

摘要

对于初学者来说，STM32 系列单片机内容繁杂，难以为继，尤其是 51 都没有搞清楚的小朋友。这是一个致命的问题。首先，要搞清楚 STM32 由哪些部分构成，都有什么作用，有哪些应用？对于软件、云服务、物联网、机器人，STM32 在这些行业中处于怎样的生态位，起到什么作用。

STM32F103 系列开发板 STM32F103 芯片、F103 板上资源、ARM Cortex-M3 架构相关知识（ARM 系列处理器的特点，中断触发，寄存器，总线结构）、Keil 编译器、STMCubeMX、Linux 下开源开发库 libopencm3、ST-Link/J-Link、DEBUG、UCOSII、RTOS、SRAM、FLASH、E2PROM、ARM 架构采用哈弗架构、改进哈弗架构、冯诺伊曼架构、看门狗（软件开门狗和硬件看门狗）、RTC 时钟、SD 卡、PWM 波（PWM 电机、DAC）、LED/LCD、触摸板库、HAL 库、CMSIS

STM32 内容：GPIO 八种模式、时钟树（四个时钟源，两个高速时钟源，两个低速时钟源，一个 PLL 时钟源，芯片和板上各有一套高低速时钟源）、NVIC 中断（四个特点）、Timer（三种定时器）、总线结构（AHB、APB1、APB2，APB 时钟频率是 AHB 的二分之一）、通信协议（UART、USART、USMART、I2C、SPI、1-wire、RS232、RS485、USB、2.4GHz、bt、esp8266、zigbee、lora、nb-iot）

1 点灯

通过配置 GPIO 来点亮小灯 STM32 开发环境搭建、GPIO 精讲、原理图

1.1 内容概述

1.1.1 GPIO 讲解

八种模式，四种输入，四种输出，GPIO 不能直接输出模拟信号，但是可以对模拟信号进行定时采样，如何保证采样过程中信号不丢失。

1.1.2 HAL 库

什么是 HAL 库？

1.1.3 delay 函数

1.2 流程概述

1.2.1 初始化

第一步：初始化 hal 库
第二步：设置时钟
第三步：初始化 delay 函数
第四步：初始化 LED(自行编写)

1.2.2 循环部分

使用 HAL 库设置两个引脚拉高或拉低，500ms 后交换状态

1.2.3 初始化 LED

1. 创建 GPIO 结构体
2. 开启时钟
3. 选择 GPIO 端口，设置 GPIO 口状态

2 按键输入

通过配置 GPIO 捕获按键按下产生的电信号 GPIO，NVIC，按键去抖

2.1 内容讲解

2.1.1 NVIC

ARM Cortex-M3 架构 NVIC

2.2 流程概述

2.2.1 初始化

2.2.2 时钟树

时钟树的组成，可以和 ARM 架构来一起讲，还能再扩展一点，把 ARM 中断也讲一下。

2.3 内容讲解

ARM 中断，现场保护，寄存器，时钟树

2.4 时钟树

AHB、APB1/APB2、sysclk、systick

3 流水灯

使用 Timer 来控制小灯

3.1 内容概述

点灯、NVIC、精讲时钟树、Timer(定时器) 精讲时钟树!

4 PWM 电机控制

使用 PWM 波控制电机

4.1 内容概述

4.2 流程概述

5 看门狗

5.1 模块介绍

IWDG、WWDG、timer

5.2 内容概述

5.2.1 看门狗讲解

看门狗就是闹钟，看门狗有一个按钮，你可以上设置 10s 之内必须按一下，否则，闹钟就会响，分为两种，一个是窗口看门狗，一个是独立看门狗
窗口看门狗，之所以称为窗口，是因为其喂狗时间是一个有上下限的范围内，你可以通过设定相关寄存器，设定其上限时间和下限时间：喂狗的时间不能过早也不能过晚。

独立看门狗由内部低速时钟，RC 振荡器提供时钟信号。

5.3 流程概述