STM32基础知识

基于B站教程的学习

LZY

2020

# STM32前置知识

——以STM32F103x8系列为例

## 1.内核与存储器

-以STM32F10x8系列，其内核为：ARM 32位的CortexM3- CPU，最高工作频率72MHz,在存储器的0等待周期访问时可达到1.25DMips/MHz。

-具备单周期乘法、硬件除法功能。

-64K或124K字节的Flash存储器

-20K字节的SRAM

红色代表优点，蓝色代表缺点

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 存储器 | 英文缩写 | 类型 | 储存内容 | 读写延迟 | 掉电数据状态 | 写入次数 | 写入条件 | 成本 | 容量 |
| 静态随机存储器 | SRAM | RAM | 运行中的数据 | 无延迟速度较快 | 消失 | 无限次 | 无限制 | 高 | 小 |
| 闪存存储器 | Flash | ROM | 用户程序 | 有延迟速度较慢 | 保存 | 10万次 | 擦除数据 | 低 | 大 |

硬件程序运行框图



## 2.时钟、复位和电源管理

- 2.0-3.6V供电和I/O引脚

- 上电/断电复位(POR/PDR)、可编程电压检测器(PVD)

- 4-16MHz晶体振荡器

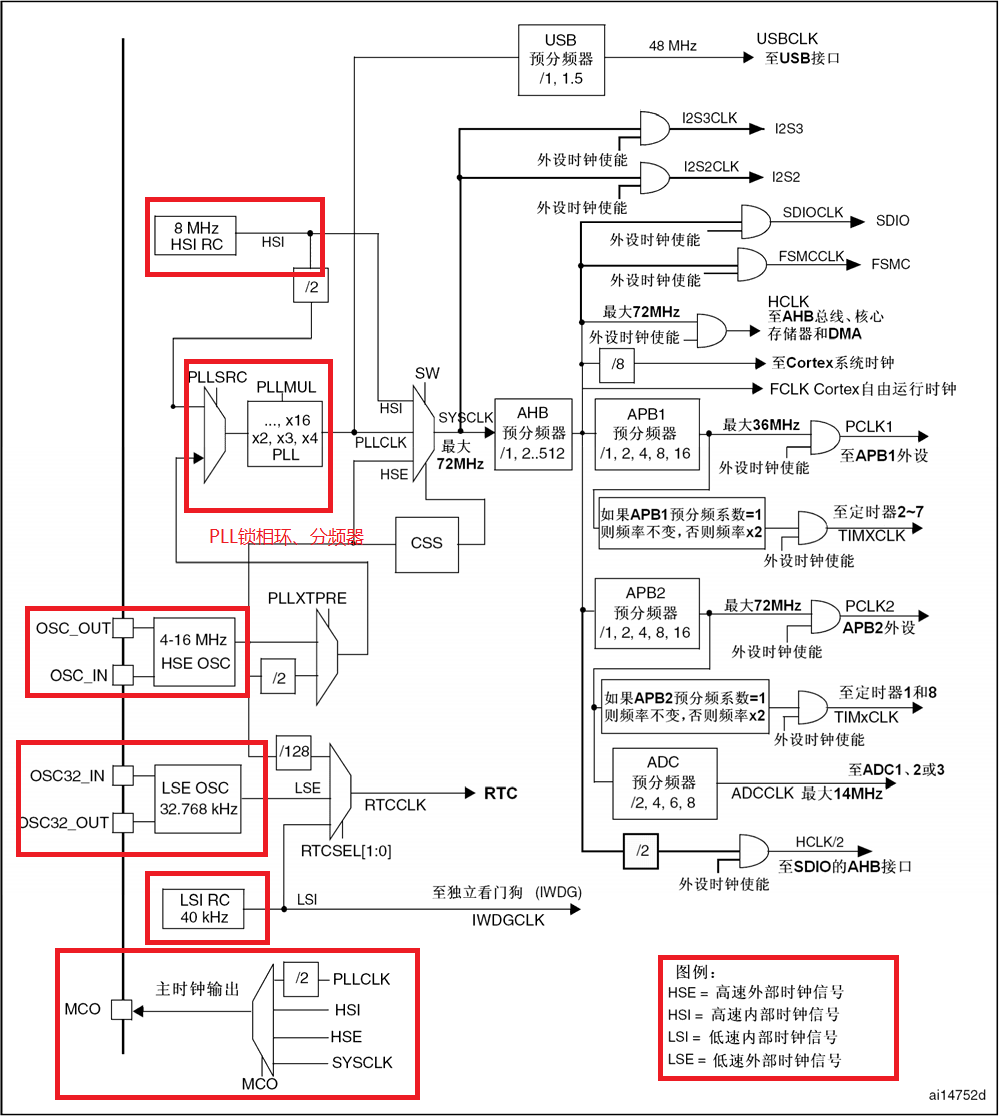
- 内部包含经出厂校准的8MHz的RC振荡器

- 带校准的40kHz的RC振荡器

- 产生CPU时钟的PLL(锁相环、分频器)——实现频率切分、倍频功能

- 带校准功能的32kHz RTC振荡器

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 名称 | 英文缩写 | 频率 | 外部连接 | 功能 | 用途 | 特性 |
| 外部高速晶体振荡器 | HSE | 4/16MHz | 4/16MHz晶体 |  | 系统时钟/RTC | 成本高温度漂移小 |
| 外部低速晶体振荡器 | LSE | 32kHz | 32.768kHz晶体 | 校准功能 | RTC | 成本高温度漂移小 |
| 内部高速RC振荡器 | HSI | 8MHz | 无 | 出场调校 | 系统时钟 | 成本低，温度漂移大 |
| 内部低速RC振荡器 | LSI | 40kHz | 无 | 校准功能 | RTC | 成本低，温度漂移大 |

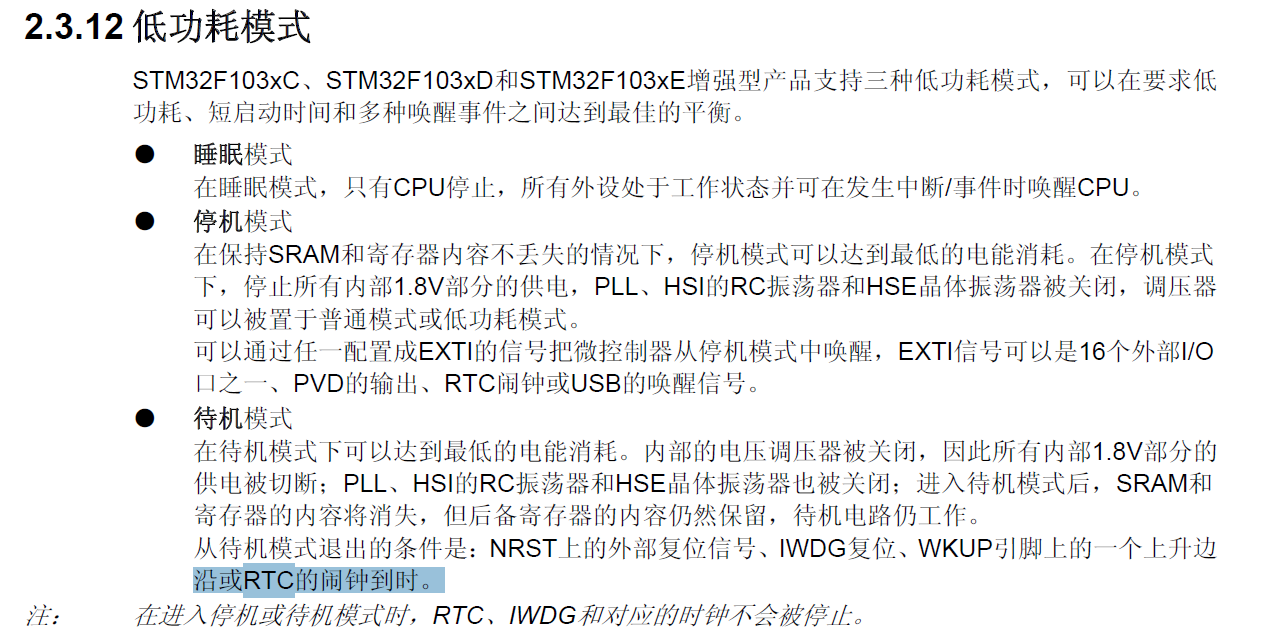


## 3.低功耗和ADC

### 低功耗

- 睡眠、停机和待机模式

- VBAT(一般为纽扣电池供电电压)为RTC和后备寄存器供电



根据数据手册中的描述，做如下总结(低功耗模式)

|  |  |  |
| --- | --- | --- |
| 工作模式 | 关闭功能 | 唤醒方式 |
| 睡眠模式 | ARM内核 | 所有内部/外部功能的中断/事件 |
| 停机模式 | ARM内核 内部所有功能 PLL分频器、HSE | 外部中断输入接口EXTI（16个I/O接口之一） 电源电压检测中断PVD RTC时钟计时结束（闹钟） USB唤醒信号 |
| 待机模式StandBy | ARM内核 内部所有功能 PLL分频器、HSE SRAM数据被清空 | NSRT引脚外部信号 独立看门狗IWDG复位 专用WakeUp引脚 RTC时钟到时（闹钟） |

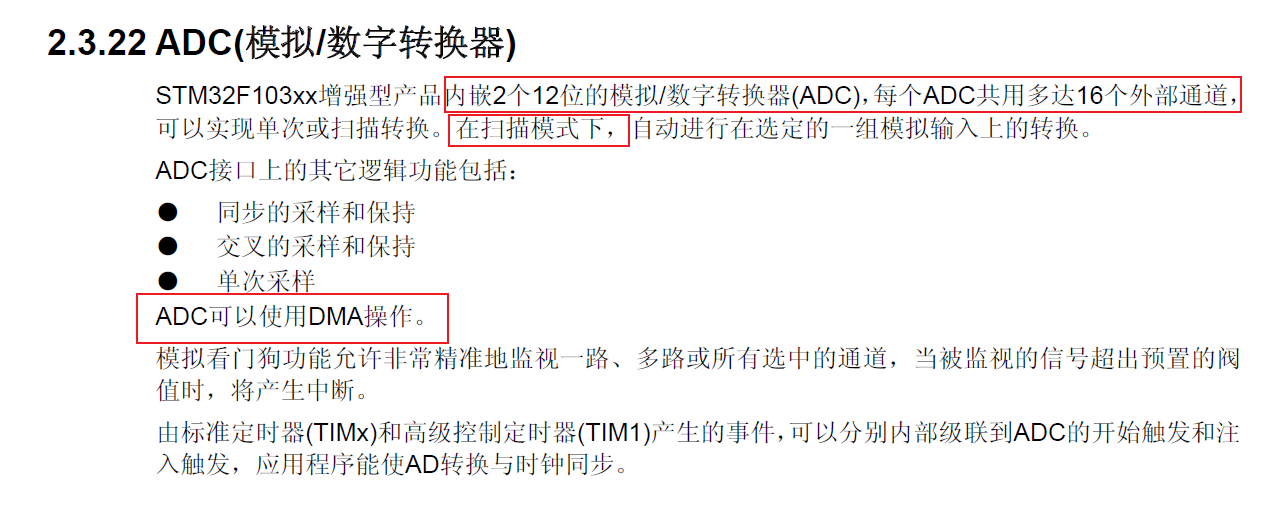
### ADC(Analog to digital converter,模拟数字转换器)

2个12bit，1μs A/D转换器（具有16个输入通道）

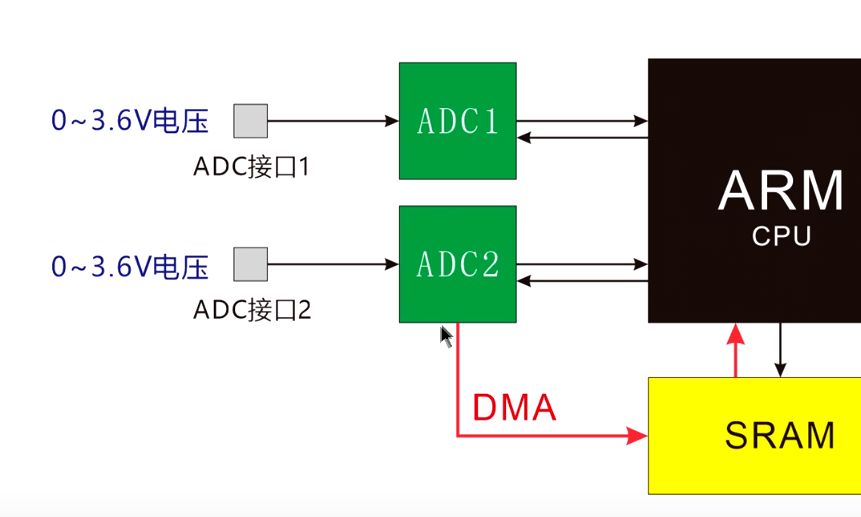
转换范围：0-3.6V

双采样和保持功能

具有内部温度传感器



ADC使用DMA操作的简要示意图



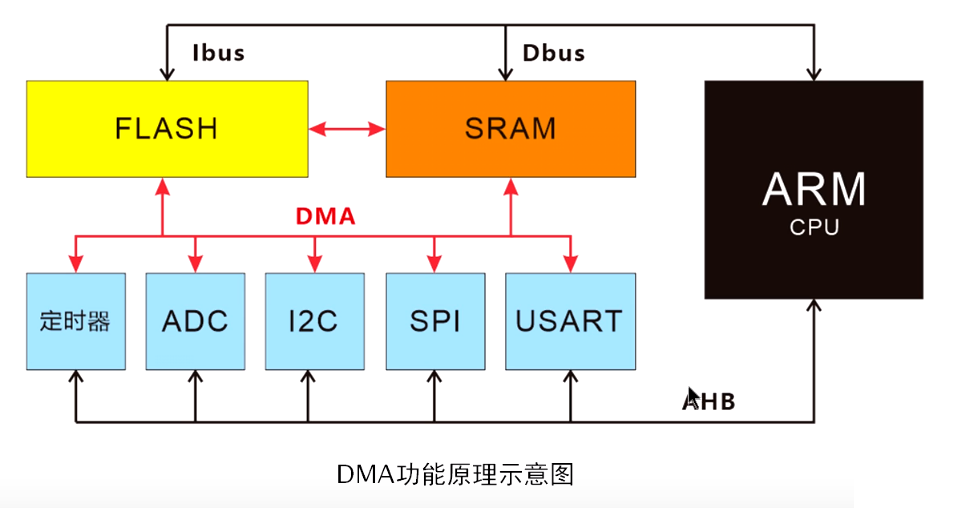
## 4.DMA和I/O端口

### DMA(Direct Memory Access，直接存储器访问)

灵活的7路通用DMA可以管理存储器到存储器、设备到存储器和存储器到设备的数据传输；DMA控制器支持环形缓冲区的管理，避免了控制器传输到达缓冲区结尾时所产生的中断。

每个通道都有专门的硬件DMA请求逻辑，同时可以由软件触发每个通道；传输的长度、传输的源地址和目标地址都可以通过软件单独设置。

DMA可以用于主要的外设：SPI、I2C、USART，通用、基本和高级控制定时器TIMx和ADC。



### GPIO(General-purpose input/output，通用输入输出)

26/37/51/80个I/O口，所有I/O口可以映像到16个外部中断；几乎所有端口均可容忍5V信号

每个GPIO引脚都可以由软件配置成输出(推挽或开漏)、输入(带或不带上拉或下拉)或复用的外设功能端口。多数GPIO引脚都与数字或模拟的复用外设共用。

I/O引脚的外设功能可以通过操作锁定，以避免意外的写入I/O寄存器。在APB2（内部通信总线）上的I/O脚可达18MHz的翻转速度。

#### GPIO输入模式

浮空输入——Input floating

上拉输入——Input pull-up

下拉输入——Input-pull-down

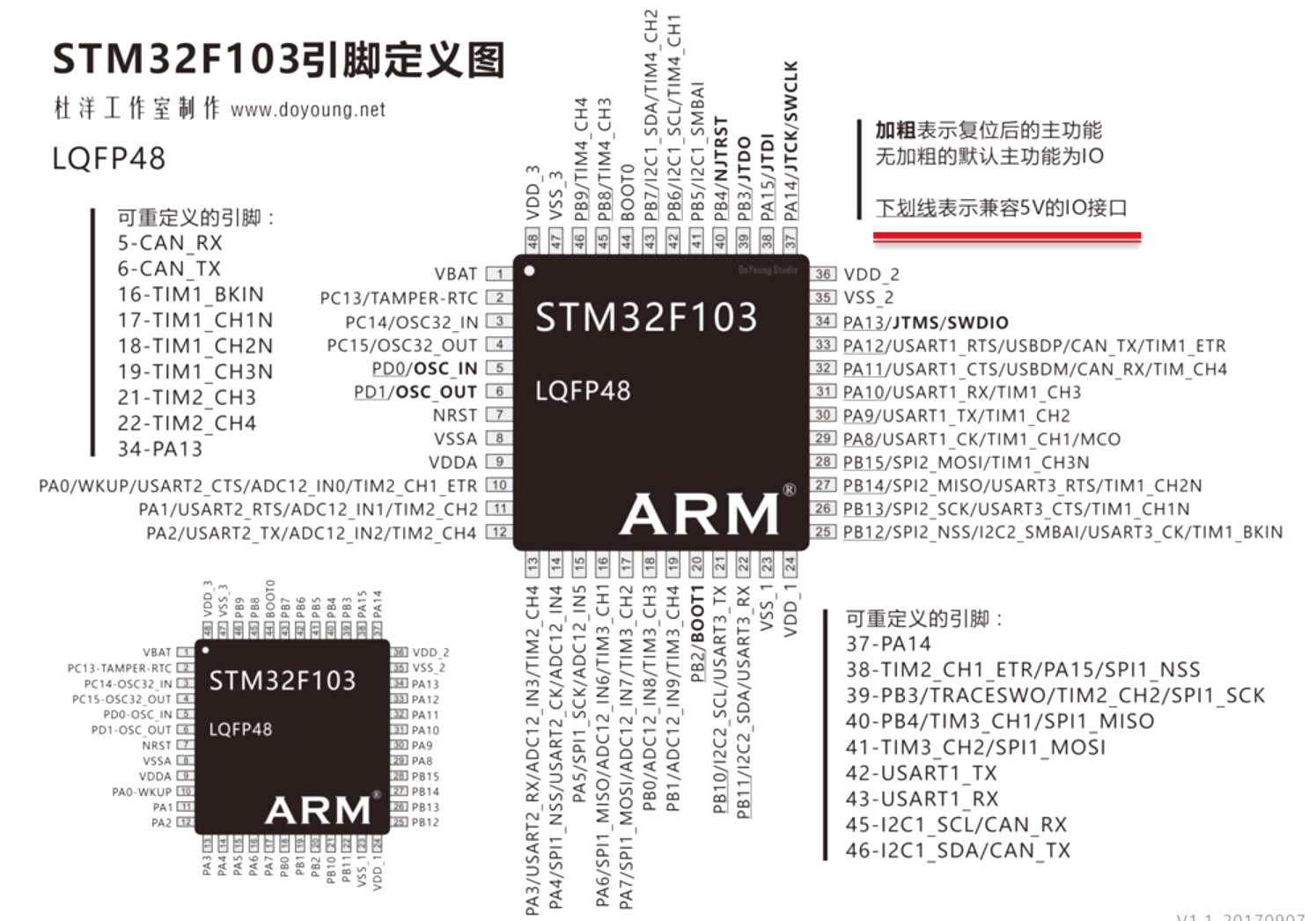
模拟输入——Analog

开漏输出——Output open-drain

推挽输出——Output push-pull

复用推挽输出——Alternate function push-pull

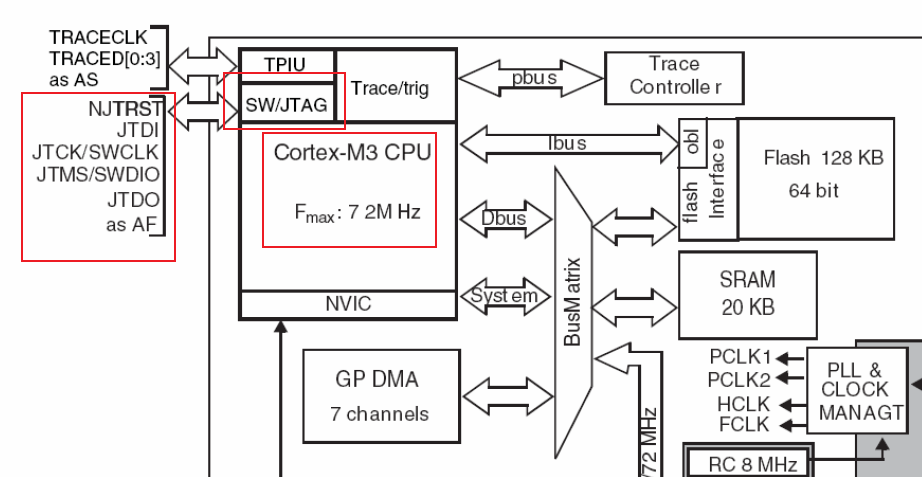
复用开漏输出——Alternate function open-drain

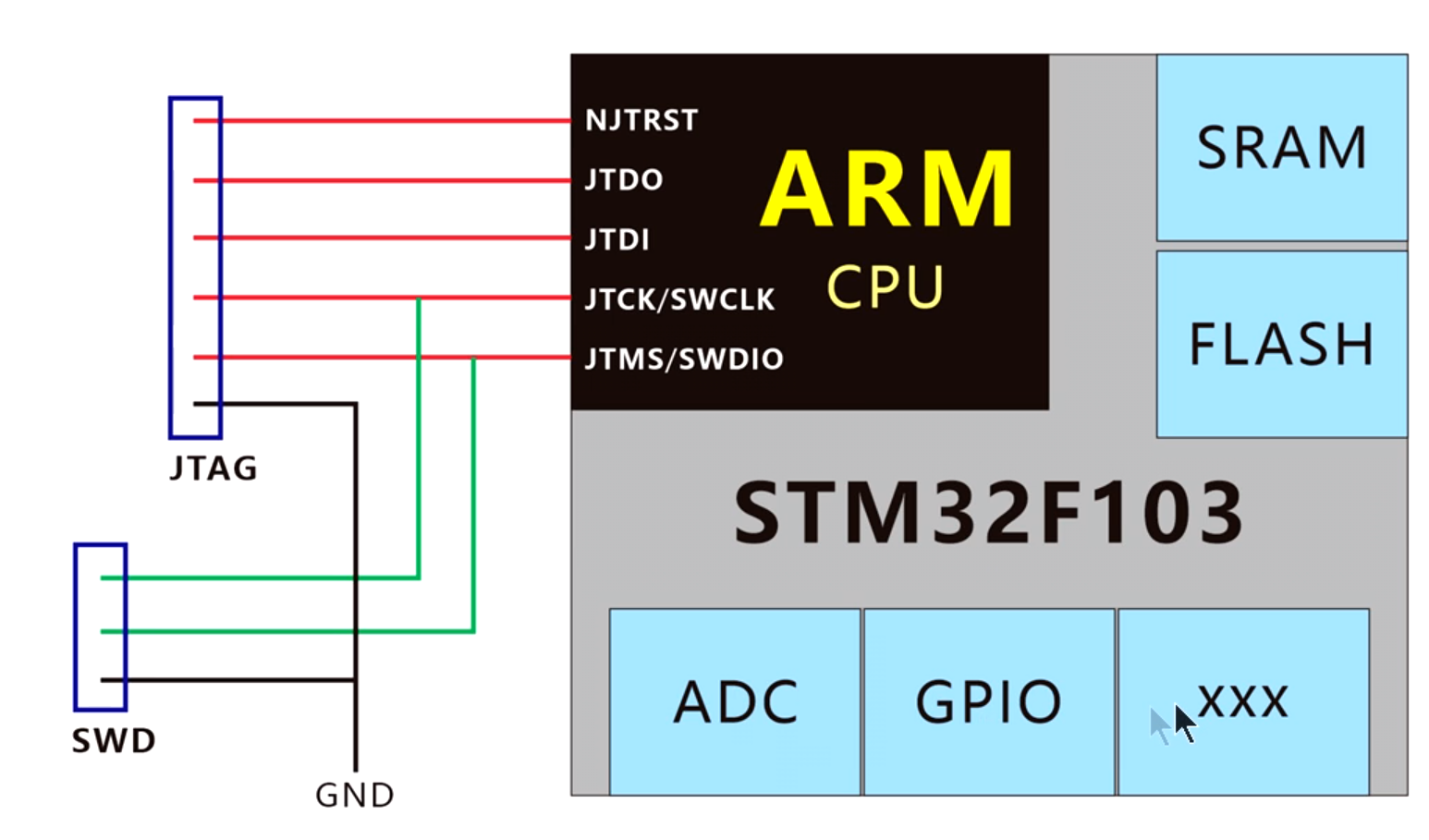


## 5.调试模式和定时器

### 调试模式

内嵌ARM的SWJ-DP接口，这是一个结合了JTAG和串行单线调试的接口，可以实现串行单线调试接口或JTAG接口的连接。JTAG的TMS和TCK信号分别与SWDIO和SWCLK共用引脚，TMS脚上的一个特殊的信号序列用于在JTAG-DP和SW-DP间切换。





### 定时器

3个16位普通定时器，每个定时器有多达4个用于输入捕获/输出比较/PWM或脉冲计数的通道和增量编码器输入

1个16位带死区控制和紧急刹车，用于电机控制的PWM高级控制定时器

2个看门狗定时器(独立的和窗口型的)

系统时间定时器：24位自减型计数器

#### 通用定时器(TIMx)

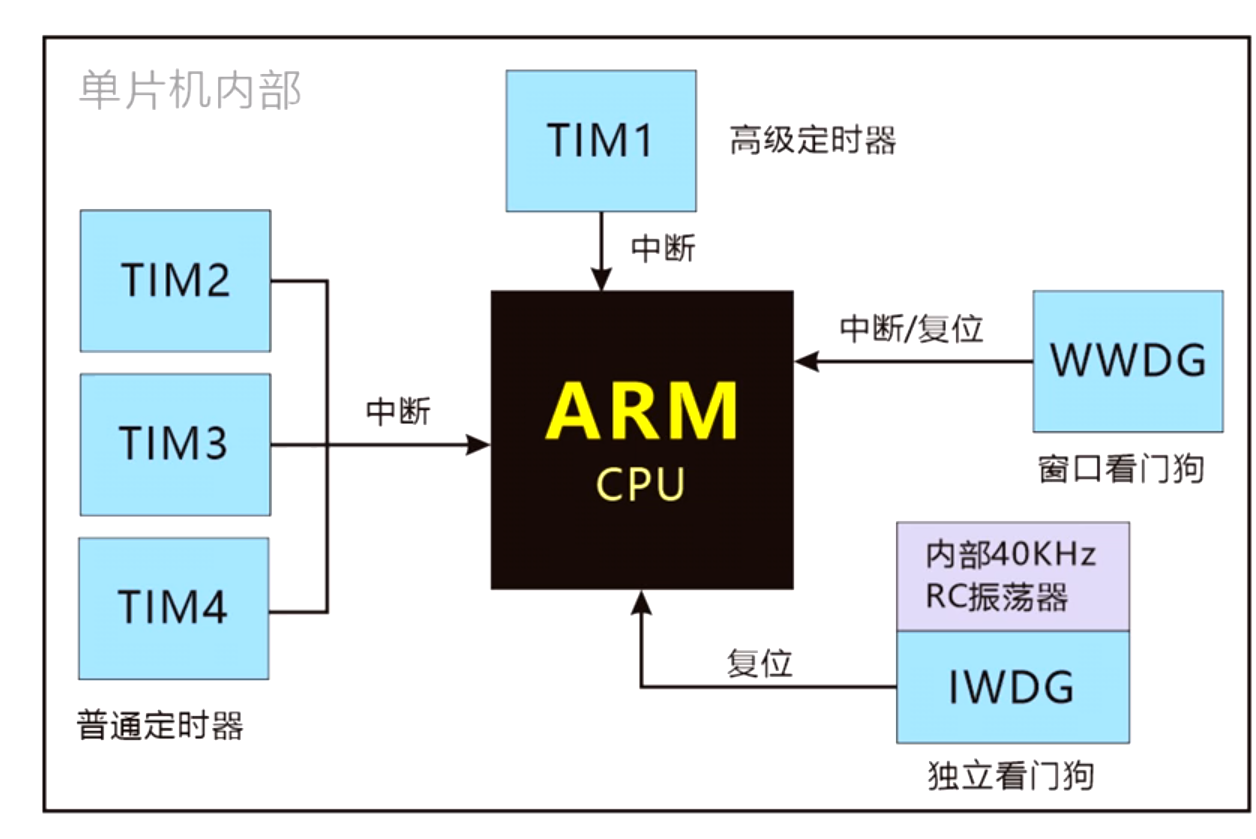
STM32F103xx增强型产品中，内置了多达3个可同步运行的标准定时器(TIM2、TIM3和TIM4)。每个定时器都有一个16位的自动加载递加/递减计数器、一个16位的预分频器和4个独立的通道，每个通道都可用于输入捕获、输出比较、PWM和单脉冲模式输出，在最大的封装配置中可提供最多12个输入捕获、输出比较或PWM通道。

它们还能通过定时器链接功能与高级控制定时器共同工作。在调试模式下，计数器可以被冻结。任意标准定时器都能用于产生PWM输出。每个定时器都有独立的DMA请求机制。这些定时器还能够处理增量编码器的信号，也能处理1至3个霍尔传感器的数字输出。

#### 高级控制定时器(TIM1)

高级控制定时器(TIM1)可以被看成是分配到6个通道的三相PWM发生器，它具有带死区插入的互补PWM输出，还可以被当成完整的通用定时器。四个独立的通道可以用于：

输入捕获、输出比较、产生PWM(边缘或中心对齐模式)、单脉冲输出



## 6.多种类通信接口

多达2个I2C接口(支持SMBus/PMBus)

多达3个USART接口(支持ISO7816接口，LIN，IrDA接口和调制解调控制)

多达2个SPI接口(18M位/秒)

CAN接口(2.0B 主动)

USB 2.0全速接口

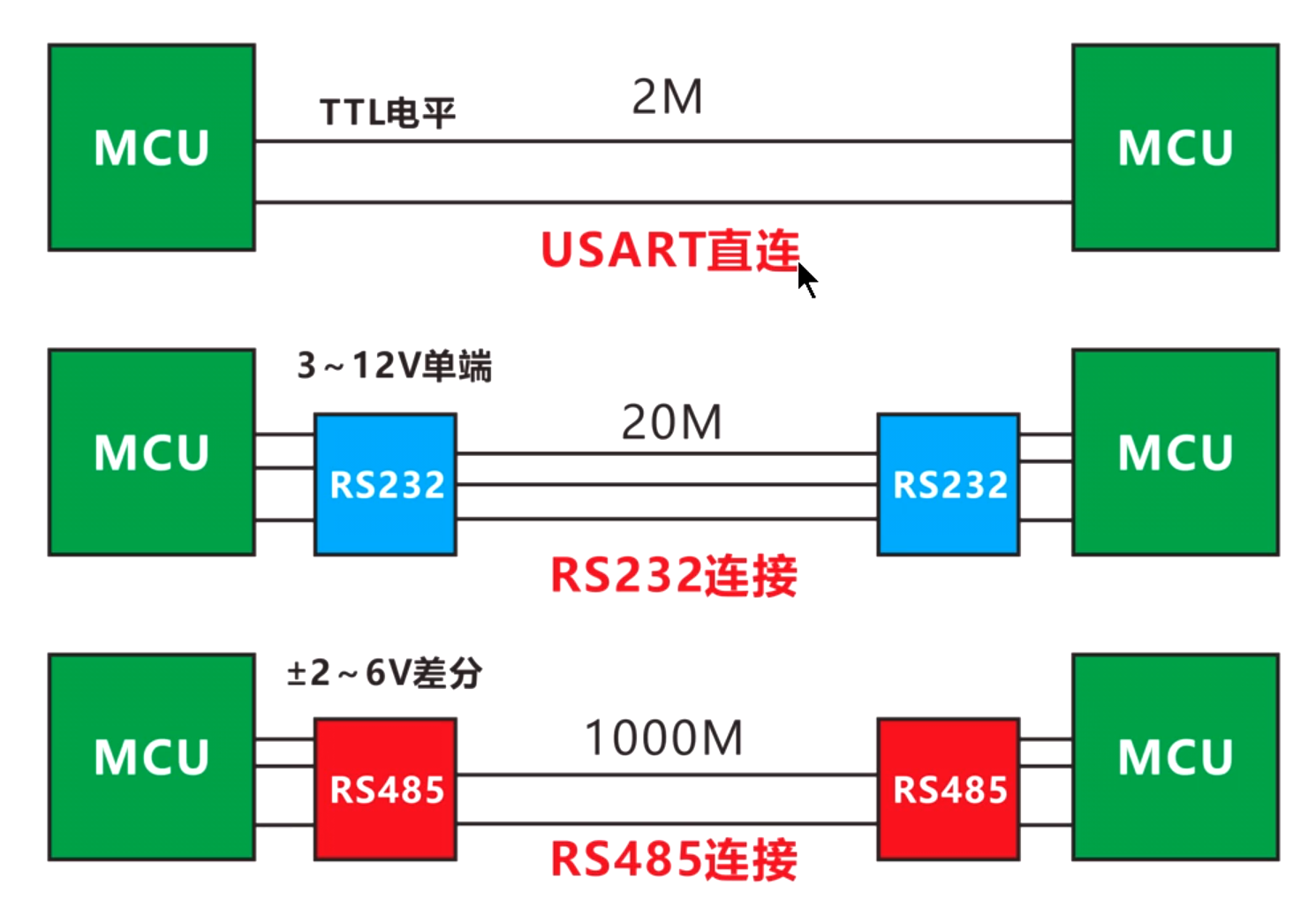
### I2C总线

多达2个I2C总线接口，能够工作于多主模式或从模式，支持标准和快速模式。I2C接口支持7位或10位寻址，7位从模式时支持双从地址寻址。内置了硬件CRC发生器/校验器。它们可以使用DMA操作并支持SMBus总线2.0版/PMBus总线。



### 通用同步/异步收发器(USART)

USART1接口通信速率可达4.5兆位/秒，其他接口的通信速率可达2.25兆位/秒。USART接口具有硬件的CTS和RTS信号管理、支持IrDA SIR ENDEC传输编解码、兼容ISO7816的智能卡并提供LIN主/从功能。所有USART接口都可以使用DMA操作。



### 串行外设接口(SPI)

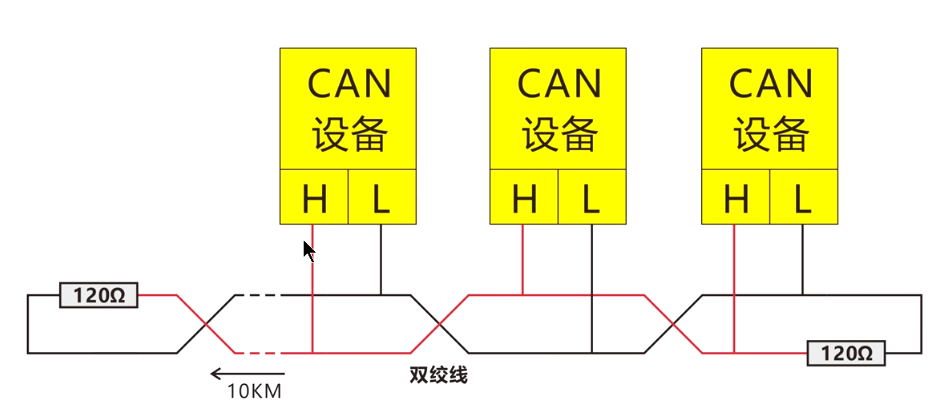
多达2个SPI接口，在从或主模式下，全双工和半双工的通信速率可达18兆位/秒。3位的预分频器可

产生8种主模式频率，可配置成每帧8位或16位。硬件的CRC产生/校验支持基本的SD卡和MMC模式。所有的SPI接口都可以使用DMA操作。



### 控制器区域网络(CAN)

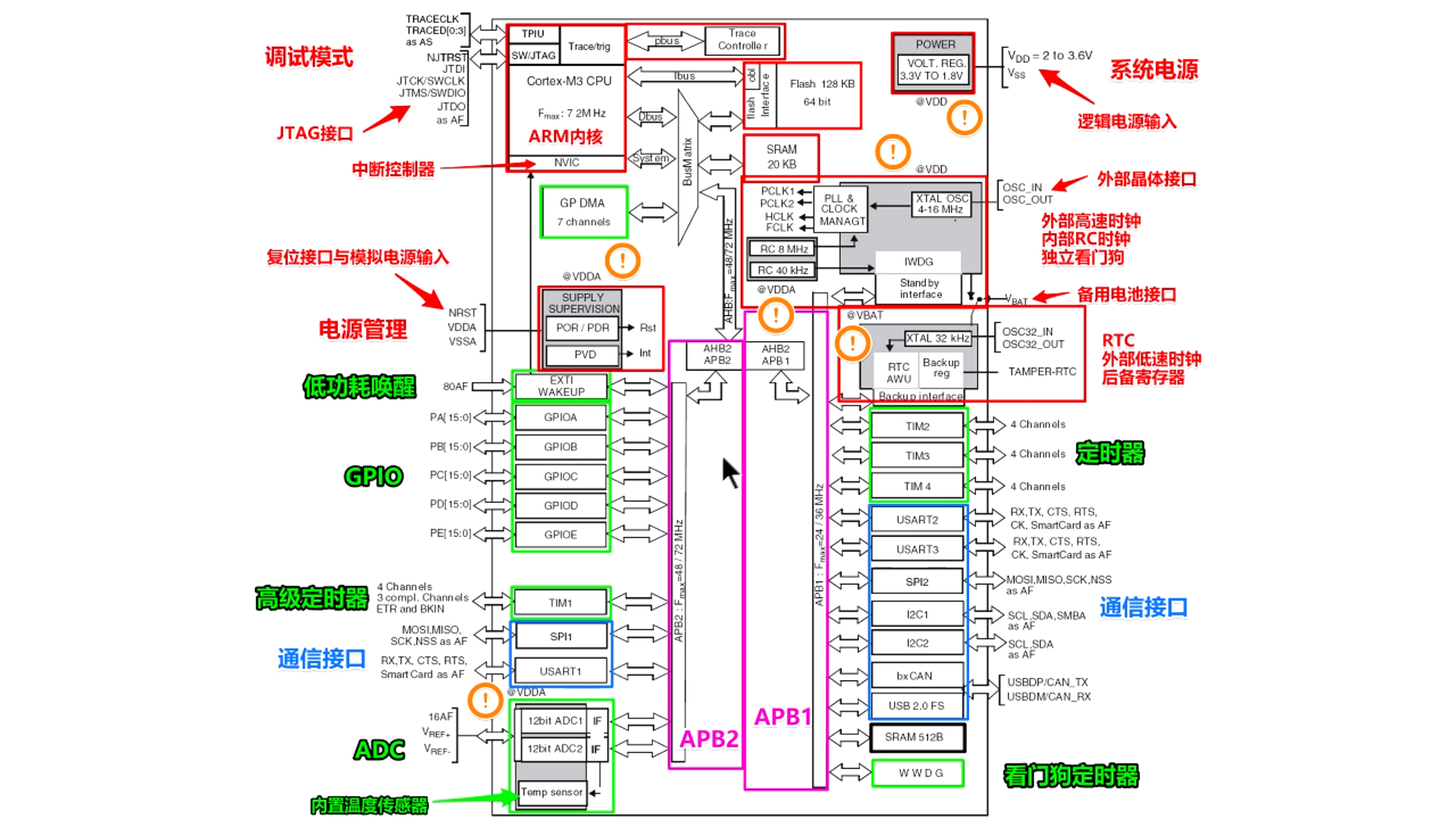
CAN接口兼容规范2.0A和2.0B(主动)，位速率高达1兆位/秒。它可以接收和发送11位标识符的标准帧，也可以接收和发送29位标识符的扩展帧。具有3个发送邮箱和2个接收FIFO，3级14个可调节的滤波器。



### 通用串行总线(USB)

STM32F103xx增强型系列产品，内嵌一个兼容全速USB的设备控制器，遵循全速USB设备(12兆位/秒)标准，端点可由软件配置，具有待机/唤醒功能。USB专用的48MHz时钟由内部主PLL直接产生(时钟源必须是一个HSE晶体振荡器)。

## 7.内部功能回顾与总结



## 8.引脚接口定义



