

《iBoard 电子学堂》第六讲 STM32 微处理器及接口技术（上）

关于 《iBoard 电子学堂》



《iBoard 电子学堂》是一个综合型的电子研发开发平台，适合在校学生、一线工程师及电子爱好者等。

交流方式：

官方博客：XiaomaGee.cnblogs.com

官方论坛：www.oshcn.com

官方淘宝店铺：i-Board.taobao.com

QQ群：

《iBoard 电子学堂 群【A】》：204255896（500人，已满）

《iBoard 电子学堂 群【B】》：165201798（500人，已满）

《iBoard 电子学堂 群【C】》：215053598（200人高级群）

《iBoard 电子学堂 群【D】》：215054675（200人高级群）

《iBoard 电子学堂 群【E】》：215055211（200人高级群）

《iBoard 电子学堂 群【F】》：78538605（200人高级群）

《iBoard 电子学堂 群【G】》：158560047（500人高级群）

王紫豪-XiaomaGee(15959622) 20:26:19

大家好，我是 XiaomaGee，这是我的第六次群课了，感谢大家对我的支持！

由于是四个群直播，可能稍微有点慢，请大家谅解！

IC 爱好者(446136066) 20:27:12



王紫豪-XiaomaGee(15959622) 20:27:48

人数较多，为了达到良好的效果；请大家不要发言，课间会留出来问问题的时间。

另外为了达到更好地观看效果；请大家把 QQ 群窗口最大化。

王紫豪-XiaomaGee(15959622) 20:29:13

我们上几次群课，分别讲解了关于电源系统、任意波发生器、及示波器电路的设计思想及电路

但是由于网友朋友们分布在天南海北；应用领域也不尽相同；所以这节课就安排一节入门级的课程，名字叫做《STM32 微处理器及接口技术》

王紫豪-XiaomaGee(15959622) 20:31:01

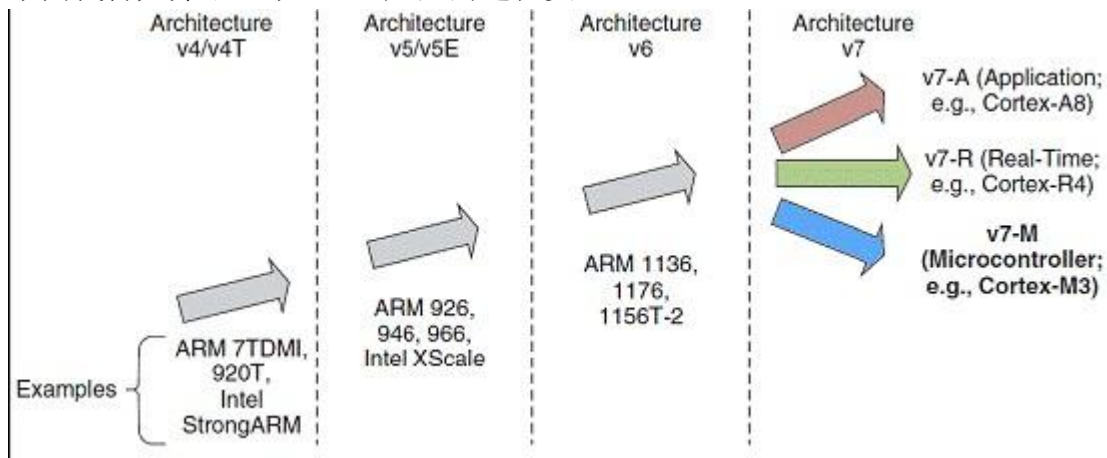
也是为了照顾不同层次的网友们。本人能力有限，说不到位的请大家谅解！

下面切入正题

stm32 这几个字母，对于经常上网的开发者或者奋斗在开发前线的研发人员一定不会太陌生；这货这两年太火了。

王紫豪-XiaomaGee(15959622) 20:33:18

下面我首先介绍一下，arm 产品的进化史



王紫豪-XiaomaGee(15959622) 20:35:16

从上图我们可以看出，从产品的角度来说， arm7, arm9, arm11, cortex 内核是 arm 内核版本不断进化的结果。。

王紫豪-XiaomaGee(15959622) 20:36:18

很多初学者问过我，要学 arm，学嵌入式，我向他们推荐 cortex，他们说学的是 arm，不想学 cortex；这个，其实是一回事哦。

n_i_o_s(534625052) 20:36:45

呵呵

王紫豪-XiaomaGee(15959622) 20:37:02

其实，从内核的先进性来说，cortex 内核已经是最先进的了。

当然，先进性不一定代表性能就比前期的产品高。这个是两码事的。

王紫豪-XiaomaGee(15959622) 20:38:23

热心的网友对 cortex-M3 内核跟他上一代类似应用领域的 ARM7，做了一个对比，我截个图给大家看看。

| 比较项目 | ARM7 | Cortex-M3 |
|--------|---------------------------------------|--|
| 架构 | ARMv4T（冯诺依曼） 指令和数据总线共用，会出现瓶颈 | ARMv7-M（哈佛） 指令和数据总线分开，无瓶颈 |
| 指令集 | 32位ARM指令+16位Thumb指令 两套指令之间需要进行状态切换 | Thumb/Thumb-2指令集 16位和32位 指令可直接混写，无需状态切换 |
| 流水线 | 3级流水线 若出现转移则需要刷新流水线，损失惨重 | 3级流水线+分支预测 出现转移时流水线 无需刷新，几乎无损失 |
| 性能 | 0.95DMIPS/MHz（ARM模式） | 1.25DMIPS/MHz |
| 功耗 | 0.28mW/MHz | 0.19mW/MHz |
| 低功耗模式 | 无 | 内置睡眠模式 |
| 面积 | 0.62mm ² （仅内核） | 0.86mm ² （内核+外设） |
| 中断 | 普通中断IRQ和快速中断FIQ太少，大量 外设不得不复用中断 | 不可屏蔽中断NMI+1-240个物理中断 每个外设都可以独占一个中断，效率高 |
| 中断延迟 | 24-42个时钟周期，缓慢 | 12个时钟周期，最快只需6个 |
| 中断压栈 | 软件手工压栈，代码长且效率低 | 硬件自动压栈，无需代码且效率高 |
| 存储器保护 | 无 | 8段存储器保护单元（MPU） |
| 内核寄存器 | 寄存器分为多组、结构复杂、占核面积 多 | 寄存器不分组（SP除外），结构简单 |
| 工作模式 | 7种工作模式，比较复杂 | 只有线程模式和处理模式两种，简单 |
| 乘除法指令 | 多周期乘法指令，无除法指令 | 单周期乘法指令，2-12周期除法指令 |
| 位操作 | 无 访问外设寄存器需分“读-改-写”3步 走 | 先进的Bit-band位操作技术，可直接访问 外设寄存器的某个值 |
| 系统节拍定时 | 无 | 内置系统节拍定时器，有利于操作系统移植 |

王紫豪-XiaomaGee(15959622) 20:39:48

这个表只是比较了部分功能，当然也不能绝对的代表全部。但是从表中可以看出，cortex 内核的确“进化”了。

王紫豪-XiaomaGee(15959622) 20:41:34

arm7 时代的产品，我也使用过多个，从我个人主观感觉上，cortex-m3 内核的cpu 比 arm7 在外设、中断响应、代码密度等方面有较大的进步！

当然，在国内做产品最重要的一点就是成本，也降低了很多。

王紫豪-XiaomaGee(15959622) 20:43:21

就拿今天我们主讲的微处理器 STM32F103VCT6 这个来说，单片零售价格都能到 16 CNY ，真的不可思议。

王紫豪-XiaomaGee(15959622) 20:45:05

话题转回来，有不少网友还没有使用 arm，我先贴个图，给大家一个感性的认识。



王紫豪-XiaomaGee(15959622) 20:46:15

这个就是我们的主角

王紫豪-XiaomaGee(15959622) 20:47:35

其实讲微处理器的应用，是没什么东西可说的，也没什么理论，所以跟上几节课不一样，我只能大概表达下我个人的观点和一些看法

数据手册



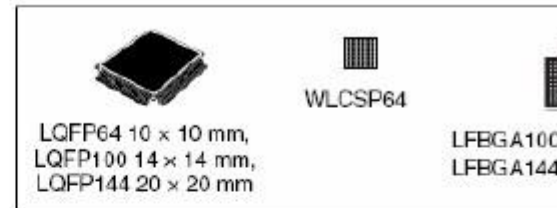
王紫豪-XiaomaGee(15959622) 20:49:33

这个是 stm32f103xc 数据手册的表头，也能大概表现出他的一些资源。不好意思，接了个电话。

我们就用 datasheet 上的第一页，简单地说一下，这个 cpu 的资源。

功能

- **内核：ARM 32位的Cortex™-M3 CPU**
 - 最高72MHz工作频率，在存储器的0等待周期访问时可达1.25DMips/MHz(Dhrystone 2.1)
 - 单周期乘法和硬件除法
- **存储器**
 - 从256K至512K字节的闪存程序存储器
 - 高达64K字节的SRAM
 - 带4个片选的静态存储器控制器。支持CF卡、SRAM、PSRAM、NOR和NAND存储器
 - 并行LCD接口，兼容8080/6800模式
- **时钟、复位和电源管理**
 - 2.0~3.6伏供电和I/O引脚
 - 上电/断电复位(POR/PDR)、可编程电压监测器(PVD)
 - 4~16MHz晶体振荡器
 - 内嵌经出厂调校的8MHz的RC振荡器
 - 内嵌带校准的40kHz的RC振荡器
 - 带校准功能的32kHz RTC振荡器
- **低功耗**
 - 睡眠、停机和待机模式
 - V_{BAT}为RTC和后备寄存器供电
- **3个12位模数转换器，1μs转换时间(多达21个输入通道)**
 - 转换范围：0至3.6V
 - 三倍采样和保持功能
 - 温度传感器
- **2通道12位D/A转换器**
- **DMA：12通道DMA控制器**
 - 支持的外设：定时器、ADC、DAC、SDIO、I²S、SPI、I²C和USART
- **调试模式**
 - 串行单线调试(SWD)和JTAG接口
 - Cortex-M3内嵌跟踪模块(ETM)
- **多达112个快速I/O端口**
 - 51/80/112个多功能双向的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



- **多达11个定时器**
 - 多达4个16位定时器，每个定时器用于输入捕获/输出比较/PWM或通道和增量编码器输入
 - 2个16位带死区控制和紧急刹车，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)
 - 系统时间定时器：24位自减型计时器
 - 2个16位基本定时器用于驱动DAC
- **多达13个通信接口**
 - 多达2个I²C接口(支持SMBus/PMBus)
 - 多达5个USART接口(支持ISO7816 IrDA接口和调制解调控制)
 - 多达3个SPI接口(18M位/秒)，2个I²S接口
 - CAN接口(2.0B 主动)
 - USB 2.0全速接口
 - SDIO接口
- **CRC计算单元，96位的芯片唯一标识码**
- **ECOPACK®封装**

表1 器件列表

| 参考 | 基本型号 |
|-------------|-------------------------|
| STM32F103xC | STM32F103RC、STM32F103ZC |
| STM32F103xD | STM32F103RD、STM32F103ZD |
| STM32F103xE | STM32F103RE、STM32F103VE |

首先我们可以看到，stm32f103 是基于 arm cortex-M3 内核的微控制器。

王紫豪-XiaomaGee(15959622) 20:51:58

其实 arm cortex 家族, 包含了三个系列, 分别为 A 系列, R 系列和 M 系列。

A 系列就是我们手机里、pad 里等等场合用的东西, 性能较高, 主频上 GHz,

王紫豪-XiaomaGee(15959622) 20:53:19

R 系列是面向实时系统场合的, 这个我了解的不多; 目前来说产品线也不是太广。

M 系列是面向“深度嵌入”的场合, 我们可以理解成“微控制器”这个四个字所代表的方向。

王紫豪-XiaomaGee(15959622) 20:54:53

当然, 每个子系列也不是简单的一种而是一个庞大的产品线。例如 A8, A15; M0, M3、M4 等。

王紫豪-XiaomaGee(15959622) 20:56:18

乍一看, 资源多的有点吓人, 3 个 12bit adc, 2 个 12bit adc, 11 个定时器, 13 个通信端口 (spi, i2c, uart, usb, can, sdio) 应有尽有,

王紫豪-XiaomaGee(15959622) 20:57:26

我们的惊讶就是他的优势所在, 就这样的微处理器, 48 脚的价格可以到 10CNY 内, 怎么可能不火呢?

王紫豪-XiaomaGee(15959622) 20:58:40

以上我对 stm32f103 这个微处理器做了一个大概的介绍, 让大家有个感性的认识, 下面结合着《iBoard 电子学堂》原理图, 来详细了解下这一部分的应用。

Router-广东<alexischat@gmail.com> 20:59:56

48 脚的价格可以到 10CNY 内, 怎么可能不火呢?

10.00 元钱内?? 这样少。。

王紫豪-XiaomaGee(15959622) 21:00:13

对的。

大家同时下载, 可能把 cnblog 的服务器拖垮了

★嵌入式学习者(1326703841) 21:00:56

我刚买的 RB 才 14.

我刚买的 RB 才 14.

王紫豪-XiaomaGee(15959622) 21:01:40

《iBoard 电子学堂》是一个综合型的学习系统; 微处理器以 cortex-M3 为主导; fpga 以 飓风四代为学习对象, 包含了大量的模拟电路和代码



下面我们要讲的，就是这货的原理图。

王紫豪-XiaomaGee(15959622) 21:05:00

第四页是 stm32 部分的原理图，他仅包含微控制器和他的外围，不实现其他功能。

王紫豪-XiaomaGee(15959622) 21:06:38

首先，这张图最重要的一个，也是最大的一个就是微处理器了，他牢牢地霸占了原理图中央的位置。

王紫豪-XiaomaGee(15959622) 21:06:38

首先，这张图最重要的一个，也是最大的一个就是微处理器了，他牢牢地霸占了原理图中央的位置。

从一个通用的微控制器角度来说，我们设计的时候要把握好以下几点：

王紫豪-XiaomaGee(15959622) 21:08:43

①供电

微处理器的电源就相当于我们的口粮，吃不饱饭是干不了活的；所以，电源电路稳定性、质量的重要性，关系到整个系统可靠地运行。

王紫豪-XiaomaGee(15959622) 21:10:04

我记得当初 stm32 推广的时候，有个广告语，叫：只需要 x 个电容，既能可靠地运行。

王紫豪-XiaomaGee(15959622) 21:11:58

电源方面的问题，请大家参考我的第一篇群课《嵌入式系统电源的设计》，这篇群课，我介绍了我们常用电源的拓扑结构及特征。

王紫豪-XiaomaGee(15959622) 21:13:09

stm32f103 仅仅使用一个 3.3V 电源既可，简化了我们的设计。（其实很多高级的玩意，需要很多组电源，就拿 iboard 上面用的 fpga 来说，cycloneIV，使用了 3.3V，2.5V，1.2V 三个）

王紫豪-XiaomaGee(15959622) 21:14:21

这一部分，就是 stm32f103 的供电
去耦电容在上面。

王紫豪-XiaomaGee(15959622) 21:14:21

这一部分，就是 stm32f103 的供电口；

去耦电容在上面。

电源说完了，下面说第二个。

王紫豪-XiaomaGee(15959622) 21:16:06

②时钟

时钟是微处理器运行的动力，就像我们的心脏跳动一样。

王紫豪-XiaomaGee(15959622) 21:17:10

从大的框架来说，微处理器我们可以认为是一个庞大的时序逻辑，所以他的每一部动作，几乎都需要时钟来驱动；时钟不稳定，微处理器当然也就不稳定了。

刚才我说过，stm32f103 最大可以工作到 72MHz，这个就是他的驱动时钟的最大频率，也就是我们常说的主频。

刚才我说过，stm32f103 最大可以工作到 72MHz，这个就是他的驱动时钟的最大频率，也就是我们常说的主频。

王紫豪-XiaomaGee(15959622) 21:19:15

为了增强电路的电磁环境，高速的微处理器一般都内建 PLL（也就是锁相环），输入较低的时钟，倍频后再给 cpu 使用。

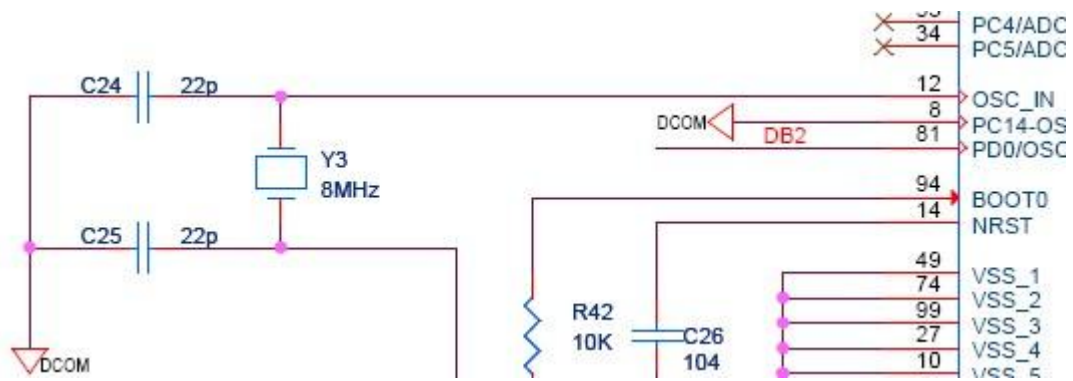
王紫豪-XiaomaGee(15959622) 21:19:15

为了增强电路的电磁环境，高速的微处理器一般都内建 PLL（也就是锁相环），输入较低的时钟，倍频后再给 cpu 使用。

stm32 其实包含了两个时钟输入，一个是高频的，用于主时钟，一个是低频的，用于 RTC 等。

我们就使用了一个。

王紫豪-XiaomaGee(15959622) 21:20:30



外接 8M 无源晶体，通过内部振荡器，然后 PLL 倍频后（9 倍），产生 72M 时钟给 cpu

下面是课间休息时间，21:30 继续

云淡风轻(948124155) 21:21:32



王紫豪-XiaomaGee(15959622) 21:21:35

我去主持一个问答抽奖

★嵌入式学习者(1326703841) 21:21:41



流水断玄月(573419891) 21:21:52

在那抽奖啊

Tracy(827468435) 21:21:55

啥奖啊

王紫豪-XiaomaGee(15959622) 21:36:10

刚才我们讲了 cpu 电路的两大要点：

①电源 ②时钟

下面我们继续说。

微处理器第三个重要的东西：

③复位电路

小牛(779746494) 21:37:48

复位电路 go ahead

王紫豪-XiaomaGee(15959622) 21:37:57

复位电路可以看做是 cpu 工作的一个起点。

小牛(779746494) 21:38:50

yeah

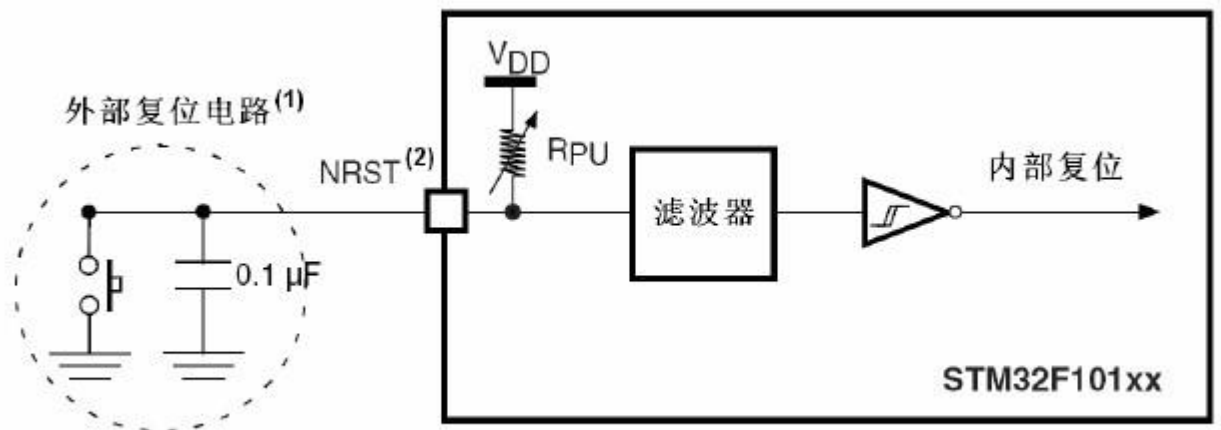
王紫豪-XiaomaGee(15959622) 21:38:52

良好的复位电路，就是 cpu 工作的一个良好的开始；当然，复位电路包含了很多（上电复位，看门狗复位。。。。）

王紫豪-XiaomaGee(15959622) 21:40:06

以往的 arm7 系统中，大部分需要一个复位芯片，例如 sp706 这样的等等。

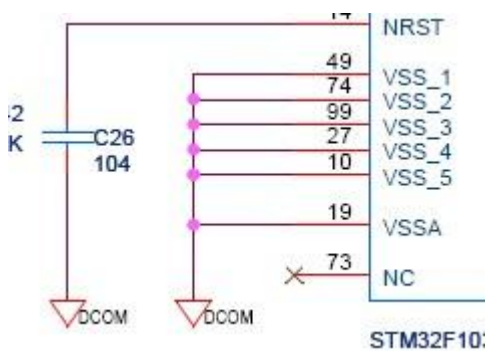
stm32f103 内部良好的设计，使得复位电路大大简化，仅仅需要一个电容完成。



ai141

王紫豪-XiaomaGee(15959622) 21:42:02

这是 datasheet 上的框图，大家可以看一下，一个很重要、很复杂的问题，这个 cpu 确如此简单的。



王紫豪-XiaomaGee(15959622) 21:43:28

经过几年时间，以及成千上万个 cpu 的实验，我发现这个复位电路还是很可靠的。我们总结一下，我们这里概括了一下，一个微处理器三个工作要点①电源②时钟③复位

王紫豪-XiaomaGee(15959622) 21:44:49

这几个要点适应很多种微处理器，包括最简单的 51 单片机

如果哪位网友用 51 单片机搭出来的板子，程序跑步起来（排除代码的原因），就从上面这三点开始找，总能找到问题

王紫豪-XiaomaGee(15959622) 21:46:43

下面，我就说说 stm32f103 其他的一些设计要点。

王紫豪-XiaomaGee(15959622) 21:46:43

下面，我就说说 stm32f103 其他的一些设计要点。

如果非要接着刚才的说，那么第四个要点就是下载、调试口了。

王紫豪-XiaomaGee(15959622) 21:47:59

怎么把我们写的代码，编译成 2 进制后，“灌”到 cpu 中，也是一个关键。

目前，新一代的微处理器都支持 isp，这个在生产中是很方便的事情。

stm32f103vc 也支持 isp 下载，不过我没有留这个功能，因为我有更好的方法

王紫豪-XiaomaGee(15959622) 21:49:40

stm32f103 包含了 JTAG 和 SWD 两种调试接口，他们都能用过商业化的仿真器来下载和调试程序，jtag 其实是历史遗留下来的、也是最通用的接口。JTAG 这东西，可是一个非常有名，也是一个非常有用的东西；他让我们能深入到 cpu 内部去了解他的运行状态。

王紫豪-XiaomaGee(15959622) 21:51:14

但是 jtag 对于 stm32f103vc 来说，占用引脚有点多了，《iboard 电子学堂》把他舍弃了，使用了新一代的两线调试口 SWD

王紫豪-XiaomaGee(15959622) 21:52:14

两根线，再加上一个地，就组成了一个非常简单，也非常实用的调试、下载电路。实际工作过程中，非常稳定，线的长度我最长弄到了 0.5 米，速度还能够达到最高速 4M

王紫豪-XiaomaGee(15959622) 21:53:29



配合我山寨的 ARM-OB JLINK 使用。

关于这个小型的下载、调试器，原理图和烧写代码都是开源的，大家可以去论坛或者我的博客下载。

电子三脚猫(49304253) 21:58:05

我想要插入一个问题可以么？ 调试器的性能指标有哪些？您做的这个调试器支持多少个断点？

王紫豪-XiaomaGee(15959622) 21:58:19

跟 jlink 完全一样（速度 4M），其他完全一样

下面我说一个我个人的习惯。

含有微处理器的工程设计中，我都会占用一个 IO 口，加一只 LED，调试初期，用来检测程序运行状态。

特别是刚开始调试的时候，上电瞬间就是整个电路最容易出问题的时刻（要么冒烟，要么就正常了）

王紫豪-XiaomaGee(15959622) 22:00:49

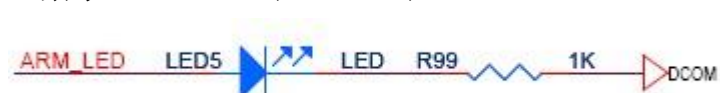
这个时候，使用万用表测量，通过串口监视都来不及，所以 LED 是最常用的监测方法。

王紫豪-XiaomaGee(15959622) 22:00:49

这个时候，使用万用表测量，通过串口监视都来不及，所以 LED 是最常用的监测方法。

当然，LED 也有坏的时候，我当时做 200A 电流的电子负载时，LED 就坏了，吓我一身冷汗。

王紫豪-XiaomaGee(15959622) 22:01:52



12345(739467432) 22:01:58



王紫豪-XiaomaGee(15959622) 22:02:27

在这个系统中，arm有，fpga有，cpld也有
这是我的一个习惯，拿出来跟大家分享。

王紫豪-XiaomaGee(15959622) 22:03:19

下面，我跟大家探讨下 IO 相关的问题。

Tracy(827468435) 22:05:07

指示灯

电子三脚猫(49304253) 22:05:53

要加在默认是下拉输入的 IO 吧？

还是通过 IO 直接驱动 led？

王紫豪-XiaomaGee(15959622) 22:06:32

数字 I/O，一般也叫 GPIO，就是通用 IO，是我们使用微处理器过程中，用到最多的东西

王紫豪-XiaomaGee(15959622) 22:08:27

举个例子说，我们输出去控制 LED 的亮和灭，是输出模式，也就是 OUTPUT；
我们需要探测按键是否按下，就是 input 模式，也就是输入模式。

别看一个小小的 IO，他的学问大了去了，接错了线，或者电平不匹配，就影响整个系统的稳定性。

王紫豪-XiaomaGee(15959622) 22:11:07

STM32 的 IO，分为以下几种模式：

- 输入浮空
- 输入上拉
- 输入下拉
- 模拟输入
- 开漏输出
- 推挽式输出
- 推挽式复用功能
- 开漏复用功能

王紫豪-XiaomaGee(15959622) 22:12:10

输入包含四种，浮空输入就是啥也不管他，就让他悬在那块。。当然，这个状态可是不定的。

王紫豪-XiaomaGee(15959622) 22:13:18

上拉、下拉是简称，就是在浮空输入的基础上，增加了上拉电阻或者下拉电阻；
让他稳定在一个我们预知的状态。

王紫豪-XiaomaGee(15959622) 22:14:20

模拟输入更简单，就是模拟信号输入，这个不归咱们数字信号内。

12345(739467432) 22:18:18

通用推挽和复用推挽有什么区别呀

王紫豪-XiaomaGee(15959622) 22:18:35

开漏，就是漏极开路的意思。

《... ..》(793042486) 22:18:56

漏极直接输出吗？

jouson(周生)(355377135) 22:19:04

有什么应用的不同

王紫豪-XiaomaGee(15959622) 22:19:06

英文叫 OD，现实中，还有开集电极（换成三级管子），叫 OC 门

Tracy(827468435) 22:19:50

通用推挽和复用推挽有什么区别呀

王紫豪-XiaomaGee(15959622) 22:20:10

这个结构应用非常广泛，他可以用作“线与”功能。

通用和复用，区别不大，这个可以自己看看，我主要讲原理，那个是细节问题的对，i2c 的应用，就是他最好的证据。

jouson(周生)(355377135) 22:21:09

线与？

电子三脚猫(49304253) 22:22:58

datasheet 里找不到 stm32 IO default 状态的描述= =

王紫豪-XiaomaGee(15959622) 22:23:06

这个是 stm32 的 io 结构图，大家要学会看框图；虽说我们通过这个不能造出 ic 来，但是他也是官方最具有说服力的资料。

默认状态是悬浮输入的

王紫豪-XiaomaGee(15959622) 22:24:43

烧程序嘛

当今微控制器经历了 5V 逻辑到 3.3 甚至更低电压的转换。

电子三脚猫(49304253) 22:24:50

哦

王紫豪-XiaomaGee(15959622) 22:24:59

你想得太多了，呵呵

今天的课面向初学者的。

王紫豪-XiaomaGee(15959622) 22:25:38

这就给我们一个很现实的问题，我们需要跟不同电平的 IO 连接，怎么去转换？怎么去匹配呢？

王紫豪-XiaomaGee(15959622) 22:26:06

客气

从结构上讲，目前我们可能遇到的有 TTL 电平，CMOS 电平和 1vttl 及 LVCMOS 等等。

王紫豪-XiaomaGee(15959622) 22:27:21

从电压上来讲，也就是 10~15V 的老逻辑器件，5V 的器件和 3.3V 器件居多

Tracy(827468435) 22:27:42

恩，是啊，可以通过 OC、OD 简单的转换

紫豪-XiaomaGee(15959622) 22:28:33

工控中，12~24V IO 电平居多，这么高的电压，我们一般通过 OC 或者 OD 门，或者 光耦等隔离手段实现，还好工控中一般速度都不快。

当然， 还可以通过继电器的。

控制接触器、电磁阀等等，最好使用隔离的方法。

王紫豪-XiaomaGee(15959622) 22:30:38

3V 到 5V 的转换，如果 5V 驱动 3V，我们可以通过增加嵌位二极管来实现，当然，这中间最好串联电阻，以减小电流。

3V 如果驱动 5V，这要看看我们的 5v

是 CMOS 逻辑，还是 TTL 逻辑，TTL 逻辑可以直接驱动的。

王紫豪-XiaomaGee(15959622) 22:32:02

如果是 CMOS 逻辑，则需要使用转换电路

王紫豪-XiaomaGee(15959622) 22:32:02

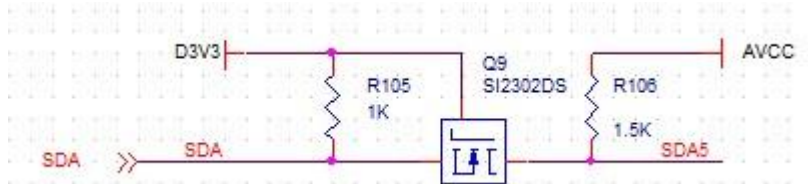
如果是 CMOS 逻辑，则需要使用转换电路

因为 CMOS 电路的高电平门限是 $0.7V_{CC}$ ，如果 $v_{cc}=5$ ，则最低 3.5V

王紫豪-XiaomaGee(15959622) 22:33:33

这里我们可以通过 例如像 74LVC164245，74LVC4245 等芯片实现，也可以使用 mos 管实现。

mos 管双向转换的方法，我给大家贴个图，稍等。



这里就是使用 一个 nmos，完成双向数据的转换（3.3V 和 5V）

这个电路的局限性就是速度上不了 MHz，不过一般也就够了。

王紫豪-XiaomaGee(15959622) 22:36:24

这个电路，来自于 I2C 的规范；NXP 的。

另外 5V 到 3.3V

还有一种方法，是使用能 5V 容忍的 3.3V 逻辑器件。

王紫豪-XiaomaGee(15959622) 22:38:21

I/O 应用五花八门，大家一定要注意电平，不同低电平不要随便接。

今天的课就到这里了，下节课接着讲 STM32 的外设，包括 UART，usb，spi，ADC，dac，timer 等等。