

STM8 用串口下载及调试从入门到精通

最近老是碰到有许多朋友问到STM8不用ST-LINK或其他仿真器怎么进行程序开发及下载,其实STM8 FLASH大于16K的产品目前都已内置ISP程序,可以用串口进行ISP方式代码下载,速度飞快.配合IAR或常用的STVD,就算你没有ST-LINK,也一样可以轻松实现程序开发.下面给出通过串口下载程序及配合STVD进行调试的具体操作流程,供有需要的朋友参考.

作为举例,本文档采用下图的官方三合一套件进行演示.(需要此套件的朋友可与深圳北天星科技公司联系)





一. 首先,你必须得下载一个CXSTM8,同时得安装如下软件:

1. STVD 软件:



sttoolset.exe
STMicroelectronics

安装好后在开始菜单可以找到如下项,其中  ST Visual Develop 是一个编译环境,

 ST Visual Programmer 是一个程序下载软件,如下图.

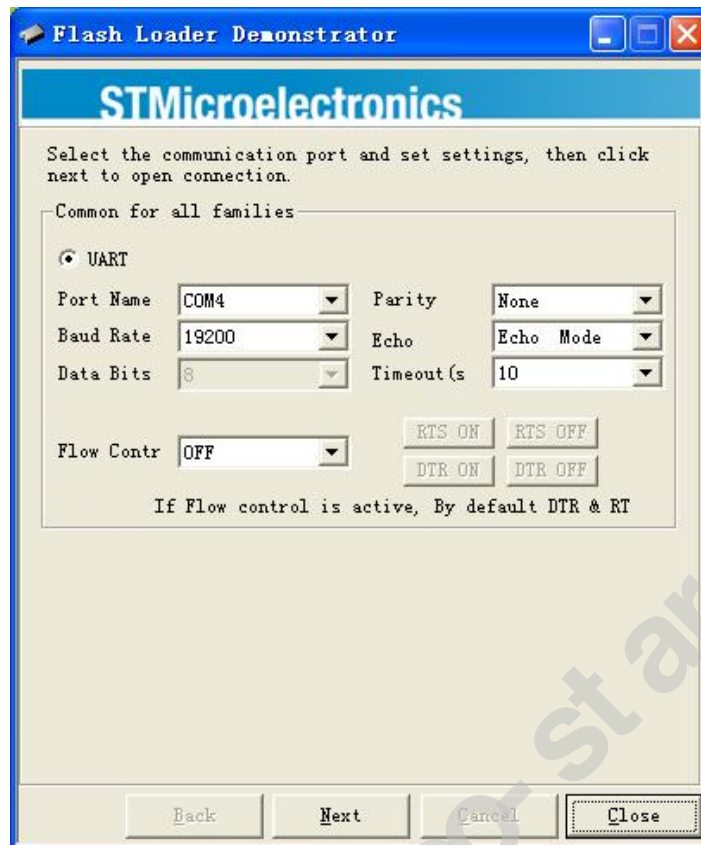


2. 安装 ST ISP 软件:



Flash Loader Dem...
Setup Launcher
STMicroelectronics

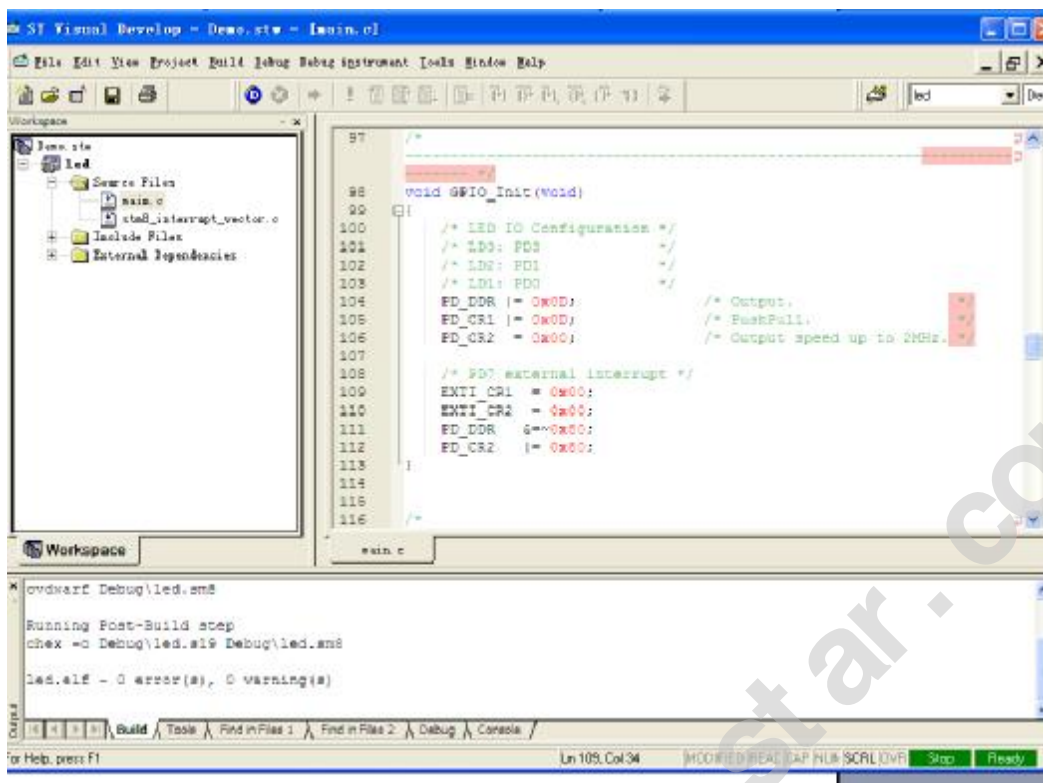
安装好打开后界面如下:



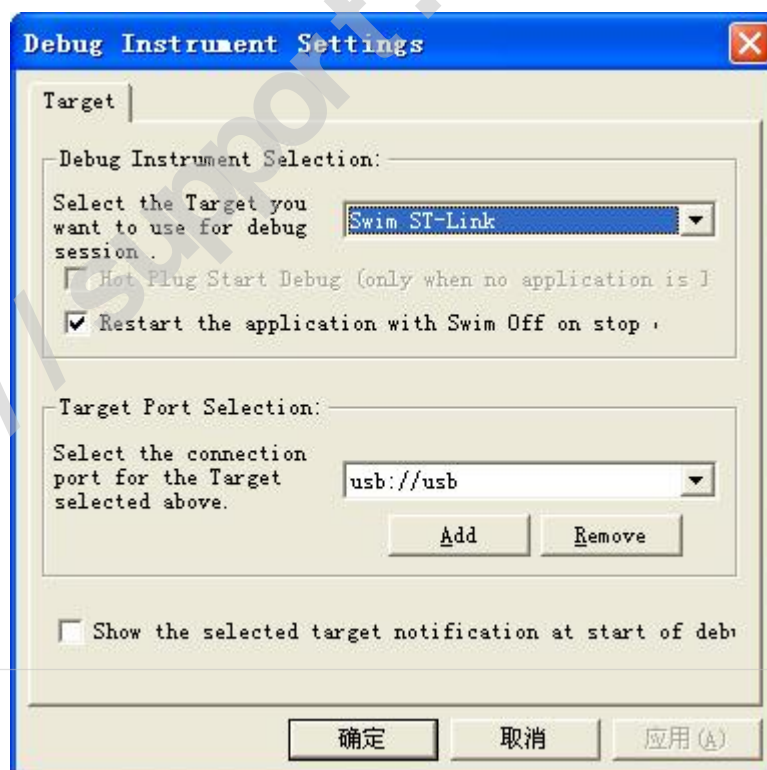
二,安装好上述软件后,便可以进行 STM8 开发了,STVD 的应用及配置网上有许多资料,下面仅对部分设置作简要介绍,



运行 STVD ,打开后界且加载例子程序后如下图:



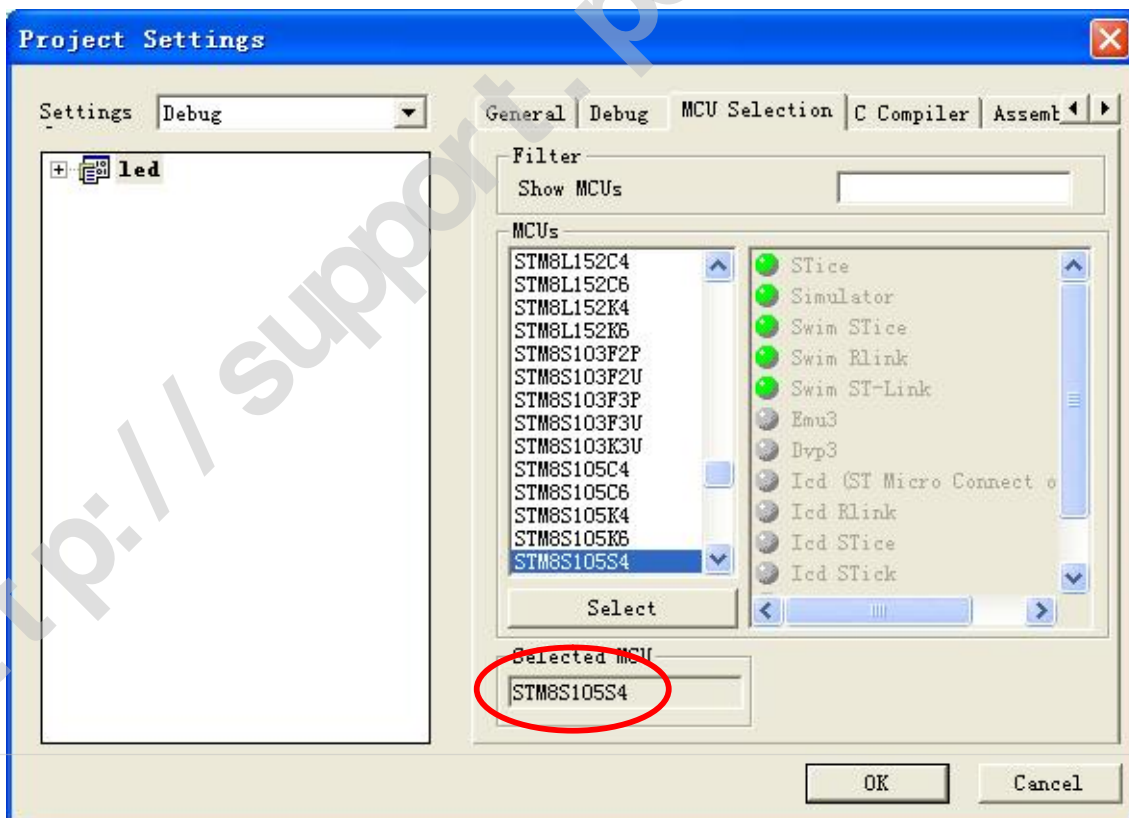
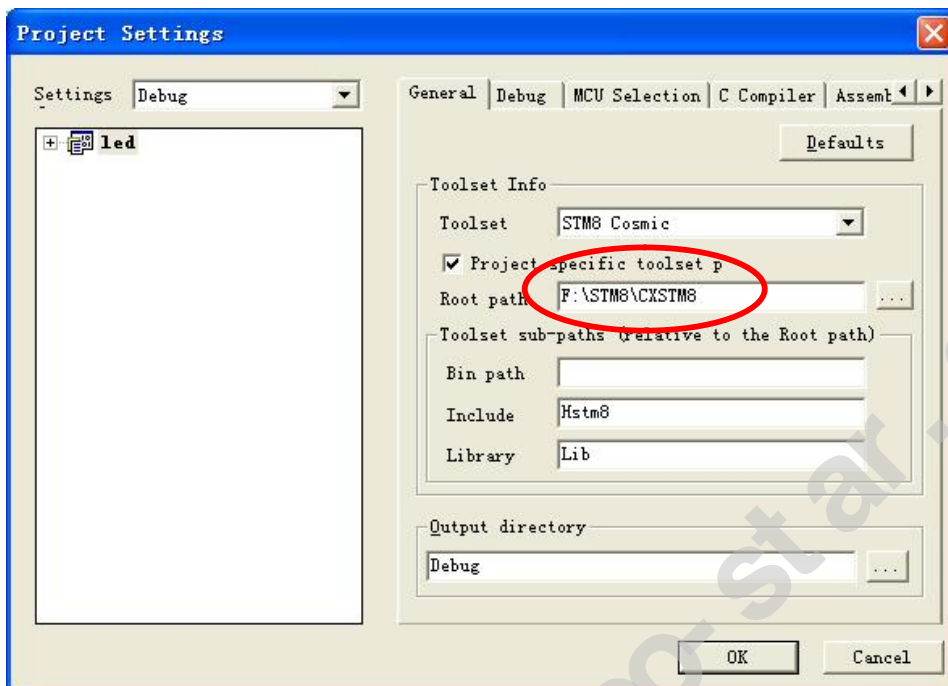
在菜单栏打开 **Debug instrument Tool** **Target Settings...** 如下图设置:






在工具栏点击工具栏如下图红色标记图标



打开设置窗,如下图设置即可:

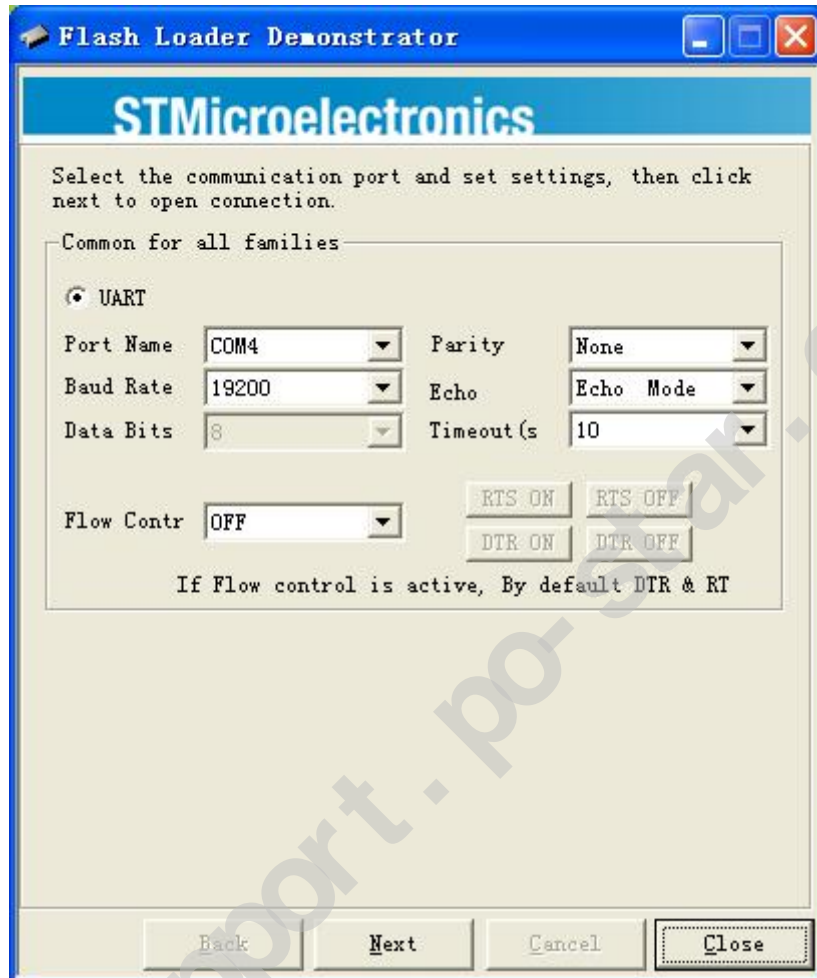


二. 设置好后按   , 如程序与设置正确, 则编译能正常通过. 此时如 ST-LINK 与目标板连接正常, 则能进行仿真调试下载等操作了.

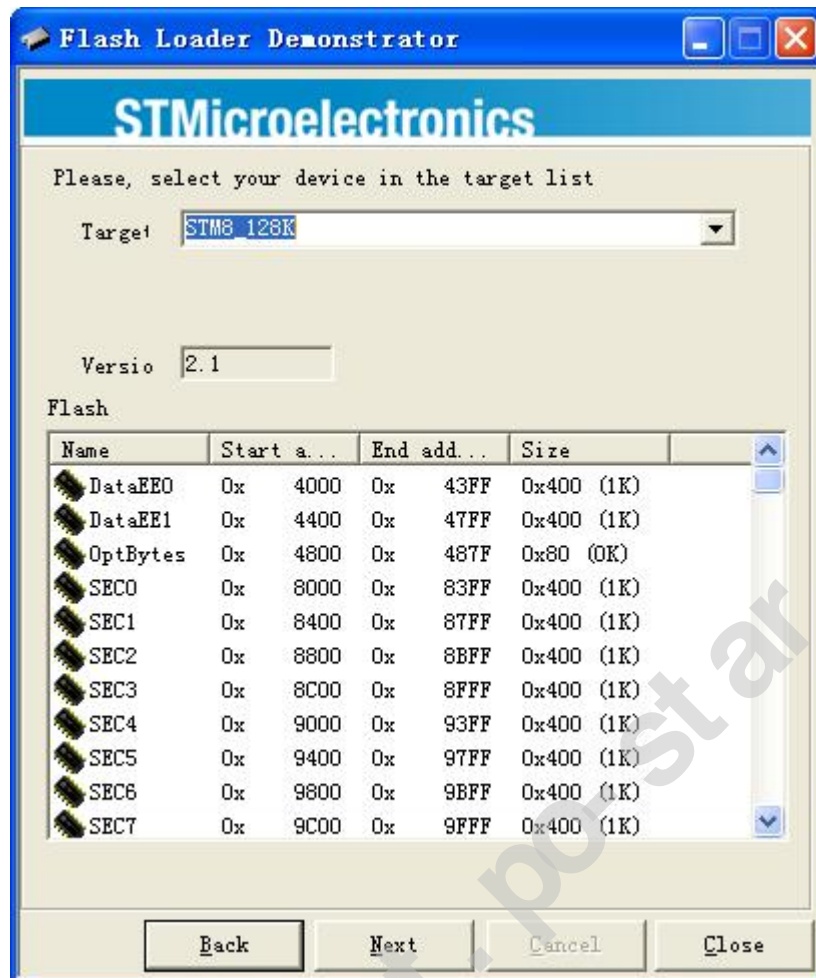
三. 串口 ISP 下载流程:



1. 打开 ,显示如下界面:



将目标板与电脑用串口线连接,同时给目标板通上电源,在上图软件中选择你能用的串口号及适当的波特率,(如是 USB 转串口,必须在打开上述软件前确保 USB 转串口线已插在电脑上且已安装好驱动),按一下目标板 RESETD 按键,再点击如上图的 NEXT 按键,正常情况下,弹出如下窗口:

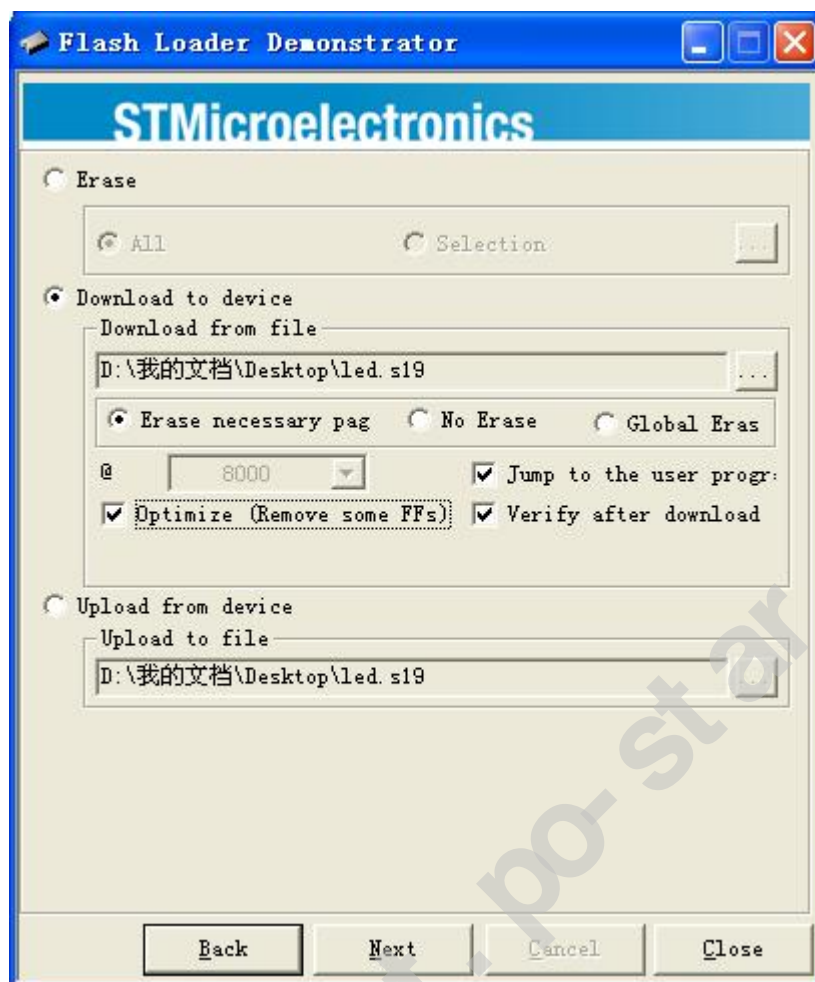


注意:如果等待一段时间后出现如下提示,则请按一次 RESET 后再行操作 NEXT.

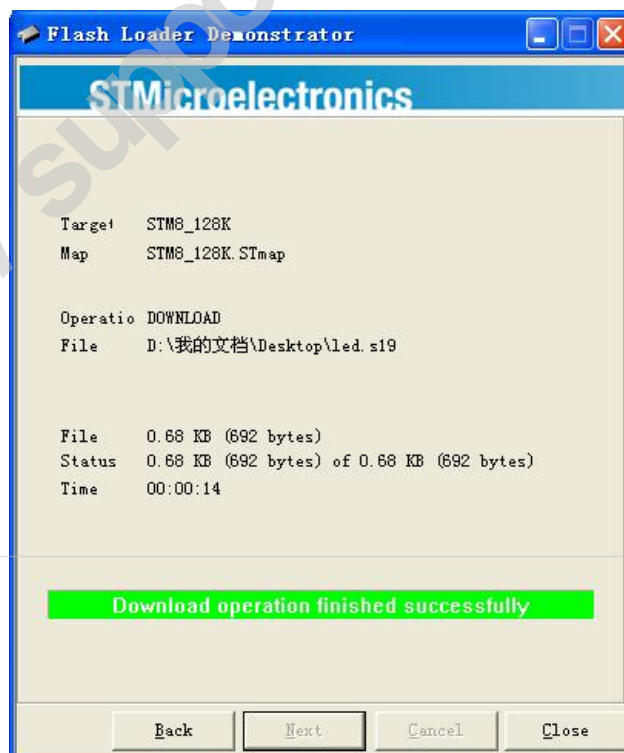


如果按 RESET 几次后仍显示上图,则很不幸,你的目标 IC 或许没有使能 ISP 功能,此时,请参照本文档的后续章节“怎样使能 STM8 ISP 功能”进行处理.

如果一切正常,选择与目标芯片相适应的 FLASH 大小,点击 NEXT,显示如下界面:



选择需要下载的目标代码,点击下一步,等待几秒,成功更新目标 IC 代码后的界面如下图所示:



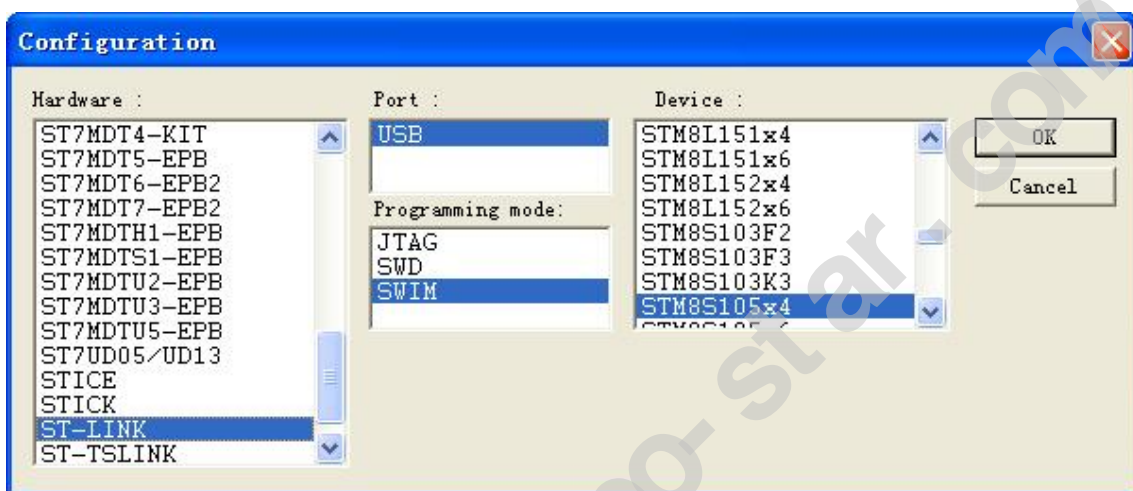
四. 怎样使能 STM8 ISP 功能:

如果发现 ISP 不能正常使用,就要考虑是否禁用了 ISP 功能,开启 ISP 功能可按下面流程处理:

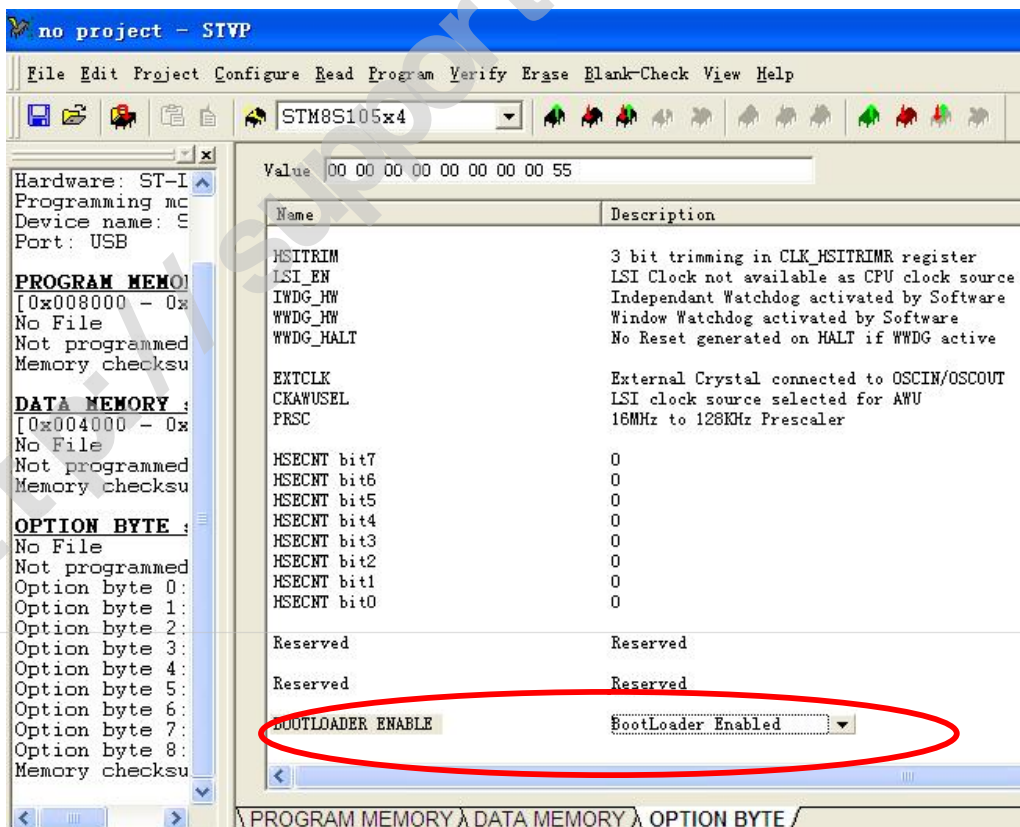
第一种方法:



点击 打开编译软件,选择 **Configure** 菜单,在弹出的设置窗口中如下图设置



然后在主窗口的 OPTION BYTE 页面中找到, **BOOTLOADER ENABLE** 且让其使能,如下图所示



设置好后点击 , 下载最新配置到目标 IC 即可, 将目标板重新上电后再测试串口下载功能, 正常情况下此时应能正常进行 ISP 程序下载了.

第二种方法:

在项目程序中加入使能 ISP 代码, 然后再将程序用 ST-LINK 或其他工具下载到目标 IC.

下面是 bluefeel 测试过的 ISP 使能代码:

```
#include "iostm8.h"
```

```
FLASH_DUKR = 0xae;
asm("nop");
FLASH_DUKR = 0x56;
asm("nop");
while(!(FLASH_IAPSR & 0x08));
asm("nop");
FLASH_CR2 = 0x80;
asm("nop");
FLASH_NCR2 = 0x7f;
asm("nop");
*((unsigned char *)0x487e) = 0x55;
asm("nop");
*((unsigned char *)0x487f) = 0xaa;
while(1);
```

V1.0

Liushengmei

2012-2-24