## 零死角玩转STM32



## 如何用串口下载程序

淘宝: firestm32.taobao.com

论坛: www.firebbs.cn

### 主讲内容



- 01 安装USB转串口驱动
- 02 硬件连接
- 03 ISP—键下载
- 04 ISP一键下载原理分析

#### ISP简介



1、ISP (In-System Programming) 在系统可编程,指电路板上的空白器件可以编程写入最终用户代码,而不需要从电路板上取下器件,已经编程的器件也可以用ISP方式擦除或再编程。

2、ISP通过芯片内部的自举程序(即Bootloader,由芯片厂家烧写好,不得更改)来选定一种串行的外设,对芯片内部的FLASH进行编程。

### ISP简介



3、最常用的ISP方式就是通过串口下载,最主要的优点就是成本低,缺点是只能用于下载程序,不能硬件仿真。

4、普通ISP和一键ISP,普通ISP在下载程序的时候需要手动配置BOOT的启动方式,而一键ISP则通过独特的硬件电路和上位机配合使用来达到一键下载的功能。

### 安装USB转串口驱动



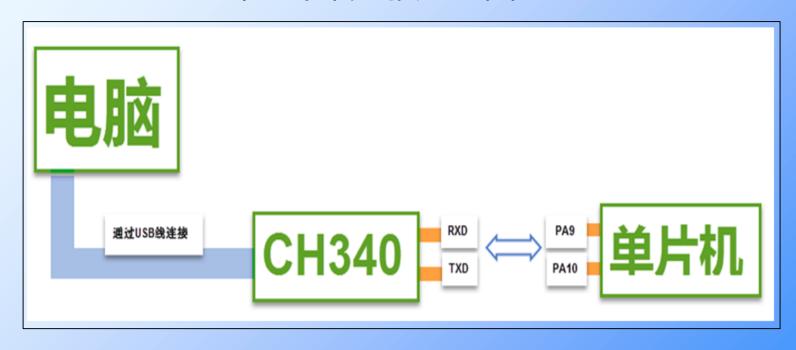
# 光盘资料\A盘(资料盘)\4-开发软件\USB转串口驱动-CH340



### 硬件连接



#### 串口下载连接示意图



- 1、USB转串口模块在开发板上是一个独立的模块,可通过调帽与其他串口连接
- 2、只有USART1才具有串口下载的功能

## 硬件连接





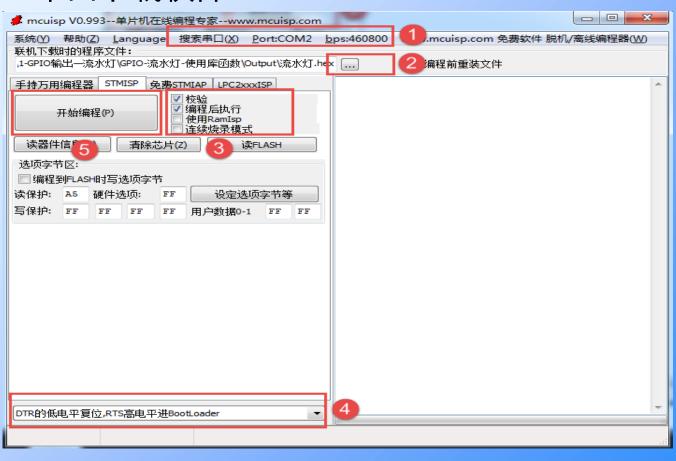
- 1、BOOTO和BOOT1接GND
- 2、RXD/TXD接 PA9/PA10

开发板默认的调帽不用动即可

#### MCUISP下载软件配置



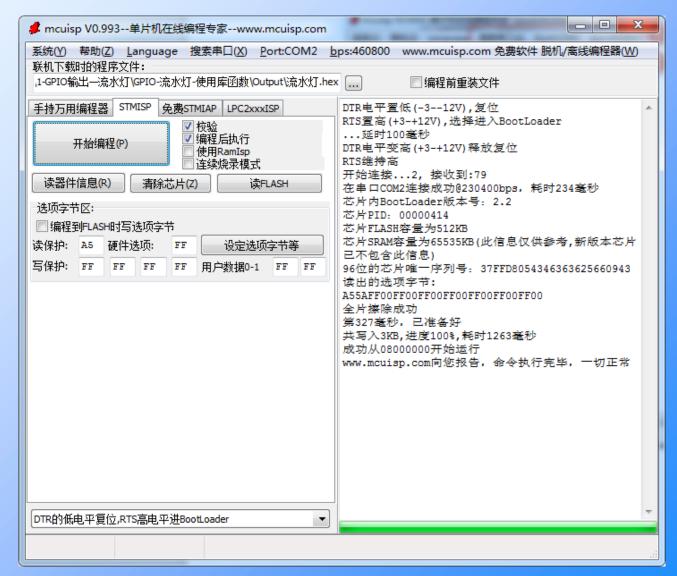
#### 光盘资料\A盘(资料盘)\4-开发软件\ 串口下载软件



#### MCUISP下载软件配置



程序下载成功, 会自动运行, 如果不自动运行, 手动按开 发板的复位按 键即可。





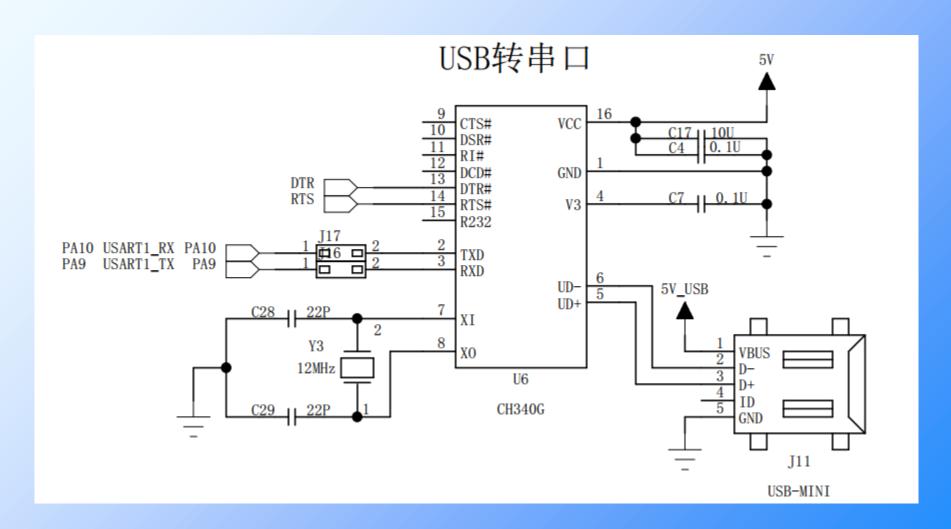
## BOOT启动方式选择

| 表格 4-1 B( | TOC | 配置。 |
|-----------|-----|-----|
|-----------|-----|-----|

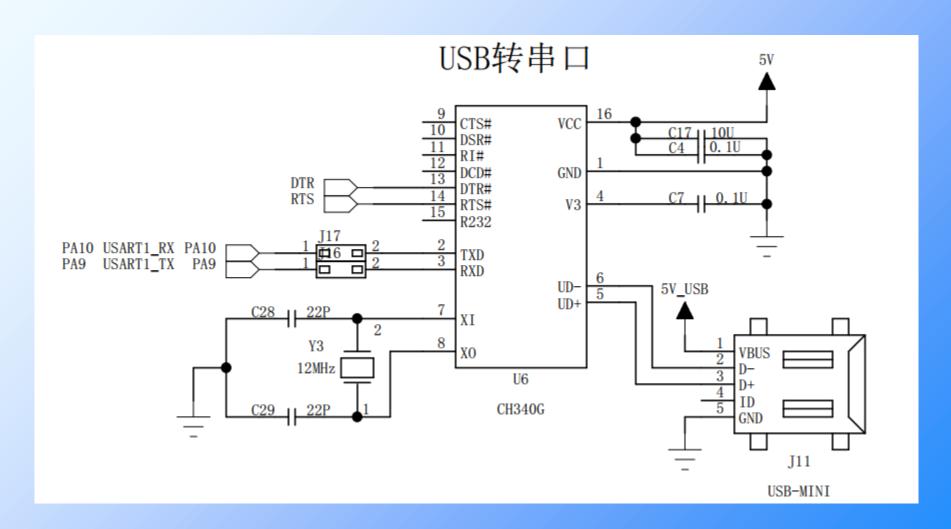
| BOOT0° | BOOT1 | 启动方式。     |
|--------|-------|-----------|
| 06     | X.o   | 内部 FLASH。 |
| 1.₽    | 0.    | 系统存储器。    |
| 10     | 1.0   | 内部 SRAM。  |

注意:开发板中的BOOT0和BOOT1均通过调帽接GND

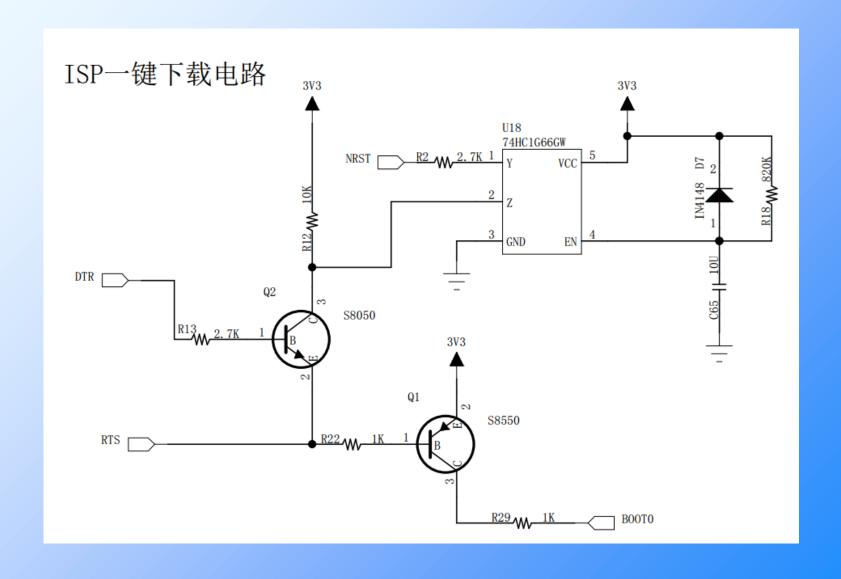














- 1、配置BOOT为ISP模式,BOOT1硬件固定为GND。RTS输出低,S8550导通,BOOT0被拉高。
- 2、复位,NRST为低,让系统重新启动进入ISP模式。DTR输出高,S8050导通,NRST被拉到跟RST一样的低电平。
- 3、系统进入ISP模式后,让BOOT0和NRST还原成原来的状态即可。(通过上位机控制RTS和DTR引脚实现)
- 4、ISP下载完毕之后,程序自动运行,(通过上位机控制RTS和DTR引脚实现)



# 上位机需要这样设置

RTS低电平进入BootLoader, DTR高电平进入复位

232标准(mcuisp软件就是按照这个标准设计的)

逻辑1:-3~-15V, 逻辑0:+3~+15V

TTL标准

逻辑1:3.3V, 逻辑0:0V

(一)、TTL 电平标准

逻辑0:0V 输出 L: <0.8V; H: >2.4V。 输入 L: <1.2V; H: >2.0V



- 5. 至此,很多人还会认为 U18、Q1、Q2 是多余的,用 U6 的 RTS 和 DTR 直接控制也可以。正常情况下,这样理解没有问题,但是我们忽略了一点,就是单片机上电瞬间如果 USB 转串口连接了电脑,DTR 和 RTS 的电平是变化的,如果不处理好,单片机会一直进入 ISP 模式,或者系统会复位多次,这种情况是不允许的。↓
- 6. 于是,就有了我们全新的一键 ISP 电路。我们主要是分析上电瞬间的逻辑关系,单片机上电时我们通过示波器观察波形得知 DTR 和 RTS 的电平是变化的,但是也有一个规律就是:只要 RTS 为低电平的时候,DTR 的电平也是低,因此一般情况 Q2 不会导通,但由于这两个 IO 口的电平存在"竞争冒险",会出现 RTS 的下降沿的时候刚好遇到 DTR 的上升沿,这个时候 Q2 导通,导致系统复位,而 BOOT0 此时有可能也为高电平,就会进入 ISP 模式。这个是不受我们控制的,我们不想系统出现这样的情况。因此加入了模拟开关来切断这种干扰。ℯ
- 7. 加入模拟开关 U18,通过控制 U18 的 4 脚的开关来达到隔离干扰电平的目的。下面 我们分析一下延时开关电路,上电瞬间,电容 C65 通过电阻 R18 来充电,由于电阻 100k 很大,电容的充电电流很小,等电容充电达到 U18 的 4 脚的有效电平 2V时,大概耗时 1S,在这个 1S 时间内 U18 的模拟开关是断开的,因此 RTS 和 DTR 的干扰电平不会影响到系统复位。系统正常运行。↩

## 零死角玩转STM32





论坛: www.firebbs.cn

淘宝: firestm32.taobao.com