

Embedded Systems Design (2022)

Report on LAB 1

- Student: JUNYAN, YANG (杨钧彦)
- Student Number: 212320028
- Date: 16/09/2022

1. Task Statement

Variants

Variant	LED blink periods, ms	LED order
1	100/300	RGBG
2	50/500	RBGB
3	300/100	RRGB
4	1000/100	BBRG
5	200/400	RGRB
6	500/1000	RGGB
7	30/300	RRBG
8	30/600	RBBG
9	400/800	RBGG
10	500/100	BRGR

Note 1. The program should run in a following way: the first LED is blinking (e.g., red) with the initial frequency. After the interrupt occurs, the frequency changes. At the next interrupt, the LED changes (e.g, to green) while the frequency stays. At the next interrupt, the frequency changes again but the LED does not.

Note 2. The variants table contains periods values that should be used to switch between. The duty cycle is set to be 50% that means $\text{time}_{\text{OFF}} = \text{time}_{\text{ON}} = \text{period} / 2$

2. Environment:

Win10, STM32CubeIDE

3. Screenshot for lab1:

main.c

```
44 /* USER CODE BEGIN PV */
45 int udelay = 30;
46 uint16_t color;
47 int step = 0;
48 /* USER CODE END PV */
```

```
94 /* Infinite loop */
95 /* USER CODE BEGIN WHILE */
96 while (1)
97 {
98     HAL_GPIO_TogglePin(GPIOB, color);
99     HAL_Delay(udelay);
100 /* USER CODE END WHILE */
101
102 /* USER CODE BEGIN 3 */
103 }
104 /* USER CODE END 3 */
105 }
```

stm32f4x_it.c

```

60 /* USER CODE BEGIN EV */
61 extern udelay;
62 extern color;
63 extern step;

207 void EXTI15_10_IRQHandler(void)
208 {
209     /* USER CODE BEGIN EXTI15_10_IRQn 0 */
210     for(int i=0;i<65535;i++){
211         if(HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_13)){
212             if (step%8 == 0){
213                 udelay = 30;
214                 color = GPIO_PIN_14;
215             }
216             if (step%8 == 1){
217                 udelay = 600;
218                 color = GPIO_PIN_14;
219             }
220             if (step%8 == 2){
221                 udelay = 600;
222                 color = GPIO_PIN_7;
223             }
224             if (step%8 == 3){
225                 udelay = 30;
226                 color = GPIO_PIN_7;
227             }
228             if (step%8 == 4){
229                 udelay = 600;
230                 color = GPIO_PIN_7;
231             }
232             if (step%8 == 5){
233                 udelay = 30;
234                 color = GPIO_PIN_7;
235             }
236             if (step%8 == 6){
237                 udelay = 30;
238                 color = GPIO_PIN_0;
239             }
240             if (step%8 == 7){
241                 udelay = 600;
242                 color = GPIO_PIN_0;
243             }
244             step++;
245             HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_RESET);
246             HAL_GPIO_WritePin(GPIOB, GPIO_PIN_7, GPIO_PIN_RESET);
247             HAL_GPIO_WritePin(GPIOB, GPIO_PIN_14, GPIO_PIN_RESET);
248         }
249     /* USER CODE END EXTI15_10_IRQn 0 */
250     HAL_GPIO_EXTI_IRQHandler(GPIO_PIN_13);
251     /* USER CODE BEGIN EXTI15_10_IRQn 1 */
252
253     /* USER CODE END EXTI15_10_IRQn 1 */
254 }
255
256 /* USER CODE BEGIN 1 */
257

```