

上海电力大学

嵌入式系统设计应用



实验名称: 实验二外部中断实验

专业班级: 集成电路设计与集成系统

姓 名: 某同学

学 号: 2021xxxx

2024 年 6 月 9 日

一、实验要求

1. 运行例程**实验 4 EXTI 外部中断实验**，分别按下 key1/key2 键，观察实验现象
2. 看懂源程序
3. 修改源程序、实现按下 key1 键，红灯、黄灯，绿灯按顺序亮，按下 key2 键，绿灯。黄灯，红灯按顺序亮。红灯亮时蜂鸣器响。
4. 完成实验报告，把修改的程序截图、实验现象截图或者图片整理到报告中。

二、实验内容及结果

1. 编写代码

需要修改 stm32f4xx_it.c 中的 void KEY1_IRQHandler(void)
首先修改 main.c，初始化蜂鸣器等

```
1  /**
2      *****
3      * @file    main.c
4
5      * @brief    使用外部中断 EXTI 检测按键，控制彩灯。
6      *****
7
8      *****
9      */
10
11 #include "stm32f4xx.h"
12 #include "../led/bsp_led.h"
13 #include "../key/bsp_exti.h"
14
15 void Delay(__IO u32 nCount);
16
17 /**
18  * @brief    主函数
19  * @param    无
20  * @retval    无
21  */
22 int main(void)
23 {
24     /* LED 端口初始化 */
25     LED_GPIO_Config();
26     /* 蜂鸣器 端口初始化 */
27     BEEP_GPIO_Config();
28
29     /* 初始化 EXTI 中断，按下按键会触发中断，
30  * 触发中断会进入 stm32f4xx_it.c 文件中的函数
31  * KEY1_IRQHandler 和 KEY2_IRQHandler，处理中断，反转 LED 灯。
32  */
33     EXTI_Key_Config();
34
```

```
35     /* 等待中断，由于使用中断方式，CPU 不用轮询按键 */
36     while(1)
37     {
38     }
39 }
40
41 /*****END OF FILE*****/
```

然后修改 stm32f4xx_it.c，添加中断处理函数

```
1  /**
2  ****
3  * @file    FMC_SDRAM/stm32f4xx_it.c
4  * @author  MCD Application Team
5  * @version V1.0.1
6  * @date    11-November-2013
7  * @brief   Main Interrupt Service Routines.
8  *          This file provides template for all exceptions handler and
9  *          peripherals interrupt service routine.
10 ****
11 * @attention
12 *
13 * <h2><center>©copy; COPYRIGHT 2013 STMicroelectronics</center></h2>
14 *
15 * Licensed under MCD-ST Liberty SW License Agreement V2, (the "License");
16 * You may not use this file except in compliance with the License.
17 * You may obtain a copy of the License at:
18 *
19 *     http://www.st.com/software_license_agreement_liberty_v2
20 *
21 * Unless required by applicable law or agreed to in writing, software
22 * distributed under the License is distributed on an "AS IS" BASIS,
23 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
24 * See the License for the specific language governing permissions and
25 * limitations under the License.
26 *
27 ****
28 */
29
30 /* Includes -----*/
31 #include "stm32f4xx_it.h"
32 #include "../led/bsp_led.h"
33 #include "../key/bsp_exti.h"
34 #include "../beep/bsp_beep.h"
35
36 /** @addtogroup STM32F429I_DISCOVERY_Examples
37 * @{
38 */
39
40 /** @addtogroup FMC_SDRAM
41 * @{
42 */
43
```

```
44  /* Private typedef -----*/
45  /* Private define -----*/
46  /* Private macro -----*/
47  /* Private variables -----*/
48  /* Private function prototypes -----*/
49  /* Private functions -----*/
50
51  /*****
52  /*          Cortex-M4 Processor Exceptions Handlers          */
53  *****/
54
55  // Add by me
56
57  void Delay(__IO u32 nCount);
58
59  /**
60   * @brief This function handles NMI exception.
61   * @param None
62   * @retval None
63   */
64  void NMI_Handler(void)
65  {
66  }
67
68  /**
69   * @brief This function handles Hard Fault exception.
70   * @param None
71   * @retval None
72   */
73  void HardFault_Handler(void)
74  {
75      /* Go to infinite loop when Hard Fault exception occurs */
76      while (1)
77      {}
78  }
79
80  /**
81   * @brief This function handles Memory Manage exception.
82   * @param None
83   * @retval None
84   */
85  void MemManage_Handler(void)
86  {
87      /* Go to infinite loop when Memory Manage exception occurs */
88      while (1)
89      {}
90  }
91
92  /**
93   * @brief This function handles Bus Fault exception.
94   * @param None
95   * @retval None
96   */
97  void BusFault_Handler(void)
```

```
98 {
99     /* Go to infinite loop when Bus Fault exception occurs */
100     while (1)
101     {}
102 }
103
104 /**
105  * @brief This function handles Usage Fault exception.
106  * @param None
107  * @retval None
108  */
109 void UsageFault_Handler(void)
110 {
111     /* Go to infinite loop when Usage Fault exception occurs */
112     while (1)
113     {}
114 }
115
116 /**
117  * @brief This function handles Debug Monitor exception.
118  * @param None
119  * @retval None
120  */
121 void DebugMon_Handler(void)
122 {}
123
124 /**
125  * @brief This function handles SVCcall exception.
126  * @param None
127  * @retval None
128  */
129 void SVC_Handler(void)
130 {}
131
132 /**
133  * @brief This function handles PendSV_Handler exception.
134  * @param None
135  * @retval None
136  */
137 void PendSV_Handler(void)
138 {}
139
140 /**
141  * @brief This function handles SysTick Handler.
142  * @param None
143  * @retval None
144  */
145 void SysTick_Handler(void)
146 {}
147
148 /************************************************************************
149 /*          STM32F4xx Peripherals Interrupt Handlers          */
150 /* Add here the Interrupt Handler for the used peripheral(s) (PPP), for the */
151 /* available peripheral interrupt handler's name please refer to the startup */
```

```
152  /* file (startup_stm32f429_439xx.s). */
153  /*****
154
155  /**
156   * @}
157   */
158
159  void KEY1_IRQHandler(void)
160  {
161      //确保是否产生了 EXTI Line 中断
162      if(EXTI_GetITStatus(KEY1_INT_EXTI_LINE) != RESET)
163      {
164          // // LED1 取反
165          // LED1_TOGGLE;
166          // 实现按下 key1 键，红灯、黄灯，绿灯按顺序亮，
167
168          // 红
169          LED_RED;
170          Delay(0xFFFFFFFF);
171          // 黄
172          LED_YELLOW;
173          Delay(0xFFFFFFFF);
174          // 绿
175          LED_GREEN;
176          Delay(0xFFFFFFFF);
177          LEDRGBOFF;
178          Delay(0xFFFFFFFF);
179
180          //清除中断标志位
181          EXTI_ClearITPendingBit(KEY1_INT_EXTI_LINE);
182      }
183  }
184
185  void KEY2_IRQHandler(void)
186  {
187      // 确保是否产生了 EXTI Line 中断
188      if(EXTI_GetITStatus(KEY2_INT_EXTI_LINE) != RESET)
189      {
190          // 按下 key2 键，绿灯。黄灯，红灯按顺序亮。红灯亮时蜂鸣器响。
191
192          // 绿
193          LED_GREEN;
194          Delay(0xFFFFFFFF);
195          // 黄
196          LED_YELLOW;
197          Delay(0xFFFFFFFF);
198          // 红
199          LED_RED;
200          BEEP(ON);
201          Delay(0xFFFFFFFF);
202          LEDRGBOFF;
203          BEEP(OFF);
204
205          //清除中断标志位
```

```
206         EXTI_ClearITPendingBit(KEY2_INT_EXTI_LINE);
207     }
208 }
209
210 void Delay(__IO uint32_t nCount) // 简单的延时函数
211 {
212     for (; nCount != 0; nCount--)
213         ;
214 }
215 /**
216  * @}
217  */
218
219 /***** (C) COPYRIGHT STMicroelectronics *****/
```

2. 下载运行

使用 FlyMCU.exe 下载程序到 STM32 开发板上，观察实验现象。

3. 实验现象

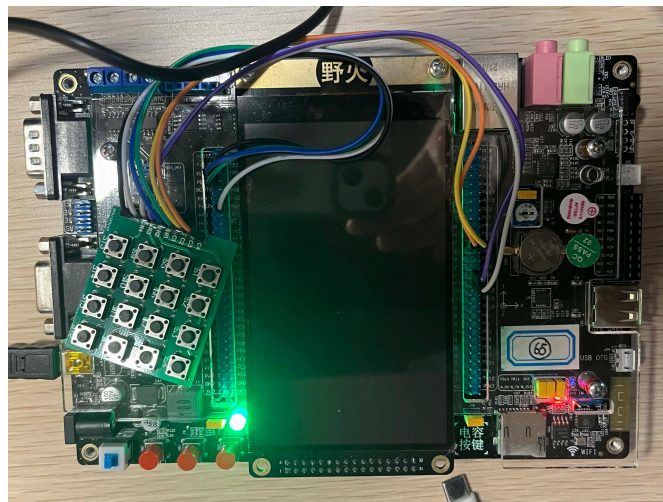


图 1: 绿灯亮

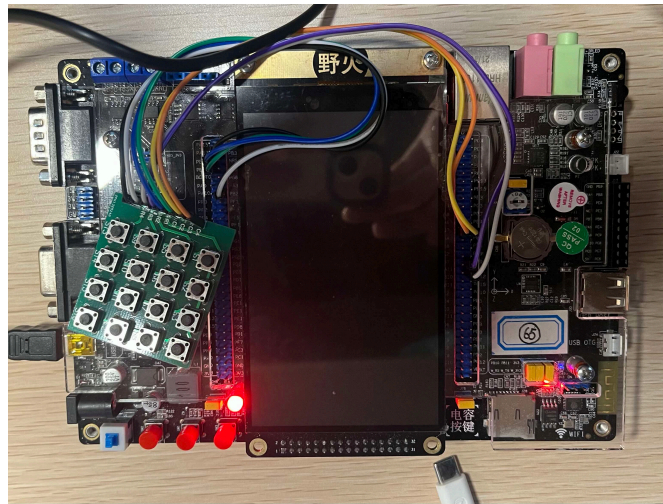


图 2: 红灯亮

三、实验小结

1. 按下 key1 键，红灯、黄灯、绿灯依次亮起。
2. 按下 key2 键，绿灯、黄灯、红灯依次亮起，且红灯亮时蜂鸣器响。

通过本实验，掌握了 STM32 外部中断的配置和使用，并能够根据实际需求修改程序实现相应功能。