决状态】
*【解决策略】
*
*【问题定位】Q2(...)
*【问题表述】
*【解决状态】
*【解决策略】
*
*****认知发问部分*****/
```

3. 总结展示部分：**在此部分对理解和实现编程主体部分过程中的收获与体会，按照规范格式进行总结展示。**具体格式如下：

```
/******总结展示部分*****
*【总结展示0】NK2023-01#-Lo
* 通过裸C的代码实现，熟悉了实验箱的硬件环境，掌握了以中断的方式通过按键的方式实现对LED灯、蜂鸣器的控制.....
*
*【总结展示1】
*
*
*【总结展示2】
```

```

*
*
*****总结展示部分*****/

```

## 二、发问与展示的要求

在浅度课设·裸C·群组点名源代码中，每个小组在“认知发问部分”和“总结展示部分”两个部分中，应分别至少提出一个问题或展示一条总结。

## 三、浅度课设·裸C·群组点名源代码模版本组化填充与修改

根据修改后的《NK2023浅度课设-裸C群组点名三栏需求C-Group-Order》文档和验证单，小组组内通过角色分工，按照代码模板中注释要求填充修改相应位置及注释，组内每个角色应使用真实信息独立完善自己角色所对应的点名函数及相关注释，如需求员应负责完善修改下图中的show\_first\_name\_req()函数及相关注释（用真实信息替换“XX”，并按照注释要求填充相关区域，实现根据点名需求文档形成的验证单上的程序流程）

```
// author: XXX
// date: 2023/xx/xx
// 显示需求员姓氏拼音首字母小写的ASCII码。需求员姓氏“XXX”的拼音首字母小写“X”所对应ASCII码为XXXX XXXXB
void show_first_name_req()
{
    /*****请在下方区域编写代码及注释，实现利用LED灯显示需求员姓氏拼音首字母小写的ASCII码高4位的功能*****/

    /*****请在上方区域编写代码及注释，实现利用LED灯显示需求员姓氏拼音首字母小写的ASCII码高4位的功能*****/

    mdelay(3000);
    led_all_off();
    mdelay(1000); // LED灯全关
                  // 延迟一秒

    /*****请在下方区域编写代码及注释，实现利用LED灯显示需求员姓氏拼音首字母小写的ASCII码低4位的功能*****/

    /*****请在上方区域编写代码及注释，实现利用LED灯显示需求员姓氏拼音首字母小写的ASCII码低4位的功能*****/

    mdelay(3000);
    led_all_off();
    mdelay(1000); // LED灯全关
                  // 延迟一秒

    beep_set_status(BEEP_STATUS_ON);
    mdelay(200);
    beep_set_status(BEEP_STATUS_OFF);
    mdelay(100);
    beep_set_status(BEEP_STATUS_ON);
    // 蜂鸣器鸣响0.2秒、关闭0.1秒；再重复一次，即快速“滴商”
}

```

每个角色都有对应的点名函数需要填充完善或修改，其中，**组长要负责显示组号和显示自己姓氏代码的修改，其他组内成员参照组长的点名函数对自己的点名函数进行填充**，组长要修改的代码如下：

```
// author: XXX
// date: 2023/xx/xx
// 显示组号的二进制转换格式，组号“XX”转换成8位二进制数为XXXX XXXXB，模板以数字“0”为例，“0”转换成8位二进制数为0000 0000B，请按照本组真实组号对本函数进行修改
void show_group_number()
{
    /*****请在下方区域修改代码及注释，实现利用LED灯显示本组组号二进制格式高4位的功能*****/
    led_set_status(LED_NAME_LED1, LED_STATUS_OFF); // 0
    led_set_status(LED_NAME_LED2, LED_STATUS_OFF); // 0
}

```

```

led_set_status(LED_NAME_LED2, LED_STATUS_ON);
led_set_status(LED_NAME_LED3, LED_STATUS_ON);
led_set_status(LED_NAME_LED4, LED_STATUS_ON);
/*****请在上方区域修改代码及注释，实现利用LED灯显示本组组号二进制形式高4位的功能*****/

mdelay(3000);
led_all_off();
mdelay(1000);

/*****请在下方区域修改代码及注释，实现利用LED灯显示本组组号二进制形式低4位的功能*****/
led_set_status(LED_NAME_LED1, LED_STATUS_OFF);
led_set_status(LED_NAME_LED2, LED_STATUS_OFF);
led_set_status(LED_NAME_LED3, LED_STATUS_OFF);
led_set_status(LED_NAME_LED4, LED_STATUS_OFF);
/*****请在上方区域修改代码及注释，实现利用LED灯显示本组组号二进制形式低4位的功能*****/

mdelay(3000);
led_all_off();
mdelay(1000);

beep_set_status(BEEP_STATUS_ON);
mdelay(200);
beep_set_status(BEEP_STATUS_OFF);
mdelay(100);
beep_set_status(BEEP_STATUS_ON);
mdelay(200);

// author: xxx
// date: 2023/xx/xx
// 显示组长姓氏拼音首字母小写的ASCII码。组长姓氏“XXX”的拼音首字母小写“X”所对应ASCII码为XXXX XXXXB
void show_first_name_Lea()
{
/*****请在下方区域修改代码及注释，实现利用LED灯显示组长姓氏拼音首字母小写的ASCII码高4位的功能*****/
led_set_status(LED_NAME_LED1, LED_STATUS_OFF);
led_set_status(LED_NAME_LED2, LED_STATUS_ON);
led_set_status(LED_NAME_LED3, LED_STATUS_ON);
led_set_status(LED_NAME_LED4, LED_STATUS_ON);
/*****请在上方区域修改代码及注释，实现利用LED灯显示组长姓氏拼音首字母小写的ASCII码高4位的功能*****/

mdelay(3000);
led_all_off();
mdelay(1000);

/*****请在下方区域修改代码及注释，实现利用LED灯显示组长姓氏拼音首字母小写的ASCII码低4位的功能*****/
led_set_status(LED_NAME_LED1, LED_STATUS_ON);
led_set_status(LED_NAME_LED2, LED_STATUS_OFF);
led_set_status(LED_NAME_LED3, LED_STATUS_ON);
led_set_status(LED_NAME_LED4, LED_STATUS_OFF);
/*****请在上方区域修改代码及注释，实现利用LED灯显示组长姓氏拼音首字母小写的ASCII码低4位的功能*****/

mdelay(3000);
led_all_off();
mdelay(1000);

beep_set_status(BEEP_STATUS_ON);
mdelay(200);
beep_set_status(BEEP_STATUS_OFF);
mdelay(100);
beep_set_status(BEEP_STATUS_ON);
mdelay(200);
}

```

各角色应全部完成自己所负责的代码部分。

对修改后的完整代码进行编译、烧写TF卡，对照本组化后的《NK2023浅度课设-裸C群组点名三栏需求C-Group-Order》到实验箱上运行，同时测量相应阶段功耗，将功耗读数填写在验证单相应位置，没有问题后，找助教进行验收即可。