目录

[Stm32基本内容 2](#_Toc57835491)

[STM32M3芯片结构 3](#_Toc57835492)

[GPIO 6](#_Toc57835493)

[时钟 12](#_Toc57835494)

[时钟源： 12](#_Toc57835495)

[滴答定时器 13](#_Toc57835496)

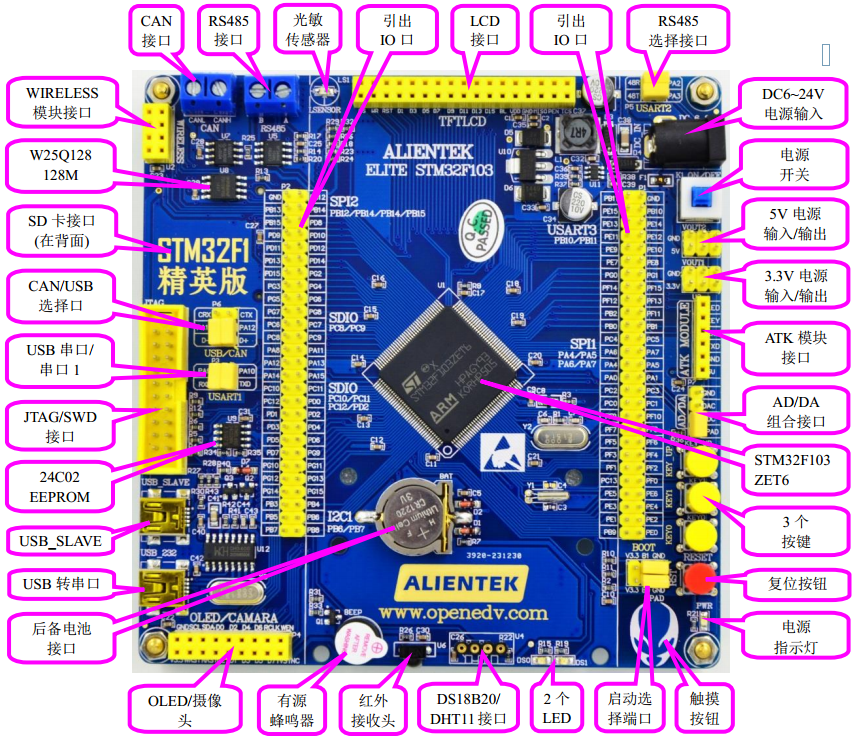
[通用定时器TIM 15](#_Toc57835497)

[定时器中断 17](#_Toc57835498)

[PWM输出： 18](#_Toc57835499)

[输入捕获输出比较 19](#_Toc57835500)

# Stm32基本内容



* 1.1 什么是STM32 我们用的STM32F103ZET6.
* 新的基于ARM内核的32位MCU系列
  + 内核为ARM公司为要求高性能、低成本、低功 耗的嵌入式应用专门设计的Cortex-M内核
  + 标准的ARM架构
* 超前的体系结构
  + 高性能
  + 低电压
  + 低功耗
  + 创新的内核以及外设
* 简单易用/自由/低风险
* 1.2 Cortex内核
* Cortex-M3系列属于ARMv7架构

ARMv7架构定义了三大分工明确的系列：

“A”系列：面向尖端的基于虚拟内存的操作系统和用户应用

“R”系列：针对实时系统；

“M”系列：对微控制器。

* STM32F1属于Cortex-M系列中的Cortex-M3内核，采用ARMv7-M架构。STM32F4属于Cortex-M4系列采用ARMv7-ME架构。 Cortex-A5/A8采用ARMv7-A架构。传统的ARM7系列采用的是ARMv4T架构。
* 2.1 CM3内核优点

1. **高性能Cortex-M内核**

**-采用ARM公司流行的标准内核Cortex-M3**

**-低动态功耗上实现的高性能**

* + **哈佛结构上实现1.25DMIPS/MHZ，功耗只有0.19mv/MHZ**

**-比ARM7TDMI改进了30%**

* + **单周期的乘法和硬件除法**
  + **不可分的位操作，实现对RAM，I/O和寄存器的最优访问。**

**2.最佳的代码密度**

**Thumb-2指令集以16位指令的密度实现32位指令性能（与ARM7TDMI的**

**ARM模式比减少了30%-45%的代码量）**

3.可预见的运行时间

-中断控制器嵌在内核之中，中断之间的间隔最少可达6个

CPU周期。

-从低功耗模式唤醒只需6个CPU周期

4.改进的调试功能

-串行单步调试和JTAG调试

Stm32优势：

1）极高的性能： 主流的Cortex内核。

2）丰富合理的外设，合理的功耗，合理的价格。

3）强大的软件支持：丰富的软件包。

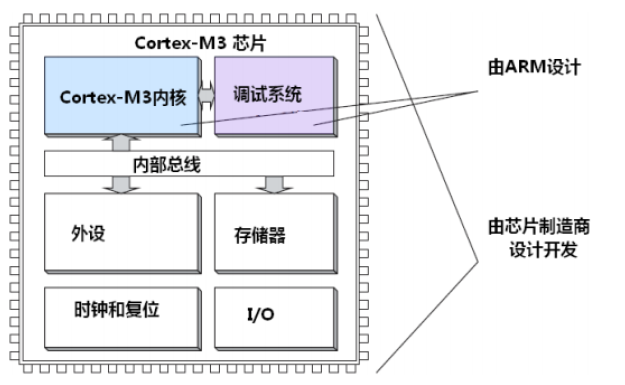
4）全面丰富的技术文档。

5）芯片型号种类多，覆盖面广。

6）强大的用户基础：最先成功试水CM3芯片的公司，积累

了大批的用户群体，为其领先做铺垫。

## STM32M3芯片结构







* 内核：

-32位 高性能ARM Cortex-M3处理器

-时钟：高达72M,实际还可以超屏一点点

-单周期乘法和硬件除法

**144引脚 112个IO**

**大部分IO口都耐5V(模拟通道除外)**

**-支持调试：SWD和JTAG，SWD只要2根数据线**

**512K FLASH，64K SRAM**

* **时钟，复位和电源管理：**

1. **2.0~3.6V电源和IO电压**
2. **上电复位，掉电复位和可编程的电压监控**
3. **强大的时钟系统**

**-4~16M的外部高速晶振**

**-内部8MHz的高速RC振荡器**

**-内部40KHz低速RC振荡器，看门狗时钟**

**-内部锁相环（PLL，倍频），一般系统时钟都是外**

**部或者内部高速时钟经过PLL倍频后得到**

**-外部低速32.768K的晶振，主要做RTC时钟源**

**低功耗：睡眠，停止和待机三种低功耗模式**

**-可用电池为RTC和备份寄存器供电**

* **AD:**

**3个12位AD【多达21个外部测量通道】**

**-转换范围：0~3.6(电源电压）**

**-内部通道可以用于内部温度测量**

**-内置参考电压**

**2个12位DA**

* **DMA:**

**12个DMA通道（7+5=12; 7通道DMA1,5通道DMA2)**

**支持外设：定时器，ADC,DAC，SDIO,I2S,SPI,I2C,和USART**

* **定时器：多达11个定时器**

**-4个通用定时器**

**-2个基本定时器**

**-2个高级定时器**

**-1个系统定时器**

**-2个看门狗定时器**

* **通信接口：多达13个通信接口**

**-2个I2C接口**

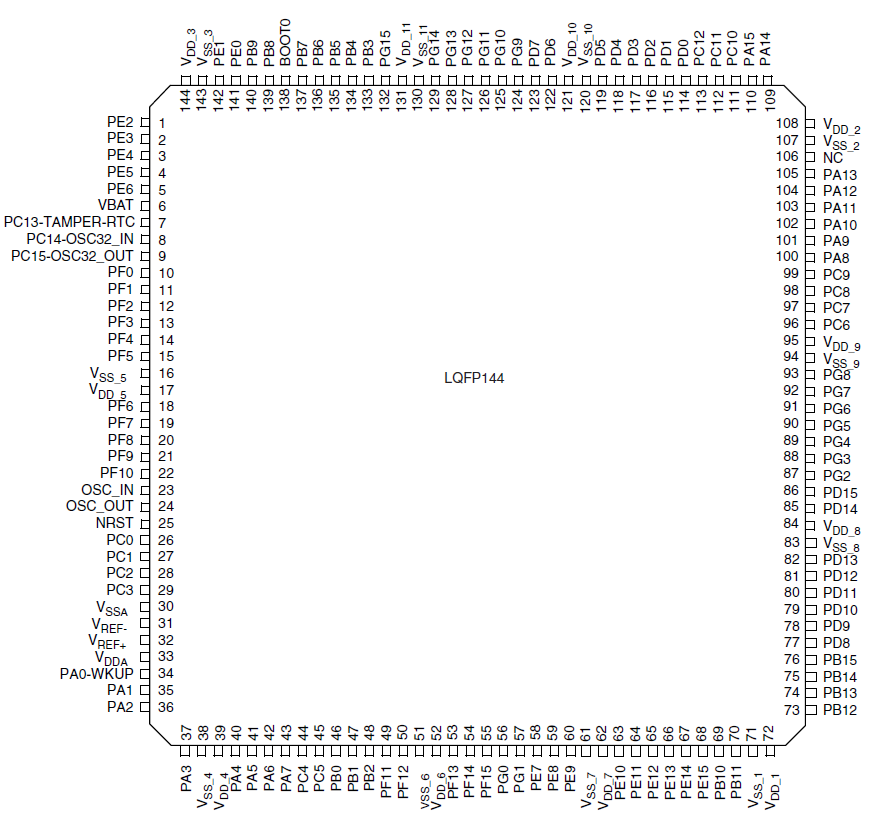
**-5个串口**

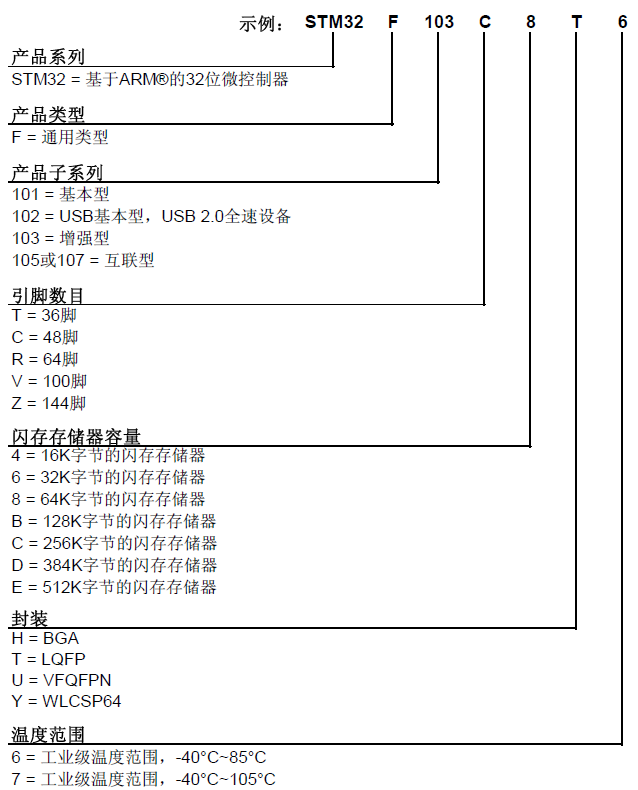
**-3个SPI接口**

**-1个CAN2.0**

**-1个USB FS**

**-1个SDIO**

****



# GPIO

一共有7组IO口A-G 每组IO口有16个IO 一共16X7=112个IO

（1）GPIO\_Mode\_AIN 模拟输入

（2）GPIO\_Mode\_IN\_FLOATING 浮空输入

（3）GPIO\_Mode\_IPD 下拉输入

（4）GPIO\_Mode\_IPU 上拉输入

（5）GPIO\_Mode\_Out\_OD 开漏输出 只可以输出强低电平，高电平得靠外部电阻拉高。输出端相当于三极管的集电极. 要得到高电平状态需要上拉电阻才行. 适合于做电流型的驱动,其吸收电流的能力相对强(一般20ma以内)

（6）GPIO\_Mode\_Out\_PP 推挽输出 可以输出强高低电平，连接数字器件

（7）GPIO\_Mode\_AF\_OD 复用开漏输出

（8）GPIO\_Mode\_AF\_PP 复用推挽输出

每组GPIO端口的寄存器包括：

两个32位配置寄存器(GPIOx\_CRL ，GPIOx\_CRH) ，

STM32每个端口有16个IO口。所以要分2个寄存器来操作。这2个寄存器用来选择对应的IO口，*IO0—7使用低*， *8-15使用高。*

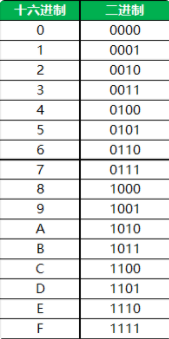
两个32位数据寄存器 (GPIOx\_IDR和GPIOx\_ODR)，

一个32位置位/ 复位寄存器(GPIOx\_BSRR)，

一个16位复位寄存器(GPIOx\_BRR)，

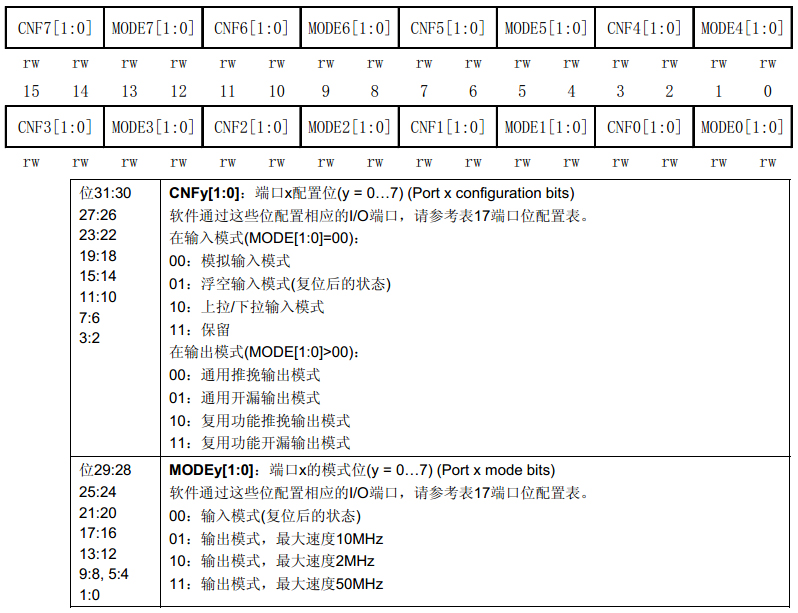
一个32位锁定寄存器(GPIOx\_LCKR)。

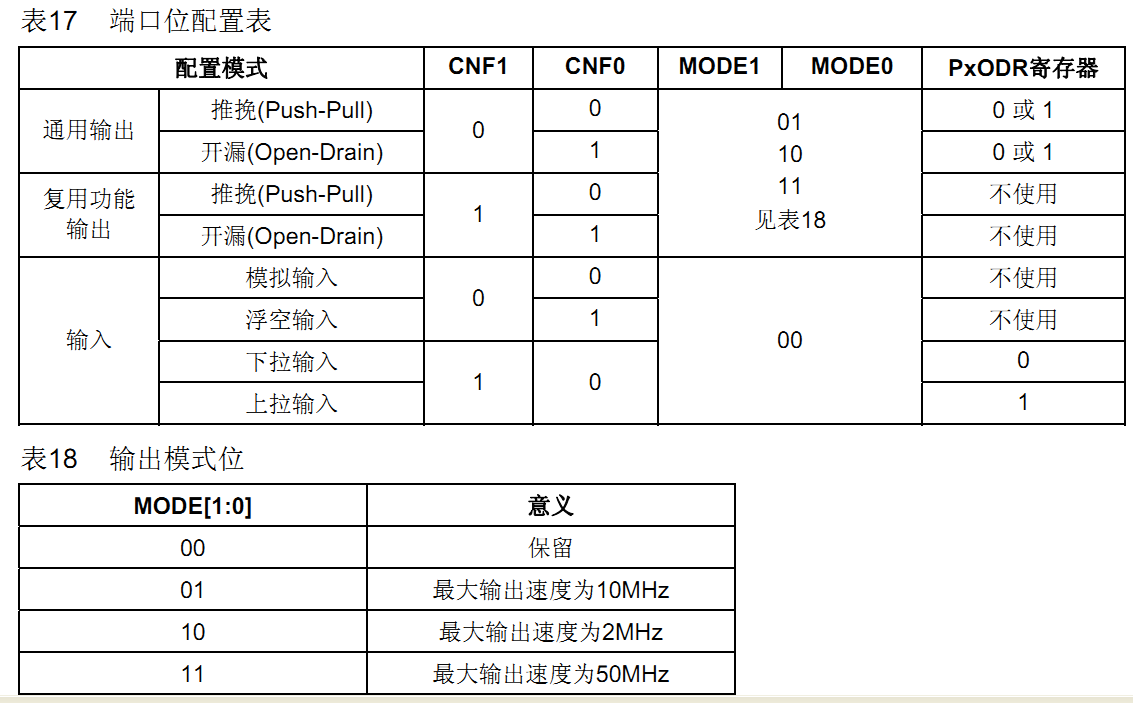
每个I/O端口位可以自由编程，然而I/O端口寄存器必须按32位字被访问(不允许半字或字节访问)



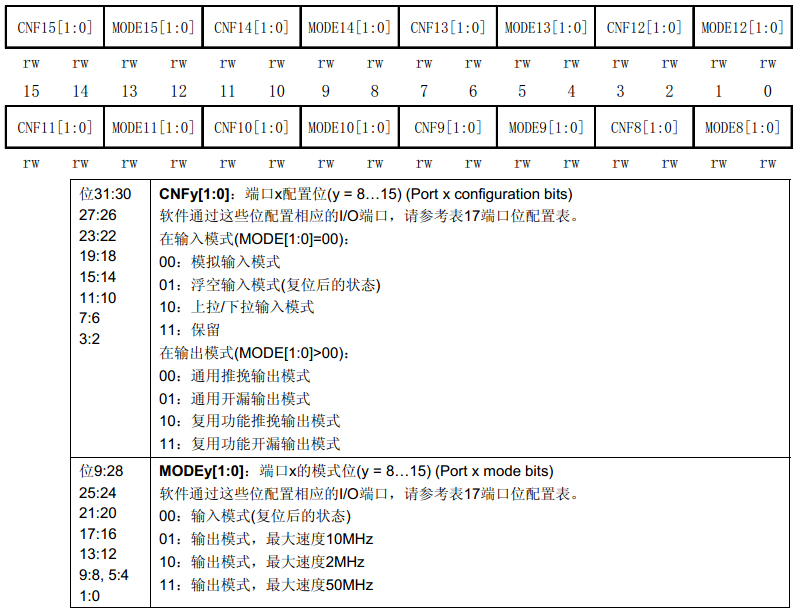
寄存器左高右低 FF0FFFFF = 1111 1111 0000 1111 1111 1111 1111 1111 四位四位换

**端口配置低寄存器(GPIOx\_CRL)**

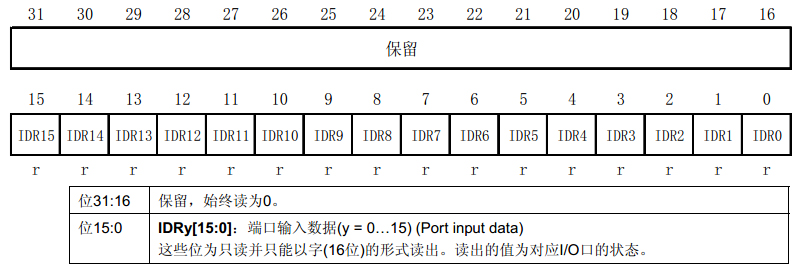




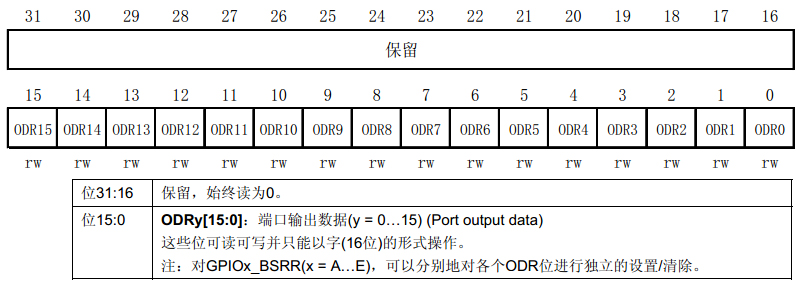
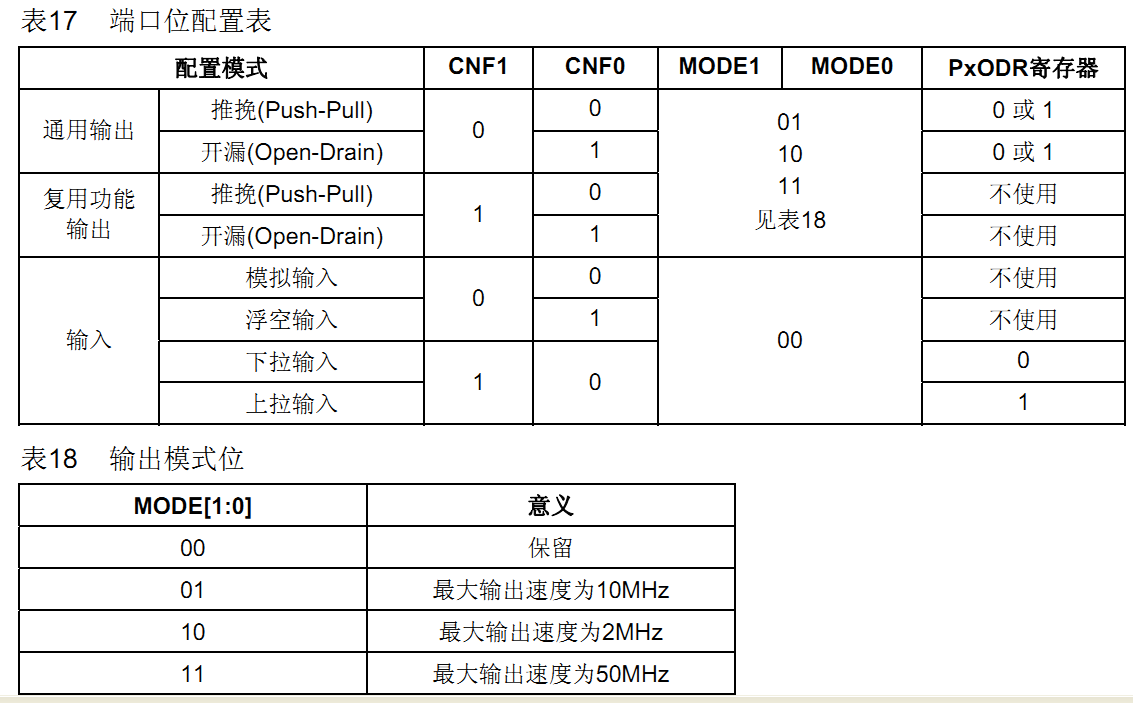
**端口配置高寄存器(GPIOx\_CRH)**



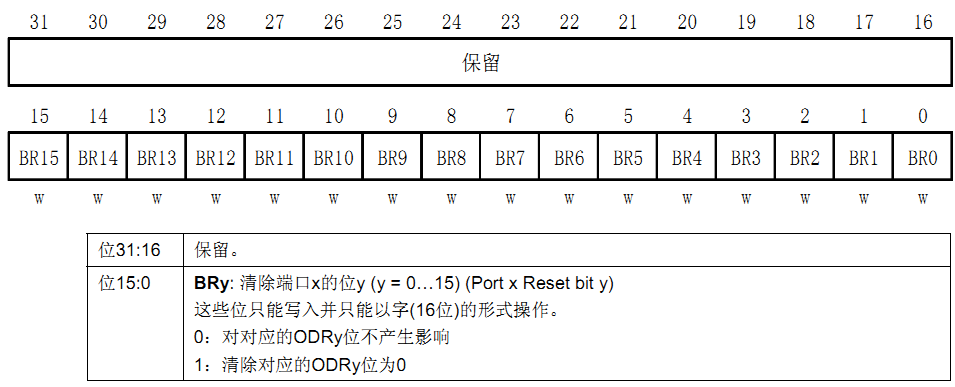
**端口输入数据寄存器(GPIOx\_IDR)**



**端口输出数据寄存器(GPIOx\_ODR)**

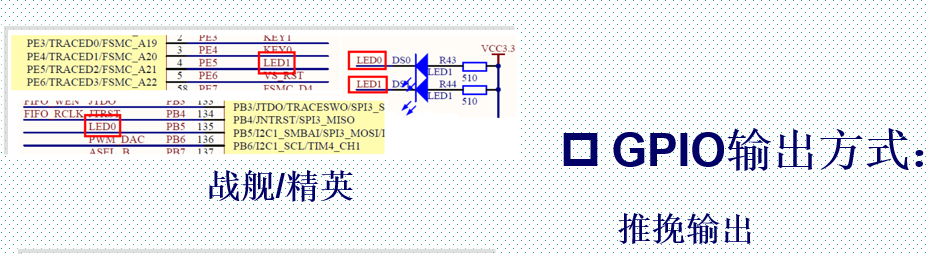
 

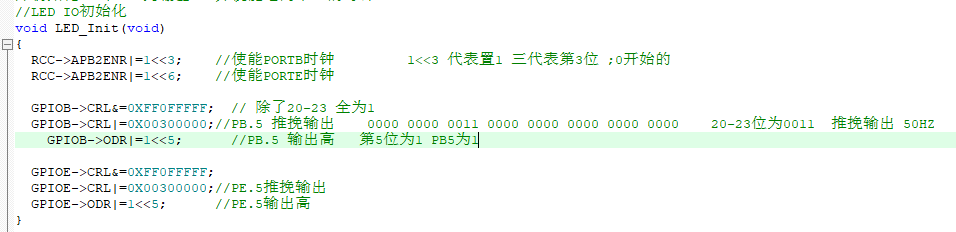
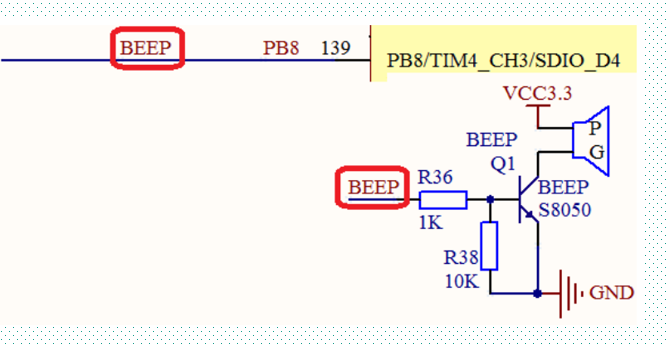
**端口位清除寄存器(GPIOx\_BRR)**

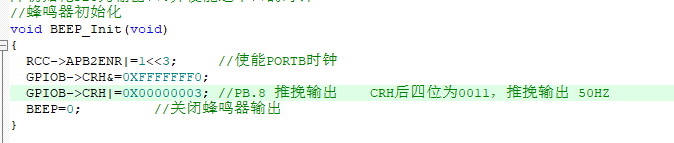


LED 例程：（上拉就是接电阻接高电平,下拉就是接个电阻接低电平;）

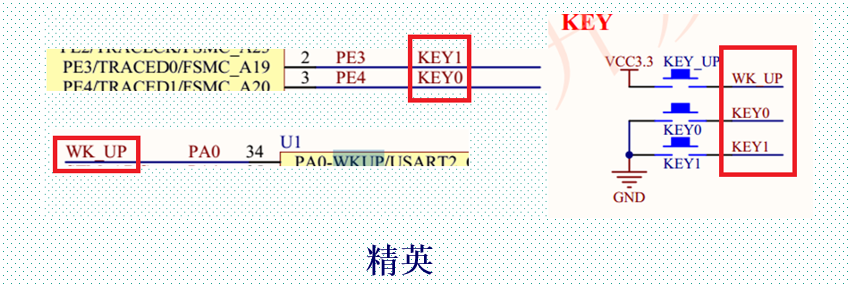
1. PGIO输出

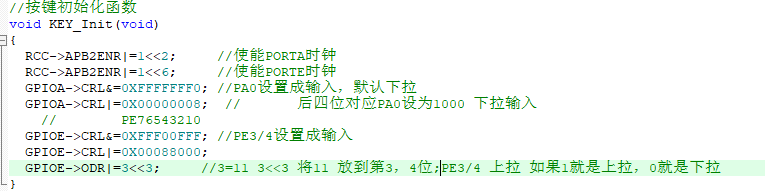
io=0 灯就亮

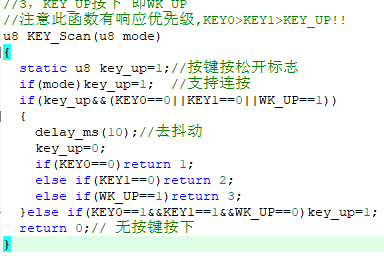
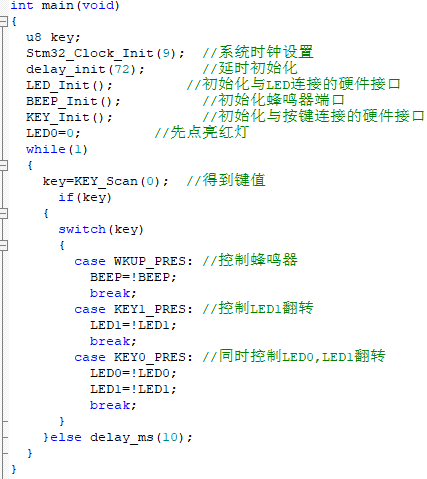
 BEEP=0 关闭蜂鸣器



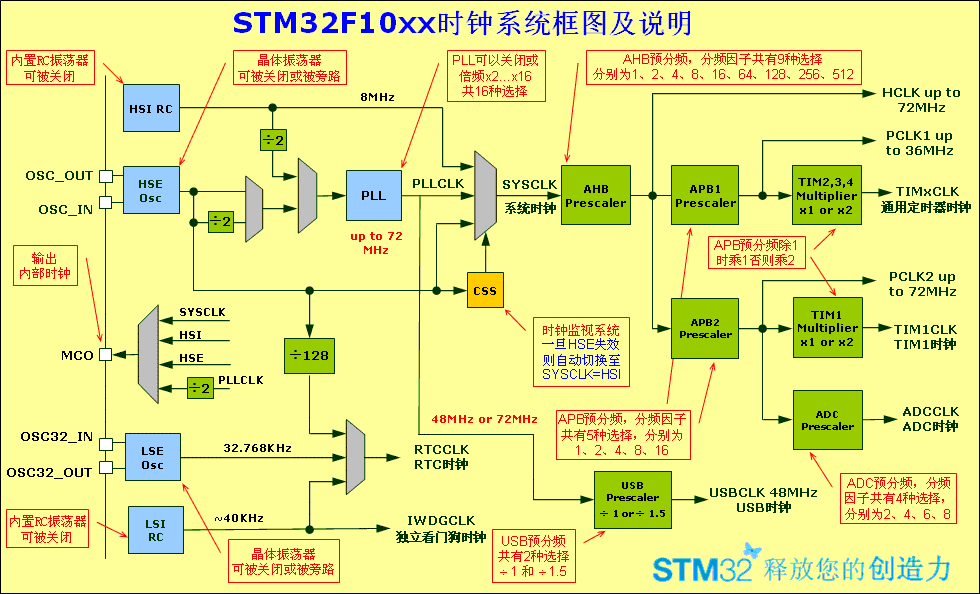
1. GPIO 输入

PA0为下拉电阻接vcc ，PE34为上拉电阻接VCC ，在板子内部。



# 时钟



## 时钟源：

STM32 有5个时钟源:**HSI、HSE、LSI、LSE、PLL。（图中蓝色部分）**

　 ①、HSI是高速内部时钟，RC振荡器，频率为8MHz，精度不高。  
　　 ②、HSE是高速外部时钟，可接石英/陶瓷谐振器，或者接外部时

钟源，频率范围为4MHz~16MHz。  
　　 ③、LSI是低速内部时钟，RC振荡器，频率为40kHz，提供低功耗时钟。WDG  
　　 ④、LSE是低速外部时钟，接频率为32.768kHz的石英晶体。RTC  
　　 ⑤、PLL为锁相环倍频输出，其时钟输入源可选择为HSI/2、HSE或者HSE/2。

倍频可选择为2~16倍，但是其输出频率最大不得超过72MHz。  
2. 系统时钟SYSCLK可来源于三个时钟源：  
 ①、HSI振荡器时钟

②、HSE振荡器时钟

③、PLL时钟

3.STM32可以选择一个时钟信号输出到MCO脚(PA8)上，可以选择为PLL

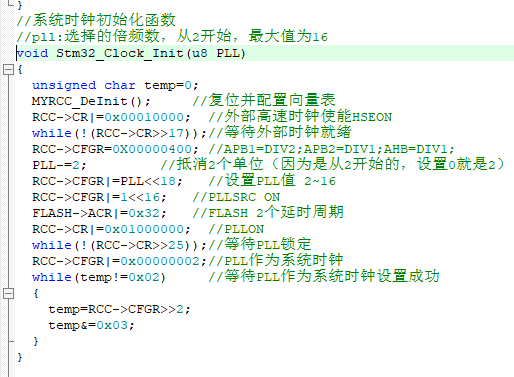
输出的2分频、HSI、HSE、或者系统时钟。

4.任何一个外设在使用之前，必须首先使能其相应的时钟。

* 几个重要的时钟：
  + SYSCLK(系统时钟) :
  + AHB总线时钟
  + APB1总线时钟(低速): 速度最高36MHz
  + APB2总线时钟(高速): 速度最高72MHz
  + PLL时钟

系统时钟初始化函数：

 **使用SystemInit 设置**



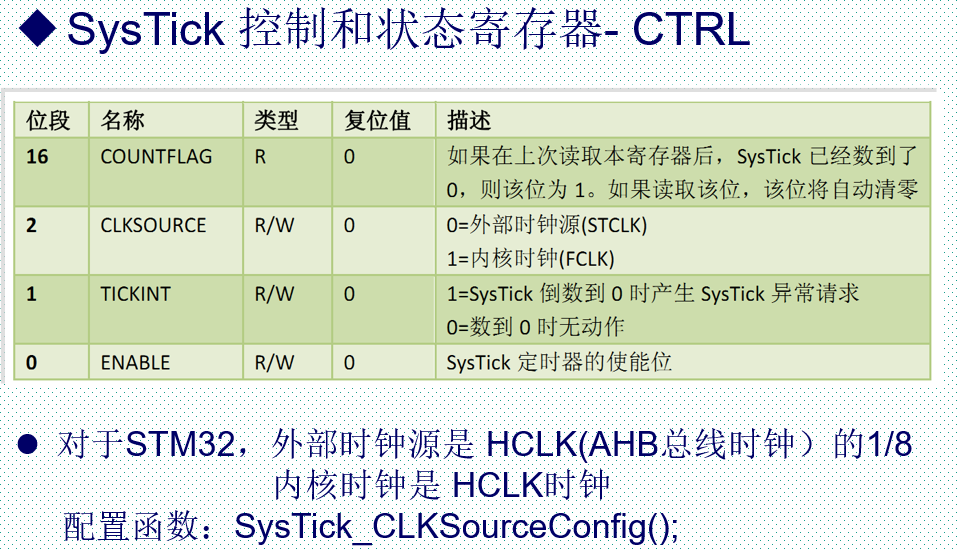
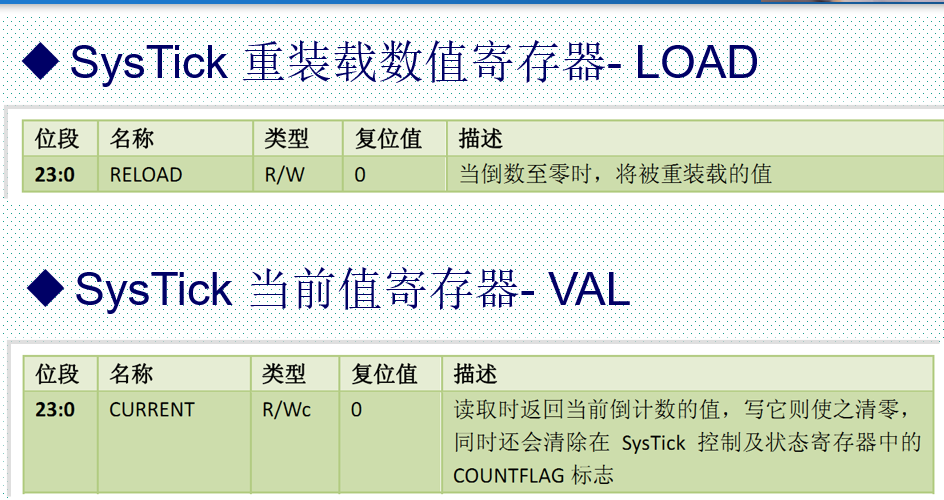
## 滴答定时器

**Systick定时器常用来做延时：一个24 位的倒计数定时器，计到0 时，将从RELOAD 寄存器中自动重装载定时初值。只要不把它在SysTick 控制及状态寄存器中的使能位清除，就永不停息，即使在睡眠模式下也能工作。**

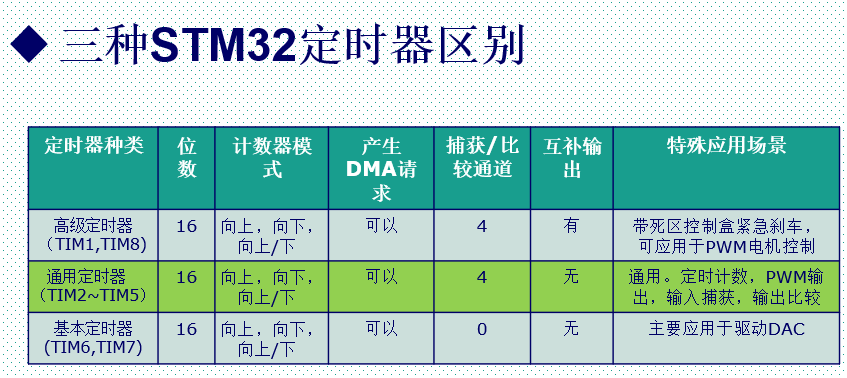
*CTRL            SysTick 控制和状态寄存器  LOAD*

*SysTick 自动重装载除值寄存器   
VAL              SysTick 当前值寄存器  CALIB*

*SysTick 校准值寄存器*

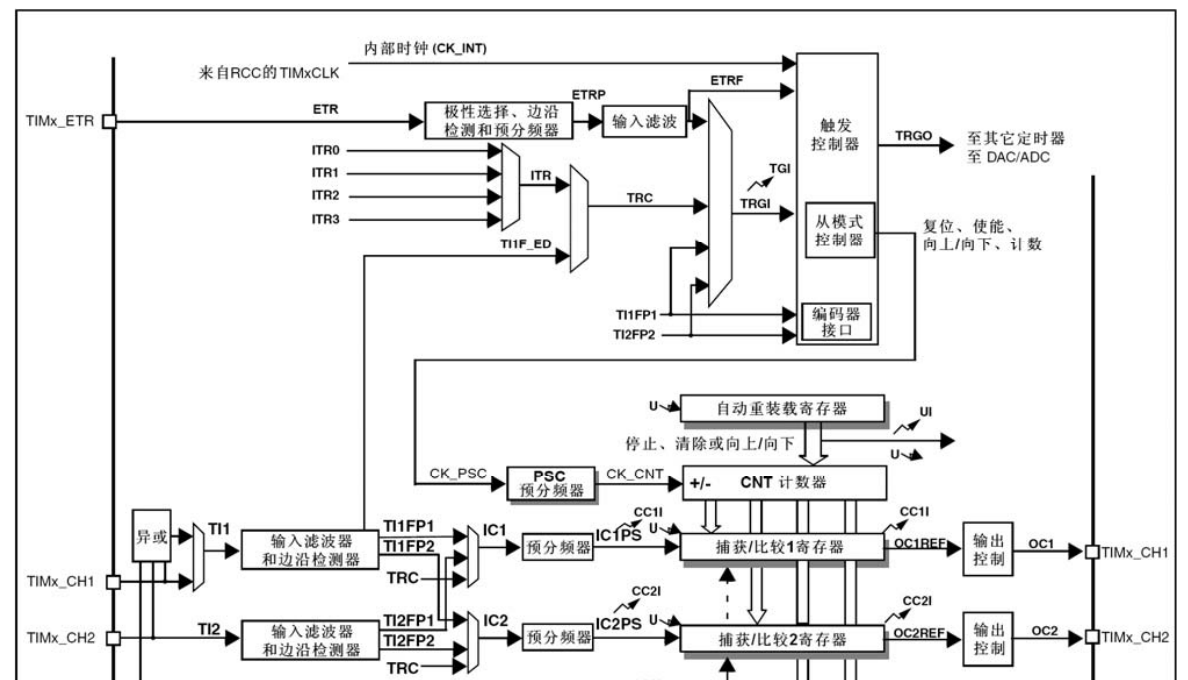
## 通用定时器TIM

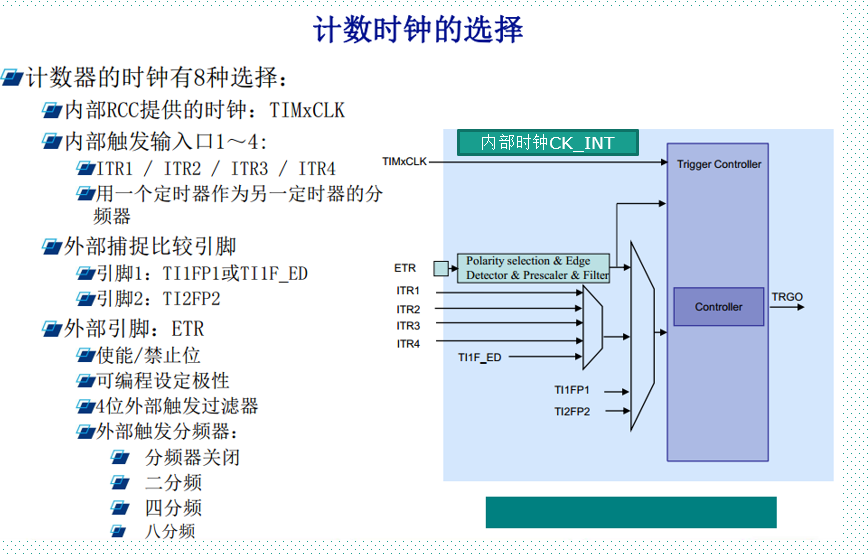


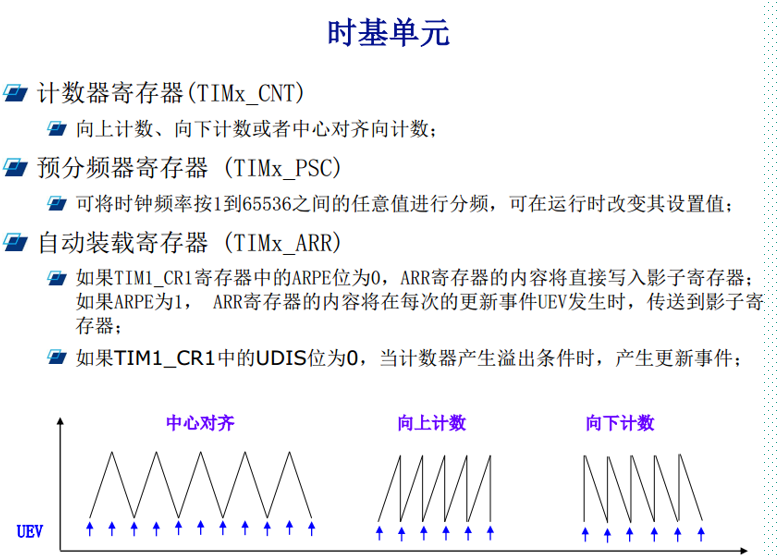
STM3 的通用 TIMx (TIM2、TIM3、TIM4 和 TIM5)定时器功能特点包括：

* 位于低速的APB1总线上(APB1)
* 16 位向上、向下、向上/向下(中心对齐)计数模式，自动装载计数器（TIMx\_CNT）。
* 16 位可编程(可以实时修改)预分频器(TIMx\_PSC)，计数器时钟频率的分频系数 为 1～65535 之间的任意数值。
* 4 个独立通道（TIMx\_CH1~4），这些通道可以用来作为：
  1. 输入捕获
  2. 输出比较
  3. PWM 生成(边缘或中间对齐模式)
  4. 单脉冲模式输出
* 可使用外部信号（TIMx\_ETR）控制定时器和定时器互连（可以用 1 个定时器控制另外一个定时器）的同步电路。

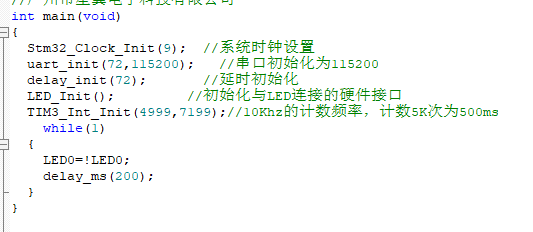
通用定时器工作过程：

****



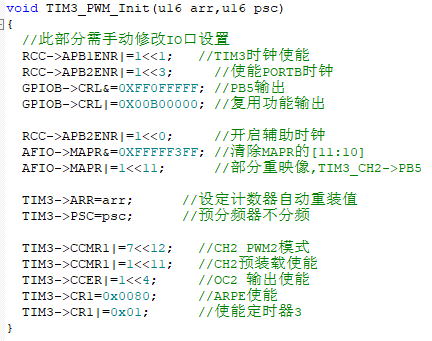


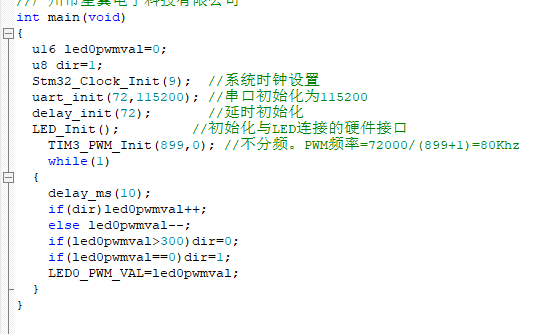
定时器中断





## PWM输出：





## 输入捕获输出比较