

## 试用 SlickEdit 开发 RTT

2011-08-28 20:46:38 | 分类: [编程开发](#) | 字号 订阅

关注 RTT 有一段时间了, 其开发方法让我用起来很不爽 😞 感觉像回到了 DOS 时代 😞, 所以 Radio 板到手有一段时间了也没有怎么动过。

准备:

SlickEdit16.0.2

MDK4.21

《使用 SlickEdit 编写、编译 ARM/KeilC 代码》

SVN 上最新的 STM32Radio 软件库

Radio 板一块 (居然没有版本号?!)

JLink 一台

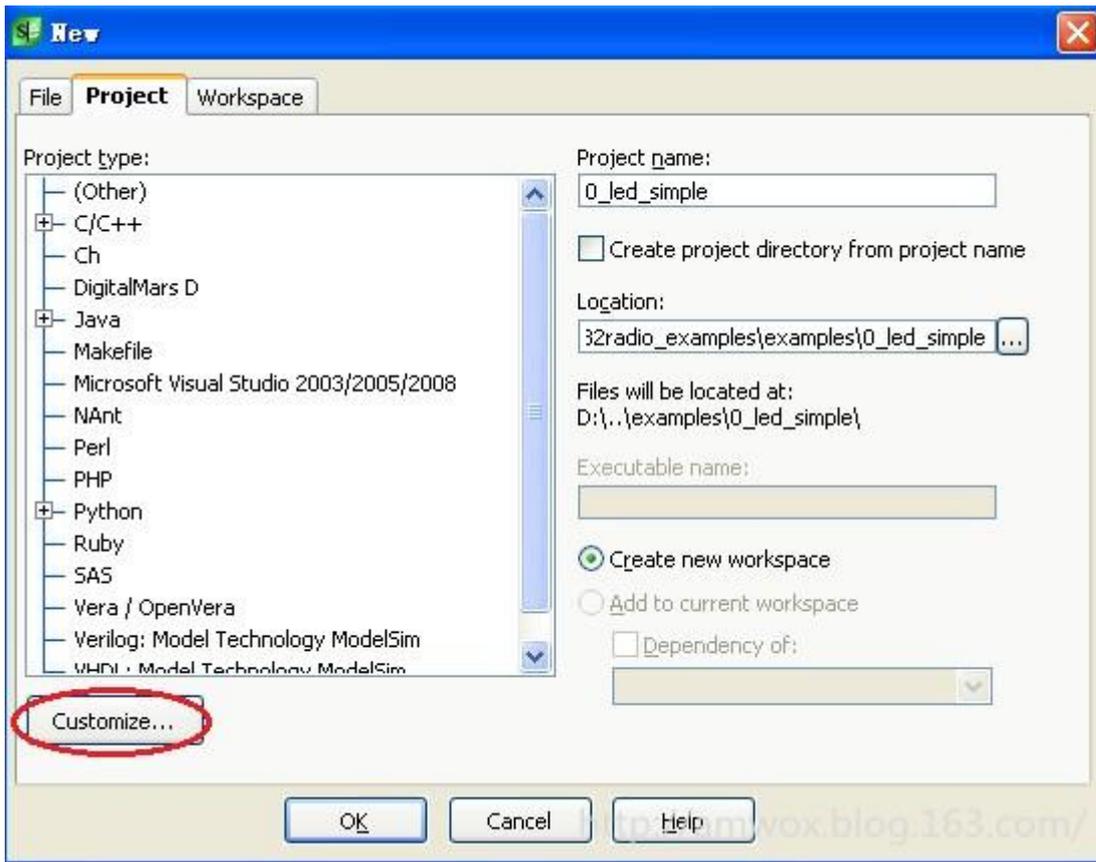
过程:

将 STM32Radio 复制出来, 去掉 SVN 相关的内容。选定 0\_led\_simple 作为测试平台。

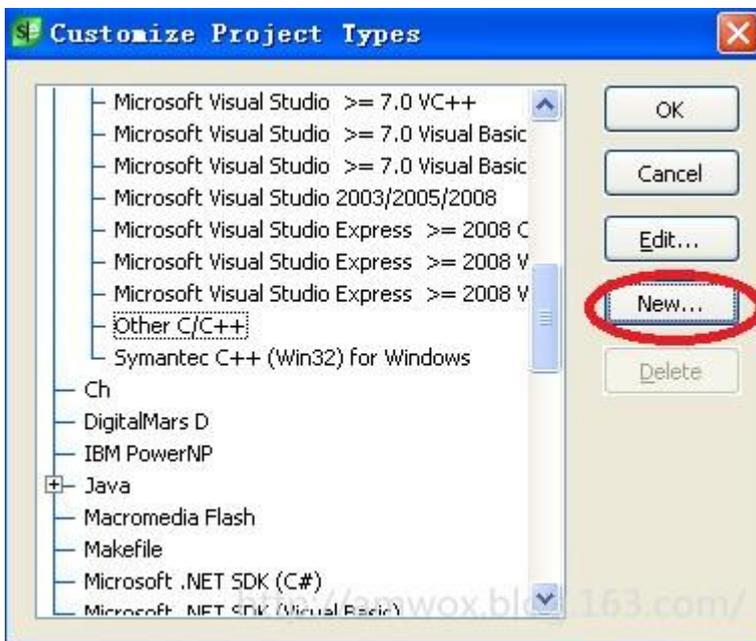
新建一环境变量 MDKARM=C:\KEIL\ARM\BIN40

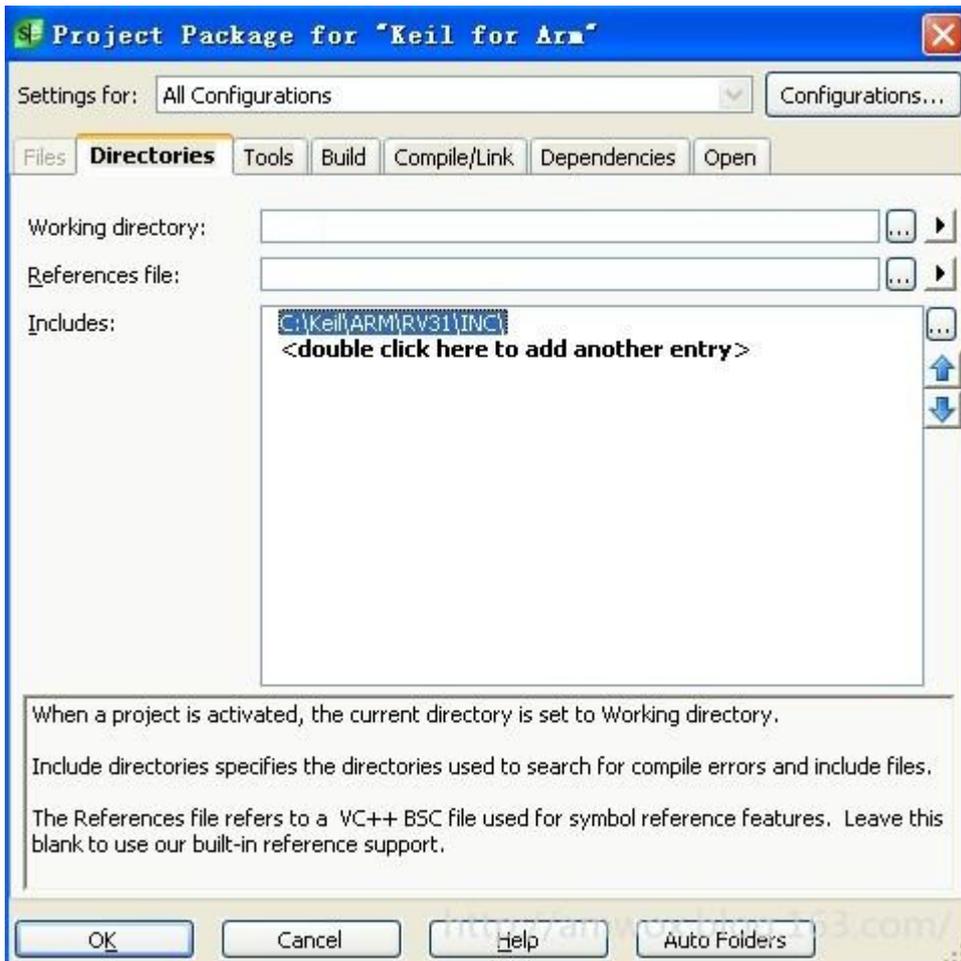


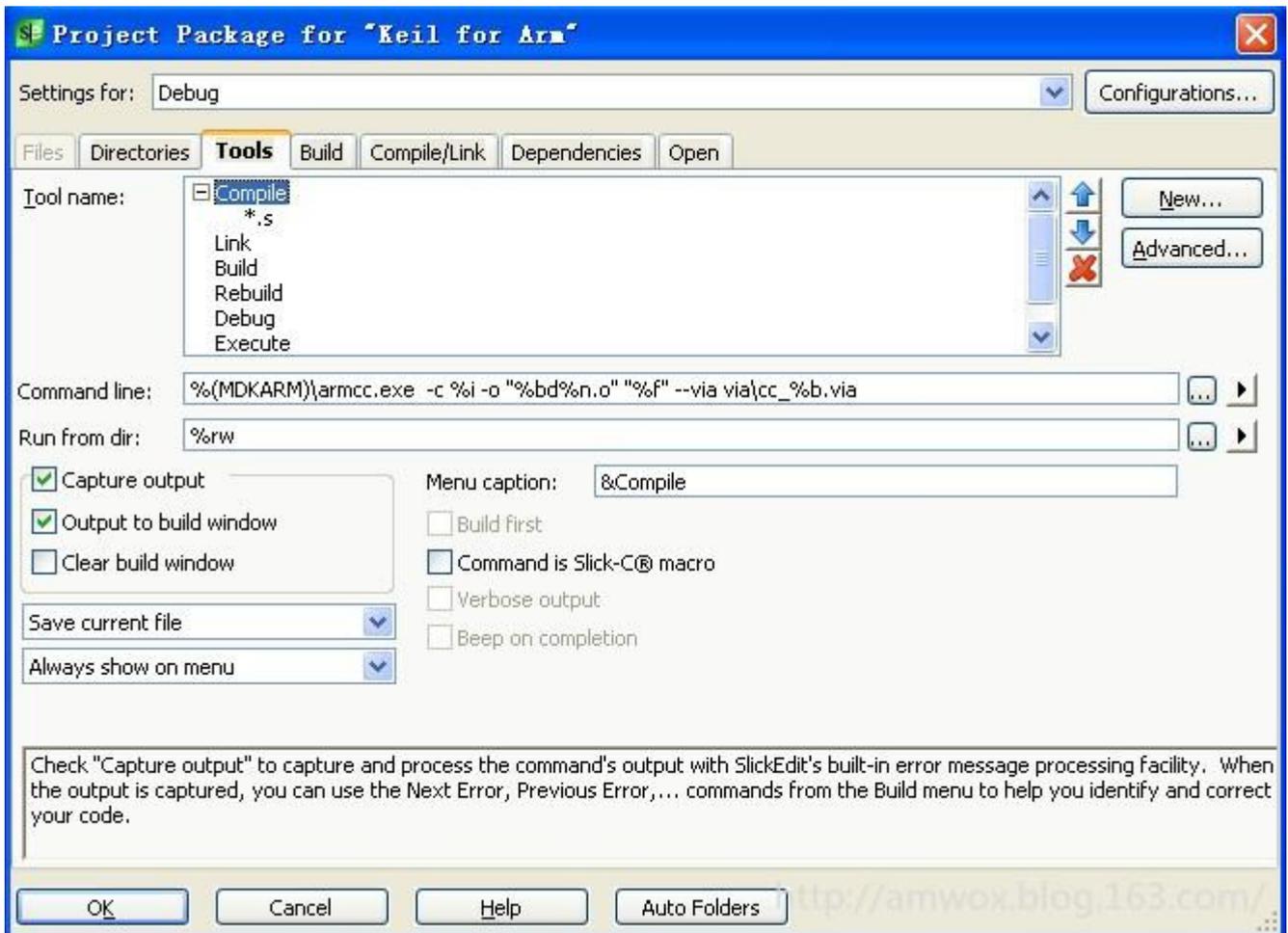
参考《使用 SlickEdit 编写、编译 ARM/KeilC 代码》和 project.uv2 新建一个 SlickEdit 工程。

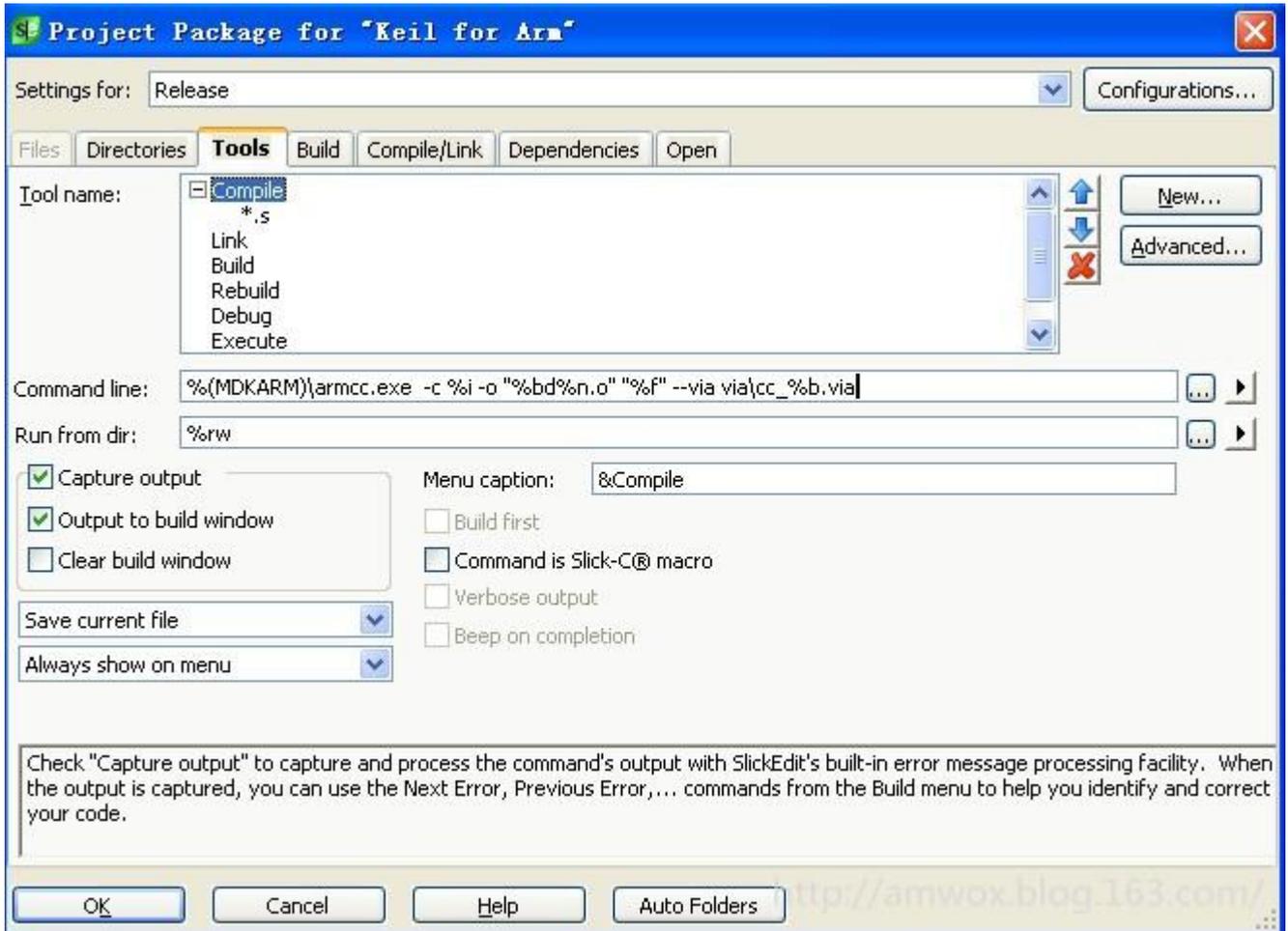


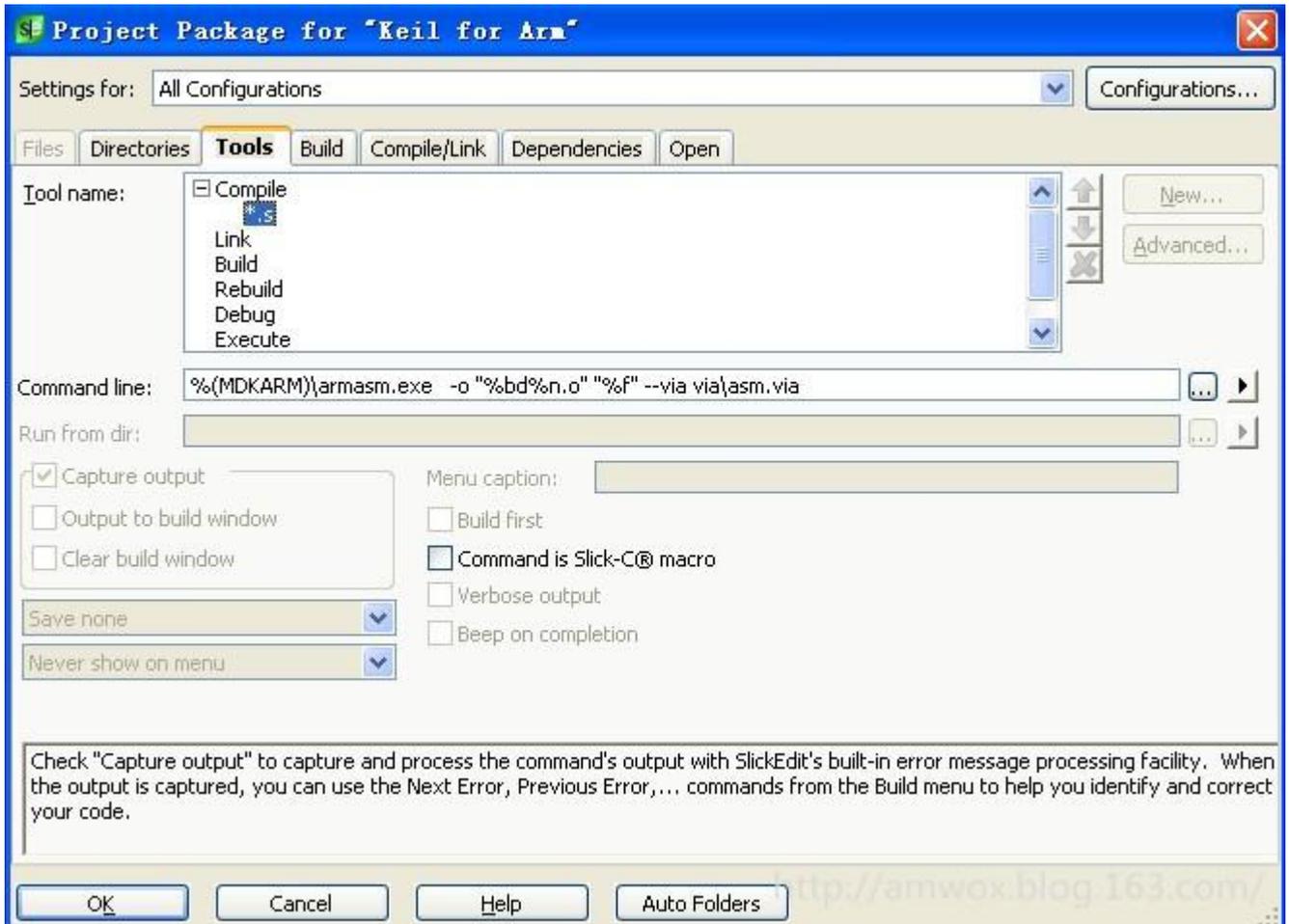
为以后使用方便建立一个新的工程类型：Keil for Arm

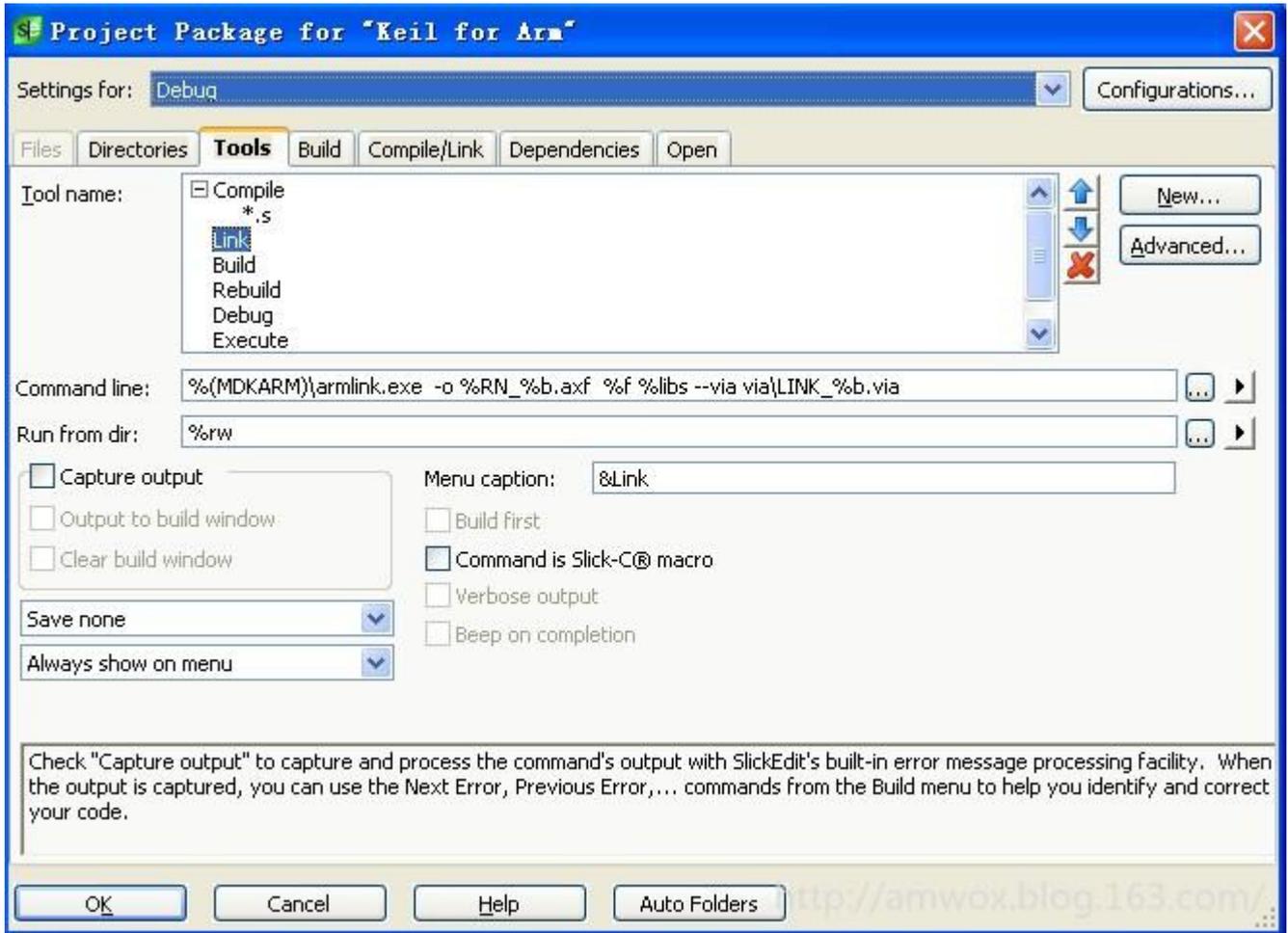


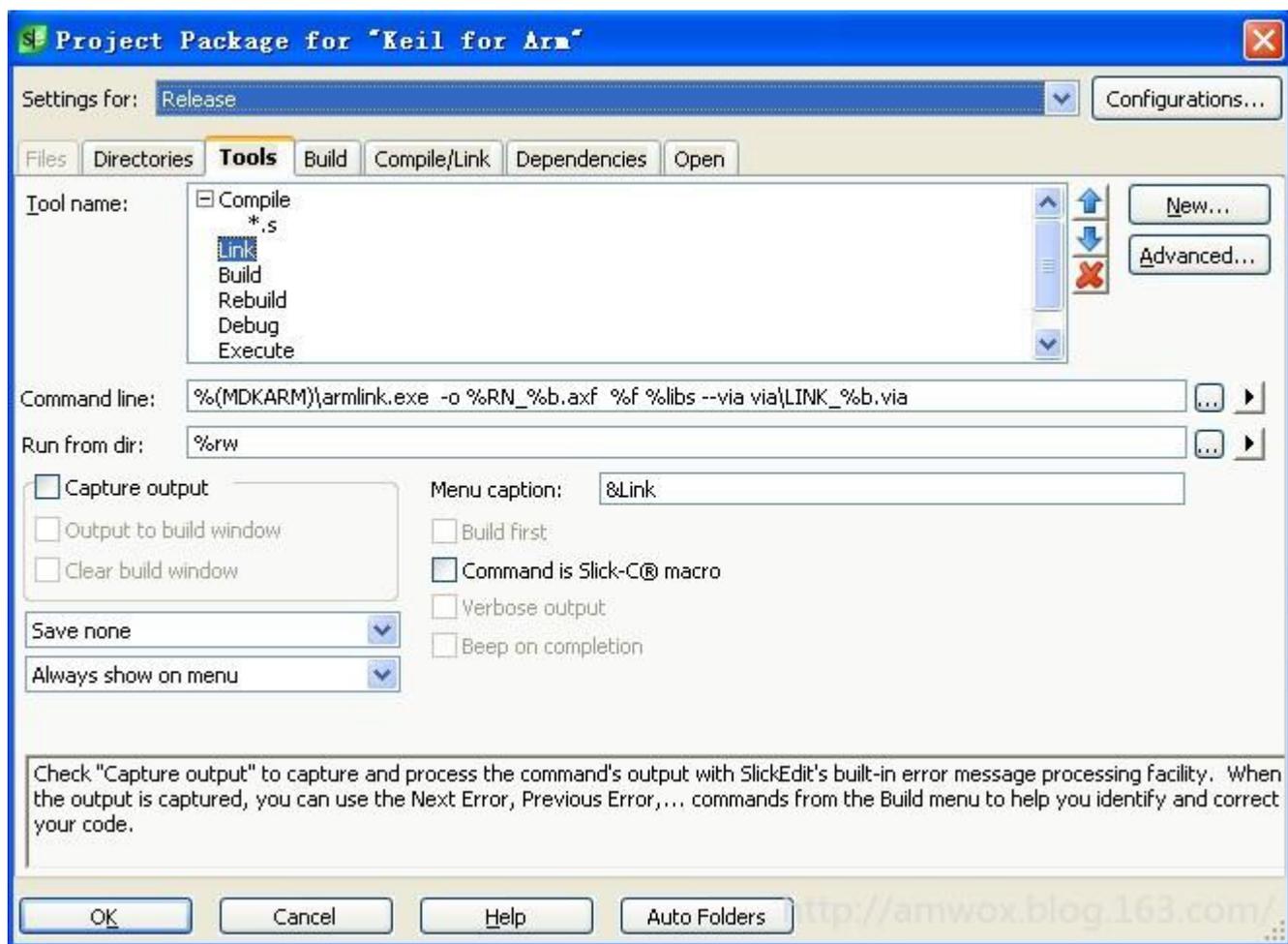




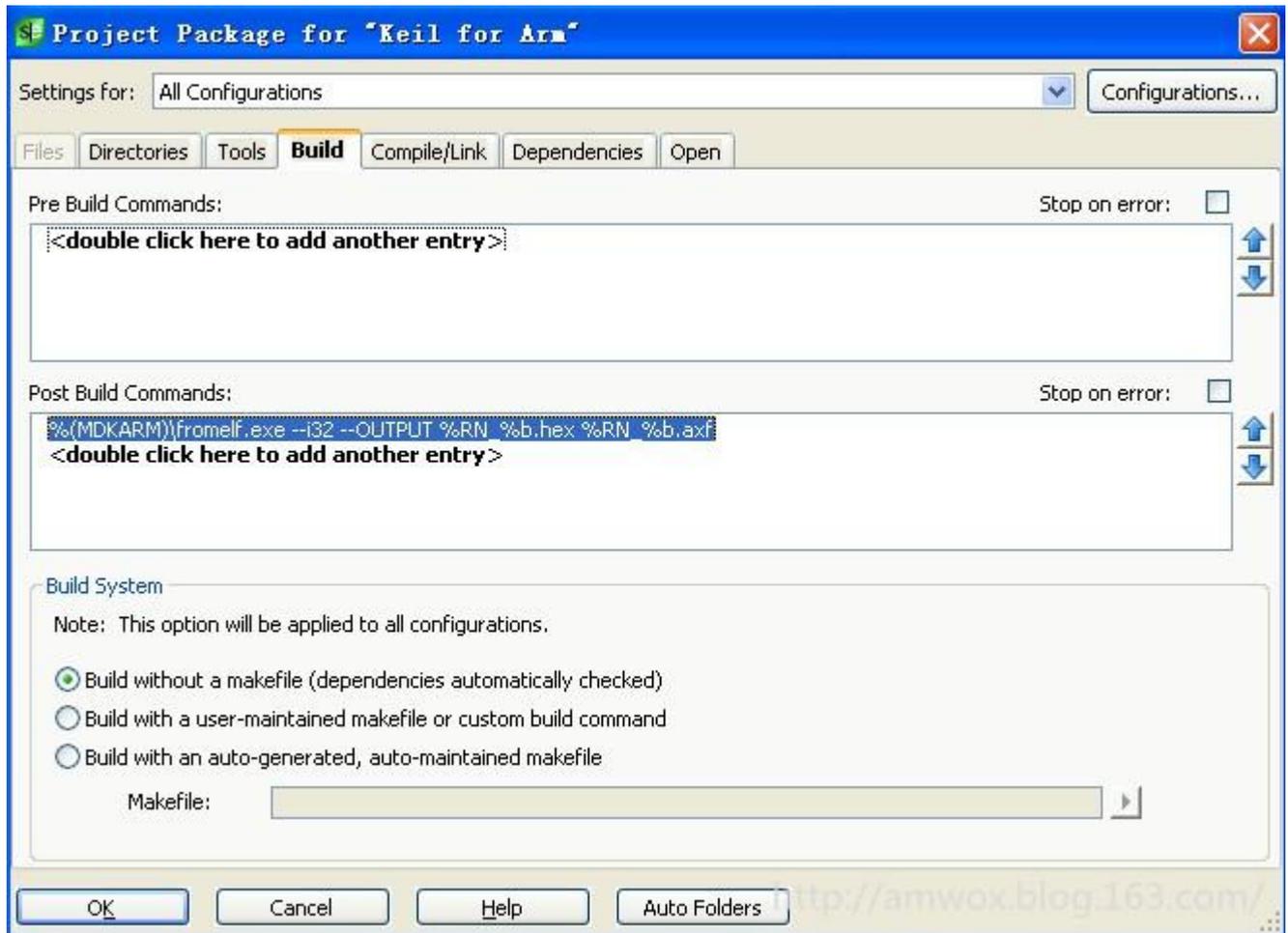






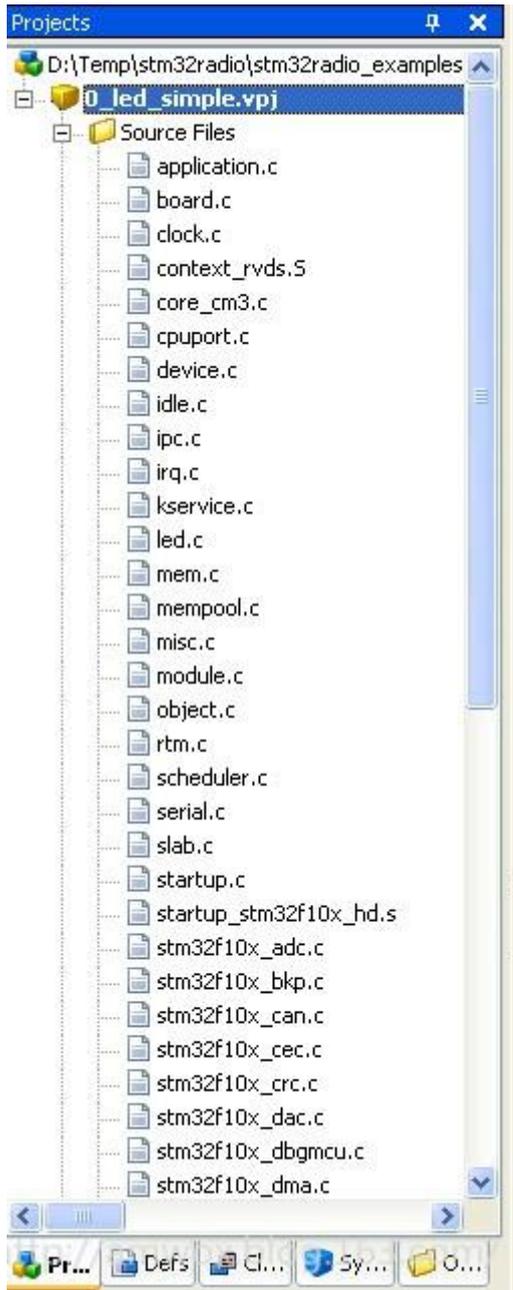


Build 和 Rebuild 不变

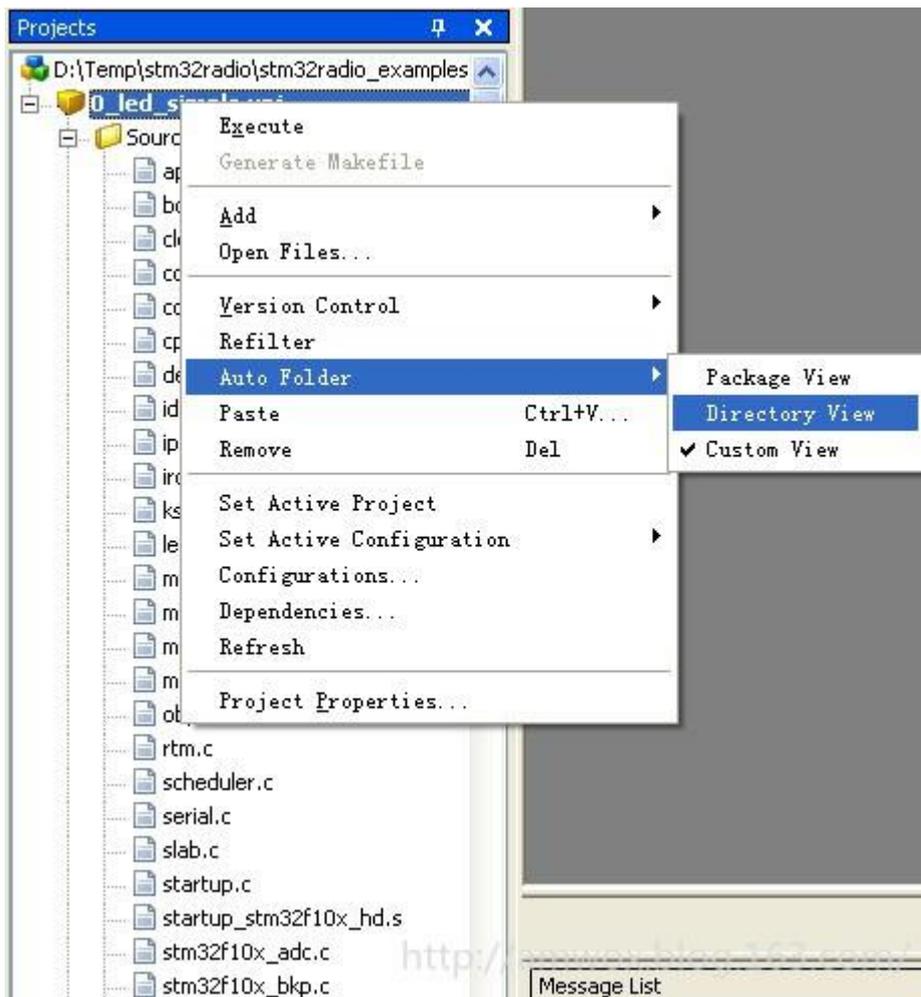


基本设定完毕。

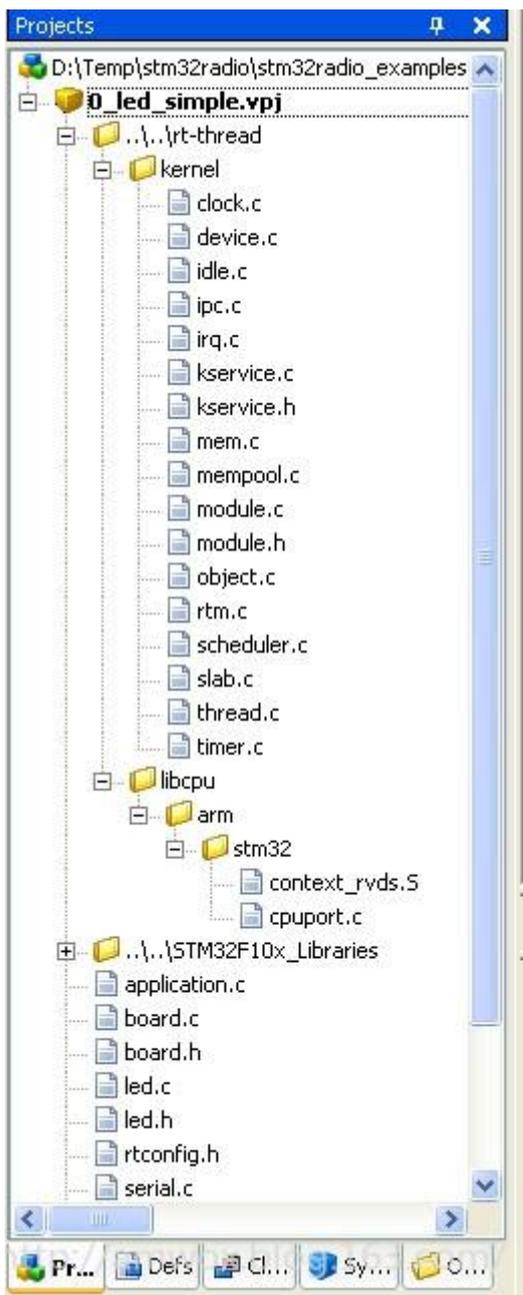
根据 Project.uv2 添加相应的文件



这次添加进来的文件组织结构不理想，改为目录显示方式



最终效果

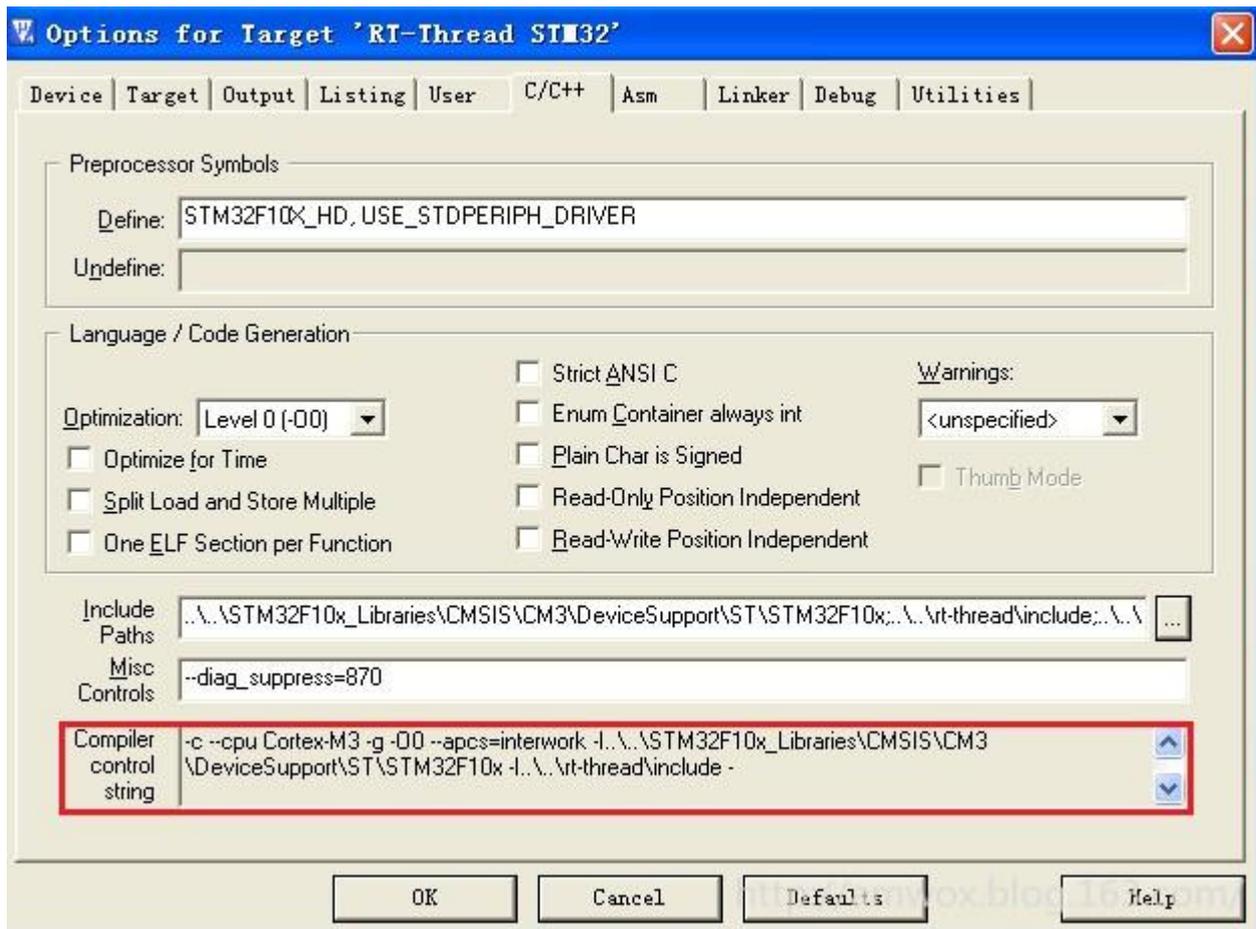


在工程目录下添加编译所需的文件 Scatter 和 via



via 中的文件指定了编译、链接行为，Scatter 中的文件指定了链接空间分配规则，可以在 KEIL 的编译输出文件夹中可以找到类似的文件

via 文件中选项可以参考 KEIL 中编译信息，每个文件都有对应的信息，不知道的可以直接 Copy 相关的内容。



好了现在可尝试编译

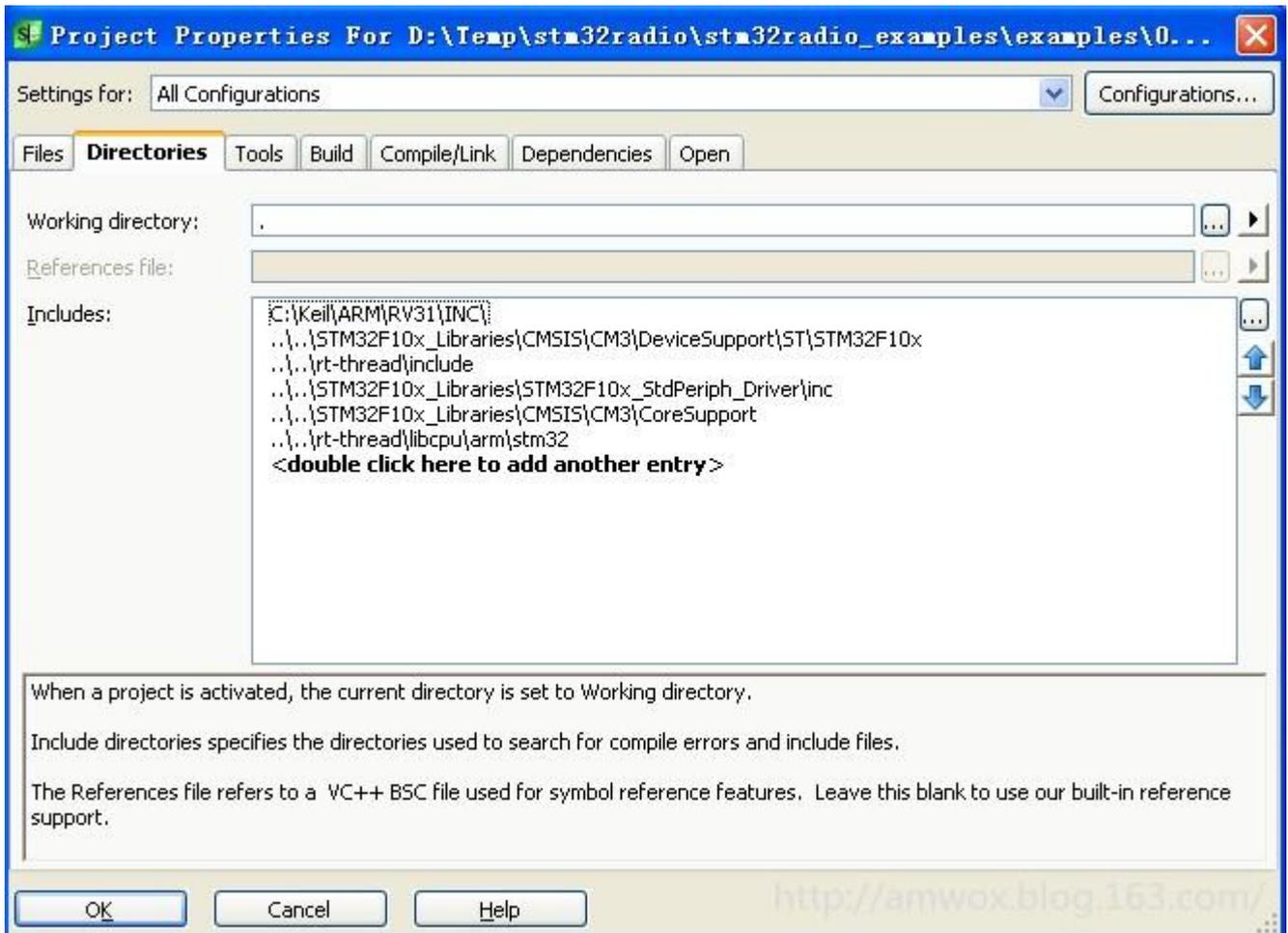
出现了错误:

```

----- Build Project: 'O_led_simple.vpj' - 'Debug' ----- VSLICKERRORPATH="D:\Temp\stm32radio\
stm32radio_examples\examples\O_led_simple"
clock.c
"..\..\rt-thread\kernel\clock.c", line 21: Error: #5: cannot open source input file "rthw.h": No such file or direc
#include <rthw.h>

```

这是没有指定包含的路径，可以在 via 文件中用 -I 选项指定，也可以配置文件中指定。



继续编译，还是出现错误 😞

```
stm32radio_examples\examples\0_led_simple"
context_gcc.S
"....\rt-thread\libcpu\arm\stm32\context_gcc.S", line 1: Error: A1167E: Invalid line start
  1 00000000 /*
"....\rt-thread\libcpu\arm\stm32\context_gcc.S", line 2: Error: A1159E: Label missing from line start
  2 00000000 * File      : context_gcc.S
"....\rt-thread\libcpu\arm\stm32\context_gcc.S", line 3: Error: A1159E: Label missing from line start
  3 00000000 * This file is part of RT-Thread RTOS
```

这是在加入文件是加多了与当前配置无关的文件造成的。再次编译...OK!

```
=====
Total RO Size (Code + RO Data)          28624 ( 27.95kB)
Total RW Size (RW Data + ZI Data)       3544 ( 3.46kB)
Total ROM Size (Code + RO Data + RW Data) 28740 ( 28.07kB)
=====
```

Build successful

```
C:\KEIL\ARM\BIN40\fromelf.exe --132 --OUTPUT 0_led_simple_Debug.hex 0_led_simple_Debug.axf 3.com
```

现在面临一个问题是如何把生成的 HEX 文件下载到 RADIO 板 🤖

从 KEIL 的帮助文件中找到这样的内容

### Program Flash Example

You may program Flash memory using the following command line:

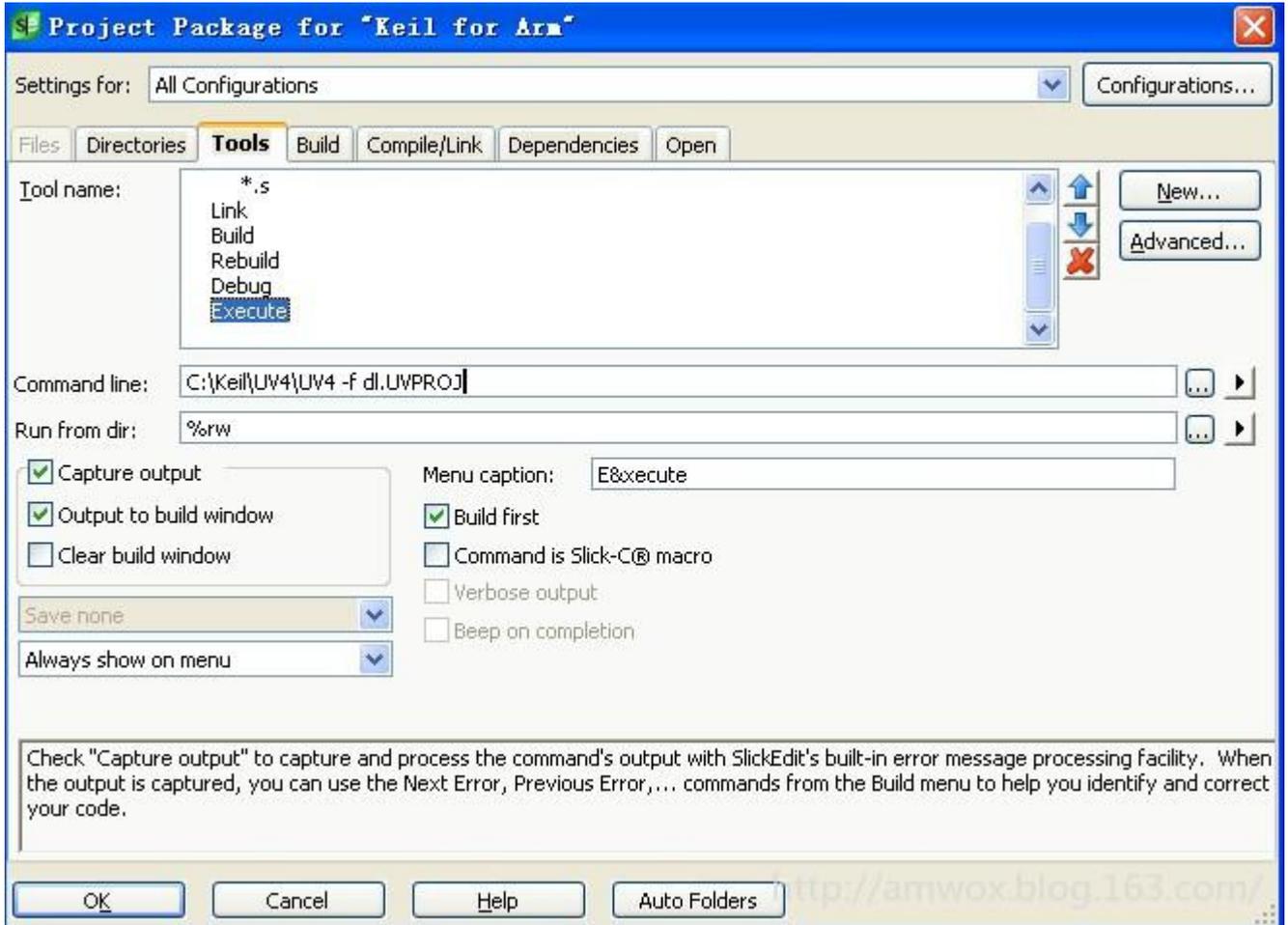
```
C:\Keil\UV4\UV4 -f Programming.UVPROJ -o Prg_Output.txt
```

#### Where

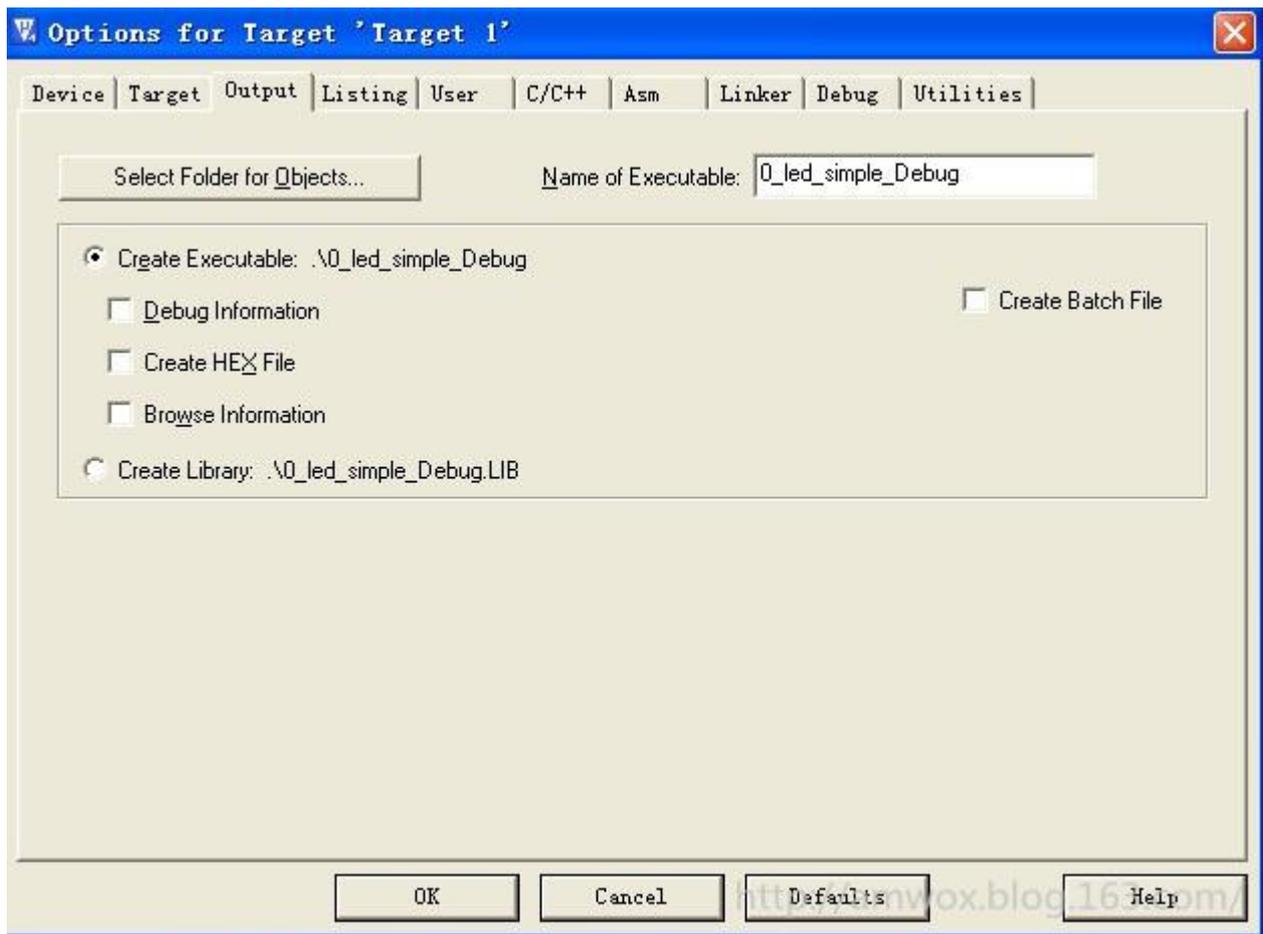
- f Programming.UVPROJ specifies the project to program to flash.
- o Prg\_Output.txt specifies the file to contain the flash program

```
Load "Flash\Blinky.hex"
Erase Done.
Programming Done.
Verify OK.
Application running ...
```

在工程属性中再加多一项

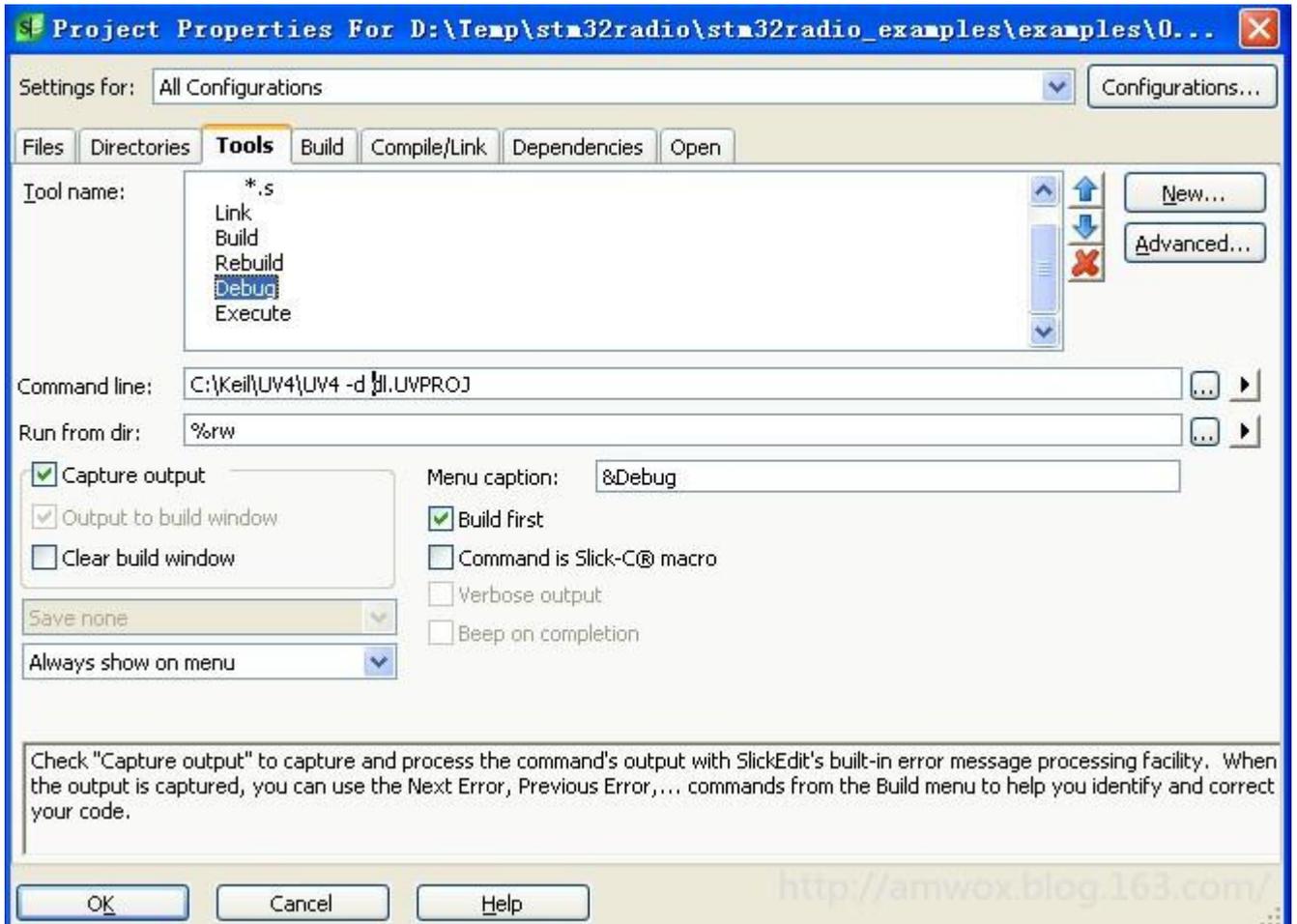


在 KEIL 中新建一个空的工程，不包含任何文件，指定 CPU，调试器（JLINK），配置与下载相关的选项，并指定输出的文件名与生成的 HEX 文件名相同“0\_led\_simple\_Debug”（编译生成的 AXF、HEX 都是相同的文件名）



这样在执行 **Execute** 时可以将程序写入目标板?? 试试吧... 🤖 成功写入, RADIO 板上的灯在闪了 😎

不知道同样的方法也可以进行调试? 再试 🤖



成功启动 KEIL 进入调试模式，并定位在相应源代码中，不是汇编文件！激动啊 🤖

打开相应的文件，设置断点，全速执行，单步执行，全通过！

试用完毕！