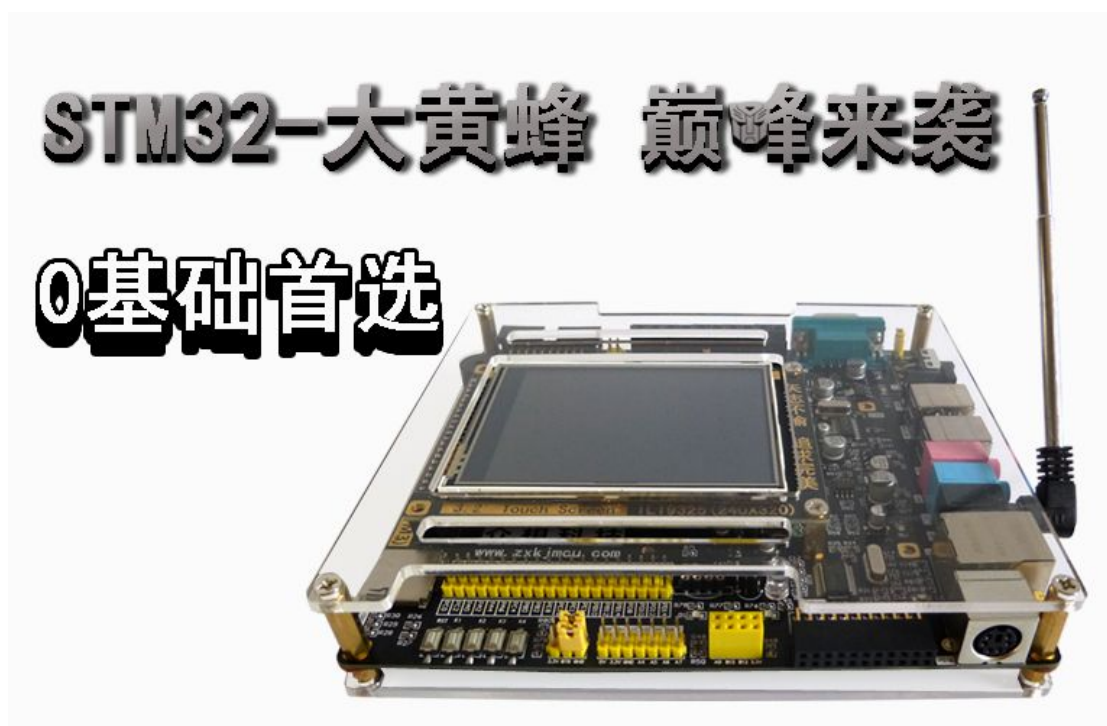


# 学 ARM 从 STM32 开始

STM32 开发板库函数教程——实战篇



官方网站: <http://www.zxkjmcu.com>

官方店铺: <http://zxkjmcu.taobao.com>

官方论坛: <http://bbs.zxkjmcu.com>

刘洋课堂: <http://school.zxkjmcu.com>

## 4.3 STM32 独立键盘操作实验

### 4.3.1 概述

键盘是人机通讯的基本设备，它广泛应用于微型计算机和各种终端设备上。各种电子设备操作者通过键盘向计算机输入各种指令、数据，指挥电子设备的工作。电子设备的运行情况输出到显示器，操作者可以很方便地利用键盘和显示器与电子设备对话，对程序进行修改、编辑，控制和观察电子设备的运行。所以键盘在电子产品中占着举足轻重的地位，作为一名程序设计人员一定要熟练编写键盘的程序，对键盘的工作原理做到心中有数。

大家最熟悉的就是计算机键盘，在 STM32 独立键盘操作实验中，我们的键盘没有那么复杂，但基本功能都具备，让初学者通过单个键盘程序的编写，熟悉人机交互的功能。

### 4.3.2 目的

使用 STM32 芯片实现键盘的操作，目的是通过键盘控制 LED 发光二极管，当键盘按下时 LED 发光，当键盘抬起时 LED 熄灭。

### 4.3.3 硬件设计

大黄蜂 STM32 开发板上有 5 个独立键盘，从左至右编号分别是复位键、K1、K2、K3、K4。PCB 线路板上的标号分别是 RESET、K1、K2、K3、K4。与原理图对应的标号是 (14. NRST、34. PC5-COL1、16. PC1-COL2、17. PC2-COL3、18. PC3-COL4)。原理图上标号的命名规则上节已经详细说明，这里就不再详细说明。

原理图与 PCB 线路板上已经都连接好了，大家可以参考电路图纸，不用自己搭接环境了，直接把程序下载到大黄蜂 STM32 开发板上运行就可以了。

### 独立键盘原理图

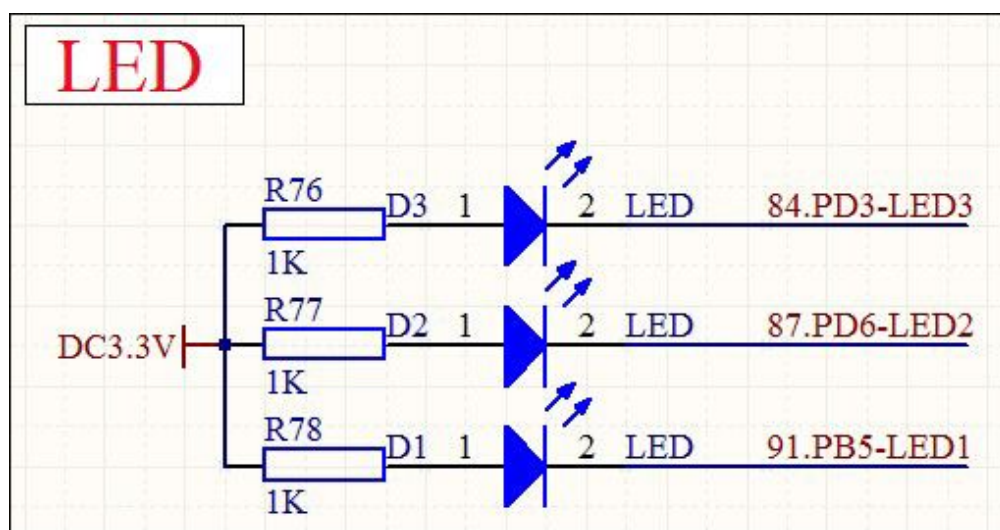
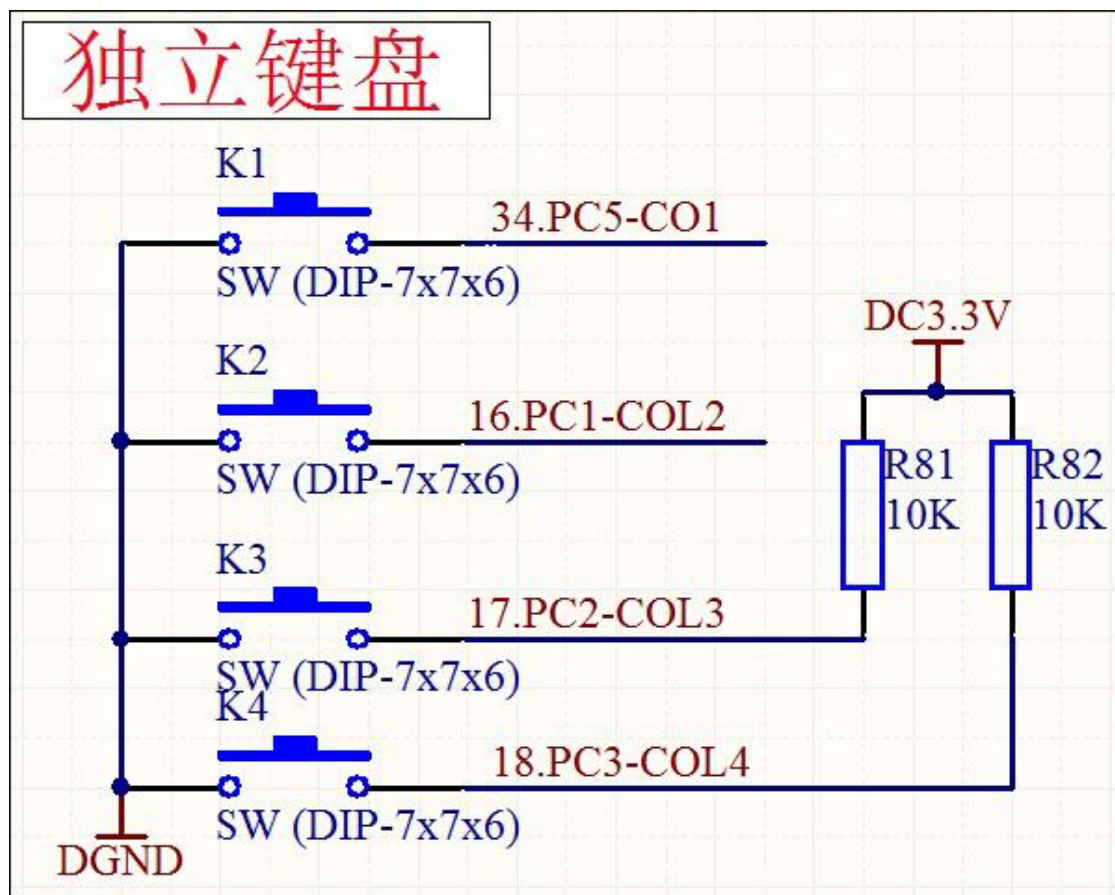


图 4.3

## 4.3.4 软件设计

### 4.3.4.1 STM32 库函数文件

```
stm32f10x_gpio.c  
stm32f10x_rcc.c
```

本节实验我们主要用到的库文件，其中 `stm32f10x_gpio.h` 头文件包含了 GPIO 端口的定义。`stm32f10x_rcc.h` 头文件包含了系统时钟配置函数以及相关的外设时钟使能函数，所以我们要把这两个头文件对应的 `stm32f10x_gpio.c` 和 `stm32f10x_rcc.c` 都加到工程中。

### 4.3.4.2 自定义头文件

```
pbdata.h  
pbdata.c
```

同时我们自己也创建了两个公共的文件，这两个文件主要存放我们自定义的公共函数和全局变量，以方便以后每个功能模块之间传递参数。

### 4.3.4.3 pbdata.h 文件里的内容是

```
#ifndef _pbdata_H  
#define _pbdata_H  
#include "stm32f10x.h"  
extern u8 dt;//定义变量  
void delay(u32 nCount);//定义函数  
#endif
```

语句 `#ifndef`、`#endif` 是为了防止 `pbdata.h` 文件被多个文件调用时出现错误提示。如果不加这两条语句，当两个文件同时调用 `pbdata` 文件时，会提示重复调用错误。`stm32f10x.h` 头文件是我们每个工程都需要调用的，里面包括了 STM32 内部寄存器地址的定义。

### 4.3.4.4 pbdata.c 文件里的内容是

```
#include "pbdata.h" //很重要，引用这个头文件  
u8 dt=0;  
void delay(u32 nCount)  
{
```

```
for(;nCount!=0;nCount--);  
}
```

这个实验的目的很简单，主要是检验键盘操作的反应，凭感觉按键盘给人的感觉是不明显，所以让 LED 配合键盘的动作，这样给人的感觉直观可信。所以 pbdata.c 文件基本没有什么。

#### 4.3.5 STM32 系统时钟配置 SystemInit()

每个工程都必须在开始时配置并启动 STM32 系统时钟。关于 STM32 系统时钟我们在前面的章节中已经详细的说明了，在这里就不再叙述了。

#### 4.3.6 GPIO 引脚时钟使能

```
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE);  
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOC, ENABLE);
```

本节实验用到了 PB 和 PC 端口，所以要把 PB 和 PC 端口的时钟打开。

#### 4.3.7 GPIO 管脚电平控制函数

在主程序中采用 while(1) 循环语句，程序查询 PC5 端口的高低电平变化，来判断是否有键盘被按下，如果监测到键盘被按下，程序按照顺序执行，使 LED 发光，当键盘抬起时，程序跳转到“else”后，熄灭 LED。下面是 while(1) 循环体内的执行程序语句。

```
if(GPIO_ReadInputDataBit(GPIOC, GPIO_Pin_5) == Bit_RESET)  
{  
    //LED 发光  
    GPIO_ResetBits(GPIOB, GPIO_Pin_5);  
}  
else  
{  
    //LED 熄灭  
    GPIO_SetBits(GPIOB, GPIO_Pin_5);  
}
```

#### 4.3.8 main.c 文件里的内容是

```
#include "pbdata.h" //很重要，引用这个头文件  
void RCC_Configuration(void);
```

```
void GPIO_Configuration(void);

int main(void)
{
    RCC_Configuration(); //系统时钟初始化
    GPIO_Configuration(); //端口初始化
    while(1)
    {
        if(GPIO_ReadInputDataBit(GPIOC, GPIO_Pin_5)==Bit_RESET)
        {
            GPIO_ResetBits(GPIOB, GPIO_Pin_5); //LED 发光
        }
        else
        {
            GPIO_SetBits(GPIOB, GPIO_Pin_5); //LED 熄灭
        }
    }
}

void RCC_Configuration(void)
{
    SystemInit();
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE);
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOC, ENABLE);
}

void GPIO_Configuration(void)
{
    GPIO_InitTypeDef GPIO_InitStructure;
    //LED
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_5;
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;
    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_Out_PP; //设置为输出模式
    GPIO_Init(GPIOB, &GPIO_InitStructure);

    //KEY
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_5;
    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_IPU; //设置为输入模式
    GPIO_Init(GPIOC, &GPIO_InitStructure);
}
```

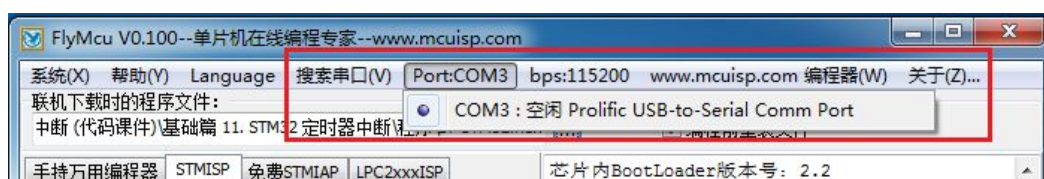
在 main(void) 程序体中代码不多，开始调用了两个函数，分别是系统时钟初始化函数 RCC\_Configuration() 和端口初始化函数



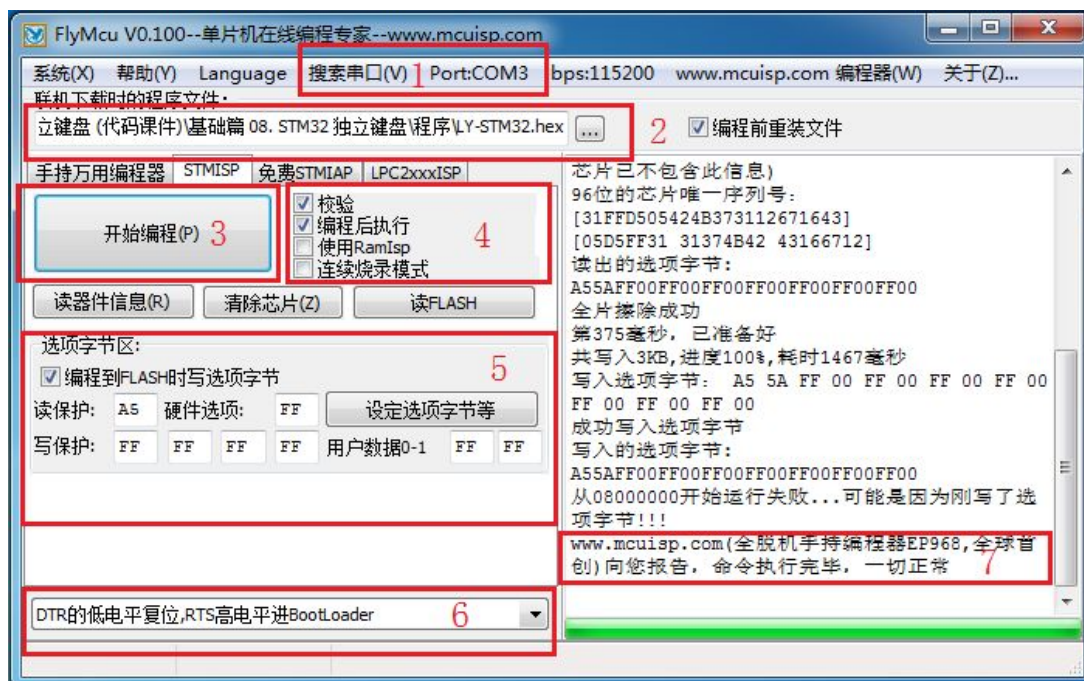
GPIO\_Configuration(), 接下来进入 while(1) 循环体, 在循环体中, 套用 if 语句, 来实现查询方式判断键盘是否按下, 如果有键盘按下就顺序执行, 点亮 LED, 如果没有键盘按下就跳转到 "else" 后, 熄灭 LED。

### 4.3.9 程序下载

请根据下图所指向的 7 个重点区域配置。其中 (1) 号区域根据自己机器的实际情况选择, 我的机器虚拟出来的串口号是 COM3。



(2) 号区域请自己选择程序所在的文件夹。(7) 号区域当程序下载完后, 进度条会到达最右边, 并且提示一切正常。(4、5、6) 号区域一定要按照上图显示的设置。当都设置好以后就可以直接点击 (3) 号区域的开始编程按钮下载程序了。



本节实验的源代码在光盘中: (LY-STM32 光盘资料\1. 课程\1, 基础篇\基础

篇 08. STM32 独立键盘\程序)

#### 4.3.10 实验效果图

独立键盘通过图片看不出来,通过 LED 发光二极管来直观的看出键盘功能。请学习者通过实验来体验程序实际效果,以上程序比较好理解,只要认真看书和看视频,一定会产生事半功倍的效果,再加上自己的亲自操作和按照自己的想法修改程序,一定会印象深刻的。