**HarmonyOS智慧设备开发-LED灯控制案例**

1. 技术相关

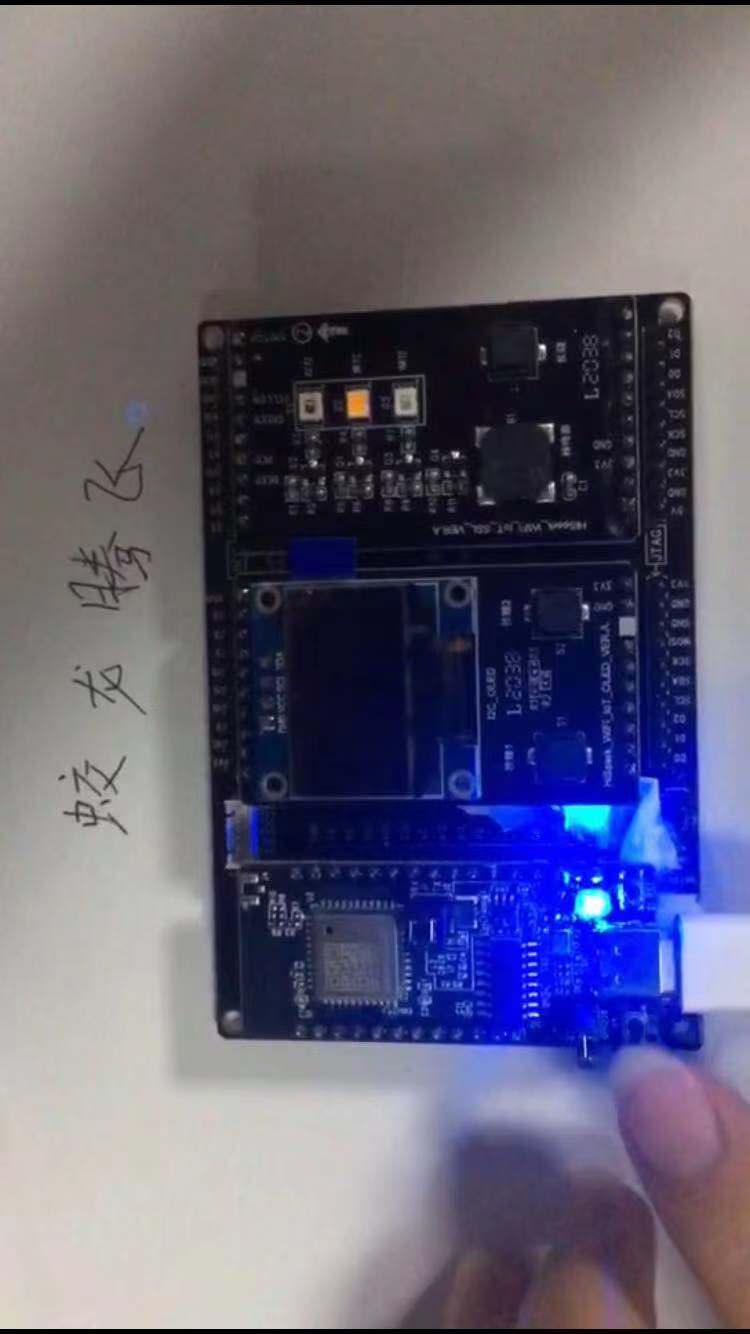
项目名称: led\_kz\_02

项目语言: C语言

开发板：Hi3861

工具：vscode

二、效果如下

三、开发过程

**1.建立项目**

在applications/sample/wifi-iot/app/下新建一个文件夹用于存放项目文件，在文件中创建一个led\_kz\_02.c文件用于存放业务代码，和一个BUILD.gn用于存放编译配置。

**2.编写代码**

控制原理：

在Hi361开发板中，GPIO09控制LED灯，GPIO05与user按钮连接，通过函数hi\_gpio\_register\_isr\_function监控user按钮中断并执行相应操作；

|  |
| --- |
| static void \*UserButton(const char \*arg){ //中断后执行的函数  USER\_KZ\_GPIO=!USER\_KZ\_GPIO;  }  static void \*kzLed(const char \*arg)  {  hi\_gpio\_register\_isr\_function(LED\_KZ\_GPIO,HI\_INT\_TYPE\_EDGE,  HI\_GPIO\_EDGE\_FALL\_LEVEL\_LOW,UserButton, NULL);  while (1)  {  IoTGpioSetOutputVal(LED\_Out\_GPIO, USER\_KZ\_GPIO);  }  return NULL;  } |

**3.运行代码**

修改applications/sample/wifi-iot/app/BUILD.gn文件,使其编译新建的led\_kz.c文件

|  |
| --- |
| import("//build/lite/config/component/lite\_component.gni")  lite\_component("app") {  features = [  "led\_kz:led\_kz"  ]  } |

然后进行编译烧录，按下res键运行项目后，按一次user，led灯会随之熄灭；再按一次便会亮起。

**4.关键代码**

led\_kz.c

|  |
| --- |
| #include <stdio.h>  #include <unistd.h>  #include "ohos\_init.h"  #include "cmsis\_os2.h"  #include "hi\_gpio.h"  #include "hi\_io.h"  #define LED\_TASK\_STACK\_SIZE 512  #define LED\_TASK\_PRIO 25  #define LED\_Out\_GPIO 9  #define LED\_KZ\_GPIO 5  int USER\_KZ\_GPIO = 0;  static void \*UserButton(const char \*arg){  USER\_KZ\_GPIO=!USER\_KZ\_GPIO;  }  static void \*kzLed(const char \*arg)  {  hi\_gpio\_register\_isr\_function(LED\_KZ\_GPIO, HI\_INT\_TYPE\_EDGE, HI\_GPIO\_EDGE\_FALL\_LEVEL\_LOW,  UserButton, NULL);  while (1)  {  IoTGpioSetOutputVal(LED\_Out\_GPIO, USER\_KZ\_GPIO);  }  return NULL;  }  static void LedExampleEntry(void)  {  osThreadAttr\_t attr;  hi\_gpio\_init();  hi\_io\_set\_func(LED\_Out\_GPIO, HI\_IO\_FUNC\_GPIO\_9\_GPIO);  hi\_gpio\_set\_dir(LED\_Out\_GPIO, HI\_GPIO\_DIR\_OUT);  hi\_io\_set\_func(LED\_KZ\_GPIO, HI\_IO\_FUNC\_GPIO\_5\_GPIO);  hi\_gpio\_set\_dir(LED\_KZ\_GPIO, HI\_GPIO\_DIR\_IN);  hi\_io\_set\_pull(LED\_KZ\_GPIO, HI\_IO\_PULL\_UP);    attr.name = "kzLed";  attr.attr\_bits = 0U;  attr.cb\_mem = NULL;  attr.cb\_size = 0U;  attr.stack\_mem = NULL;  attr.stack\_size = LED\_TASK\_STACK\_SIZE;  attr.priority = LED\_TASK\_PRIO;  if (osThreadNew((osThreadFunc\_t)kzLed, NULL, &attr) == NULL) {  printf("[LedExample] Falied to create LedTask!\n");  }  }  SYS\_RUN(LedExampleEntry); |

BUILD.gn：

|  |
| --- |
| static\_library("led\_kz") {  sources = [  "led\_kz.c"  ]  include\_dirs = [  "//utils/native/lite/include",  "//kernel/liteos\_m/components/cmsis/2.0",  "//base/iot\_hardware/peripheral/interfaces/kits",  ]  } |

完整代码地址：<https://gitee.com/jltfcloudcn/smart-devices/tree/master/led_kz_zd>