

# 嵌入式系统课程设计

题目名称: **GPIO 端口实验要求**



姓名: 王云鹏

学号: **8213180228**

专业: 物联网工程

班级: **1802**

指导教师: 贺建飏

编写日期: **2021.6.2**

## 目录

实例复现.....	3
1. 问题描述.....	3
2. 实验设备.....	3
3. 系统设计.....	3
4. 源代码清单.....	4
5. 运行结果测试与分析.....	5
6. 结论与心得.....	6
摩斯电码闪灯.....	7
1. 问题描述.....	7
2. 实验设备.....	7
3. 系统设计（关键 20 行代码）.....	7
4. 源代码清单.....	10
5. 运行结果测试与分析.....	13
6. 结论与心得.....	14

# 实例复现

## 1. 问题描述

- 熟悉并掌握 Keil MDK 开发环境的使用以及在线调试方法；
- 掌握 STM32F746NG 芯片 GPIO 端口寄存器的配置；
- 通过实验掌握 Cortex-M7 控制 GPIO 端口的方法，实现对 LED 的控制。
- 编写程序，对指定 GPIO 端口进行初始化并完成配置过程，实现对 LED 的控制。学习使用超级终端，对其进行配置并完成串口调试。实验中观察 GPIO 端口输出数据寄存器（GPIOx\_ODR）的值对 LED 灯的明灭的影响，学习 GPIO 端口的输入输出方式、输出类型和输出速度的设置方法。

## 2. 实验设备

系统：Windows10

IDE：µVision IDEfor ARM 集成开发环境

## 3. 系统设计

main 函数中首先进行系统初始化与配置，然后是死循环亮灯，其中 HAL\_GPIO\_TogglePin 函数的作用是改变灯的亮灭，HAL\_Delay 的作用是延时一段时间

```
int main(void)
{
    System_Init();//系统初始化

    GPIO_Config();//GPIO 配置

    printf("\n\rExample finished\n\r");

    /* Toggle IOs in an infinite loop */
    while (1)
    {
        HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);//改变管脚状态
        /* Insert delay */
        HAL_Delay(delay);//延迟一段时间
    }
}
```

## 4. 源代码清单

### main.c

```
/**
 * 本程序演示了GPIO的PA8引脚的初始化及配置。
 * 通过LED灯来显示GPIO的状态。
 */
#include "main.h"
#include "system_init.h"

/* Private variables */
uint16_t delay = 100;

void System_Init(void);
void GPIO_Config(void);

/* main */
int main(void)
{
    System_Init();//系统初始化

    GPIO_Config();//GPIO 配置

    printf("\n\rExample finished\n\r");

    /* Toggle IOs in an infinite loop */
    while (1)
    {
        HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);//改变管脚状态
        /* Insert delay */
        HAL_Delay(delay);//延迟一段时间
    }
}
```

### config.c

```
/**
 * GPIO配置
 */
#include "main.h"
```

```

static GPIO_InitTypeDef  GPIO_InitStruct;

void GPIO_Config(void)
{
    /* Enable each GPIO Clock (to be able to program the configuration registers) */
    __HAL_RCC_GPIOA_CLK_ENABLE();

    /* Configure IOs in output push-pull mode to drive external LED */
    GPIO_InitStruct.Mode   = GPIO_MODE_OUTPUT_PP; // 22222222
    GPIO_InitStruct.Pull   = GPIO_PULLUP;         // 2222
    GPIO_InitStruct.Speed  = GPIO_SPEED_HIGH;     // 2222

    GPIO_InitStruct.Pin = GPIO_PIN_8;
    HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);      // 22222222PA8
}

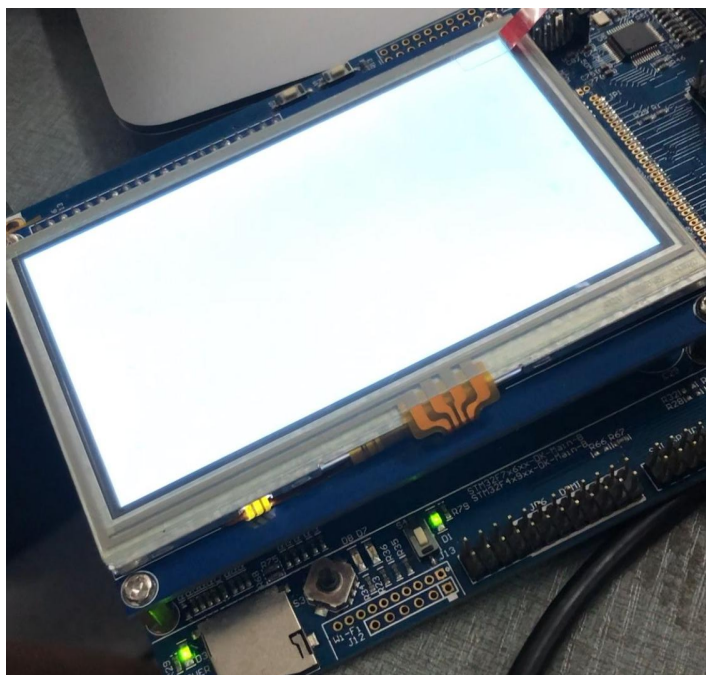
```

## 5. 运行结果测试与分析

### D1 熄灭



### D1 亮起



## 6. 结论与心得

- 掌握 Keil MDK 开发环境的使用以及在线调试方法；
- 掌握 STM32F746NG 芯片 GPIO 端口寄存器的配置；
- 掌握 Cortex-M7 控制 GPIO 端口的方法，实现对 LED 的控制。

# 摩斯电码闪灯

## 1. 问题描述

请自行搜索摩尔斯密码表，通过控制 D1 的亮灭间隔，实现自己姓氏拼音的电码实现。

## 2. 实验设备

系统：Windows10

IDE：µVision IDEfor ARM 集成开发环境

## 3. 系统设计（关键 20 行代码）

摩斯电码规则如下：

**标准摩尔斯电码对照表**



Morse-Alphabet (Punkt = kurz blinken, Strich = lang blinken.)			
a · -	i · ·	r · - ·	1 · - - -
ä · - · -	j · - - -	s · · ·	2 · · - - -
b - · · ·	k - · -	t -	3 · - · - -
c - · - ·	l · - · ·	u · · ·	4 · · · - -
ch - - - -	m - -	ü · · · -	5 · · · · -
d - · ·	n - ·	v · · - -	6 - · · · -
e ·	o - - -	w · - - -	7 - - · · ·
f · · - ·	ö - - - ·	x - · · -	8 - - - · ·
g - · - ·	p · - · ·	y - · - -	9 - - - · -
h · - · ·	q - - - ·	z - - - ·	0 - - - - ·
Verstanden · · · - · · ·			
Schlusszeichen · - · - ·			

摩尔斯电码由两种基本信号组成：短促的点信号“·”，读“滴”；保持一定时间的长信号“—”，读“嗒”。间隔时间：滴=1t，嗒=3t，滴嗒间=1t，字符间=3t，单词间=7t。

笔者姓氏拼音为“wang”

w: · - -

a: · -

n: - ·

g: - · -

则代码如下

```
while (1)
{
    //start
    //w
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay*3);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay*3);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay*3);

    //a
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay*3);

    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
```



```
HAL_Delay(delay*3);

//n
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

//g
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);
}
```

## 4. 源代码清单

### main.c

```
/**
 * 0÷³İĐò£ºİμİ³Êİμç³õÊ%»~ªóŸÔGPIOÖĐμÄPA8½øĐĐÄüÖÃ£~½«ÆüÄüÖÃÎªÊä³öŸÊ¿Ú²¿¿ØÖÆLEDÉÁÊ, ¿£
 */
#include "main.h"
#include "system_init.h"

/* Private variables */
uint16_t delay = 100;

void System_Init(void);
void GPIO_Config(void);

/* main */
int main(void)
{
    System_Init();

    GPIO_Config();

    printf("\n\rExample finished\n\r");

    /* Toggle IOs in an infinite loop */
    HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
    /* Insert delay */
    HAL_Delay(delay*10);
    while (1)
    {
        //start
        //w
        HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
        /* Insert delay */
        HAL_Delay(delay);

        HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
        /* Insert delay */
        HAL_Delay(delay);

        HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
        /* Insert delay */
        HAL_Delay(delay);
    }
}
```

```
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

//a
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

//n
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);
```

```

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);

/* Insert delay */
HAL_Delay(delay*3);

//g
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay);

HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
/* Insert delay */
HAL_Delay(delay*3);
}
}

```

## config.c

```

/**
 * GPIO
 */

#include "main.h"

static GPIO_InitTypeDef GPIO_InitStruct;

void GPIO_Config(void)
{

```

```

/* Enable each GPIO Clock (to be able to program the configuration registers) */
__HAL_RCC_GPIOA_CLK_ENABLE();

/* Configure IOs in output push-pull mode to drive external LED */
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP; // 22222222
GPIO_InitStruct.Pull = GPIO_PULLUP;        // 2222
GPIO_InitStruct.Speed = GPIO_SPEED_HIGH;    // 2222

GPIO_InitStruct.Pin = GPIO_PIN_8;
HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);    // 22222222PA8
}

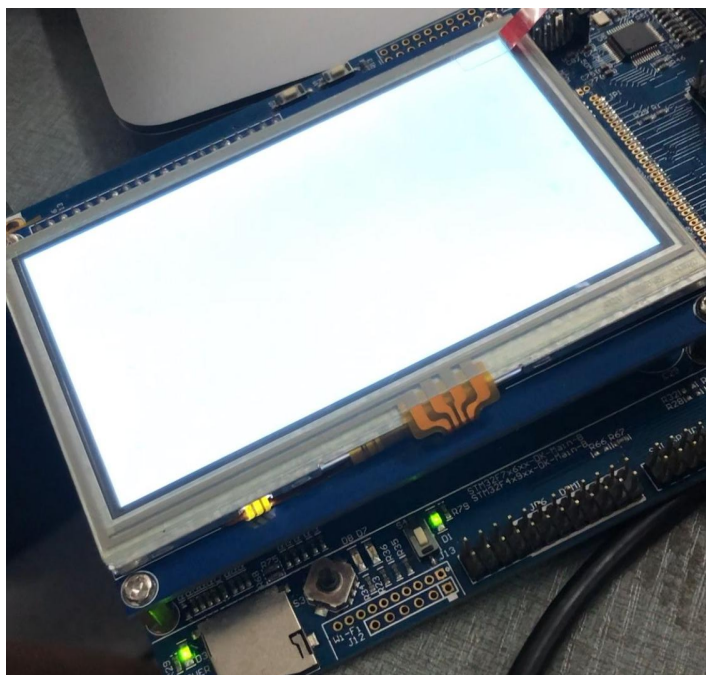
```

## 5. 运行结果测试与分析

D1 熄灭



D1 亮起



## 6. 结论与心得

- 掌握 Keil MDK 开发环境的使用以及在线调试方法；
- 掌握 STM32F746NG 芯片 GPIO 端口寄存器的配置；
- 掌握 Cortex-M7 控制 GPIO 端口的方法，实现对 LED 的控制。