

案例“beep”文档

2024 年 9 月

PLAN:

使用 STM32cubeIDE 编程环境，在“正点原子 Nano STM32F411RCT6 开发板”上实现蜂鸣器控制。

DO:

1. 复制案例“led_v0.4”项目文件夹，并将文件夹重命名为“beep”。找到“beep”文件夹中的“led_v0.4.ioc”文件，如图所示：

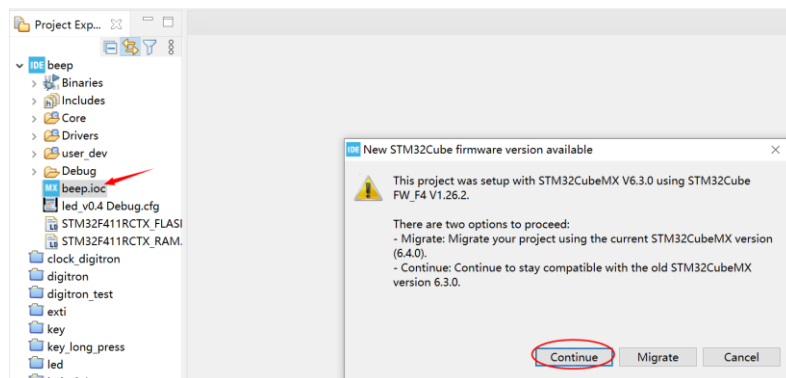
MX led_v0.4	2022/11/13 13:20	STM32CubeMX	6 KB
-------------	------------------	-------------	------

将该文件重命名为“beep.ioc”，如图所示：


MX beep	2022/11/13 13:20	STM32CubeMX	6 KB
---------	------------------	-------------	------

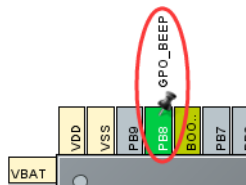
然后，再删除“beep”文件夹中“led_v0.4 Debug.launch”和“led_v0.4 Debug.cfg”两个文件；最后，使用文本编辑器打开“.project”文件，将其中的第三行“<name> led_v0.4 </name>”改为“<name>beep</name>”。

2. 双击“.project”文件，在 STM32CubeIDE 中打开该项目。双击“beep.ioc”文件，对该项目进行硬件配置，如图所示。



等待 IOC 文件加载。加载完成后，参考下图，在原有的硬件配置基础上，再将 PB8 引脚设置为

“GPIO_Output”，并添加标签为“GPO_BEEP”，使用  工具，重新生成代码。

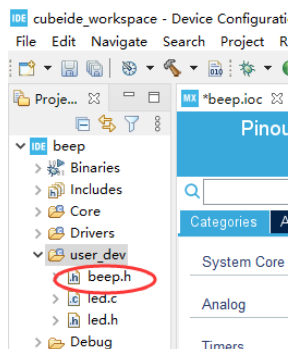


3. 添加“beep.h”文件。将案例 beep 中“user_dev”文件夹下的“beep.h”文件，复制到目标路径，结果如下图。

File Explorer view showing the contents of the 'user_dev' folder:

名称	修改日期	类型	大小
beep	2021/8/23 20:24	H 文件	1 KB
led	2021/8/23 23:44	C 文件	4 KB
led	2021/8/23 16:29	H 文件	1 KB

然后回到 STM32CubeIDE，重新编译。结果如下图所示，beep.h 文件就添加到该项目中了。



4. 打开“beep.h”文件，学习.h 文件的编写，HAL_GPIO_WritePin 函数的使用。

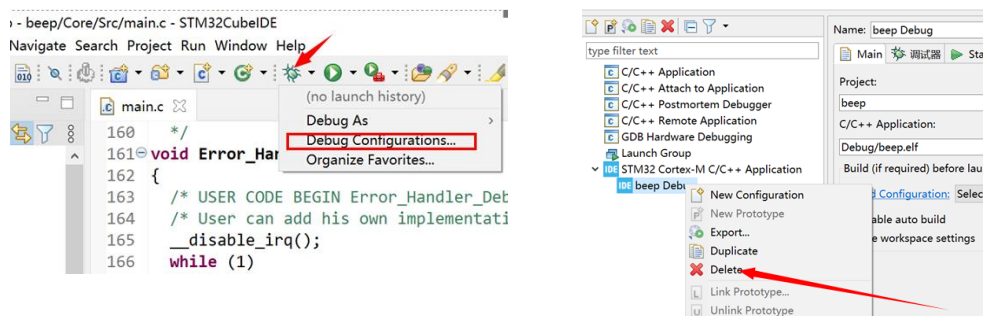
```
1 /**
2  * @file beep.h
3  * @brief 蜂鸣器驱动程序
4  * @author 王晓荣
5  * @version
6  * @date 2020-08-21
7  */
8
9 #ifndef __BEEP_H
10 #define __BEEP_H
11
12 #include "main.h"
13
14 #define beep_on() HAL_GPIO_WritePin(GPIO_BEEP_GPIO_Port, GPIO_BEEP_Pin, GPIO_PIN_RESET)
15 #define beep_off() HAL_GPIO_WritePin(GPIO_BEEP_GPIO_Port, GPIO_BEEP_Pin, GPIO_PIN_SET)
16
17 #endif
```

5. 参考下图，对原来的“main.c”进行 2 处修改。注释或取消注释语句的方法：同时按住“ctrl”和“/”按键。

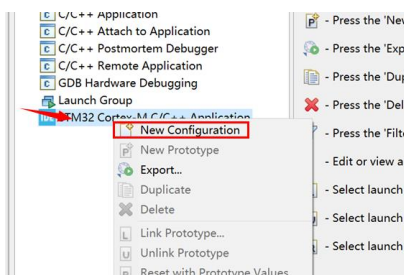
```
main.c
1  /* USER CODE BEGIN Includes */
2  #include "led.h"
3  #include "beep.h"
4  /* USER CODE END Includes */
5
6  /* USER CODE BEGIN WHILE */
7  while (1)
8  {
9      // led_toggle(LED_0);
10     beep_on();
11     HAL_Delay(1);
12     beep_off();
13     HAL_Delay(1000);
14 }
15 /* USER CODE END WHILE */
16
17 /* USER CODE BEGIN 3 */
18
```

6. 编译程序，查看编译结果，直至编译正确无误。

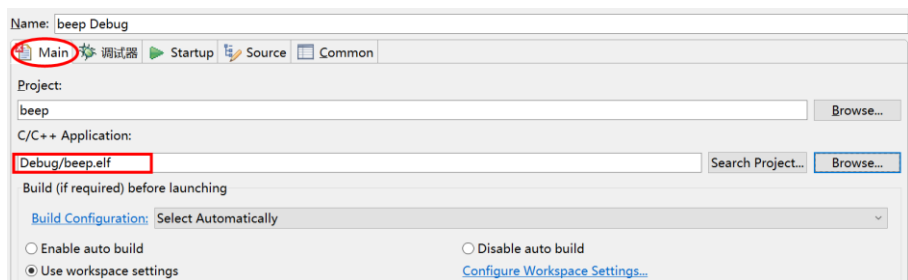
7. (1) 创建调试下载文件。如果已经存在 debug 配置文件，就先删除该文件，操作顺序如下图。然后再创建一个新的配置文件。



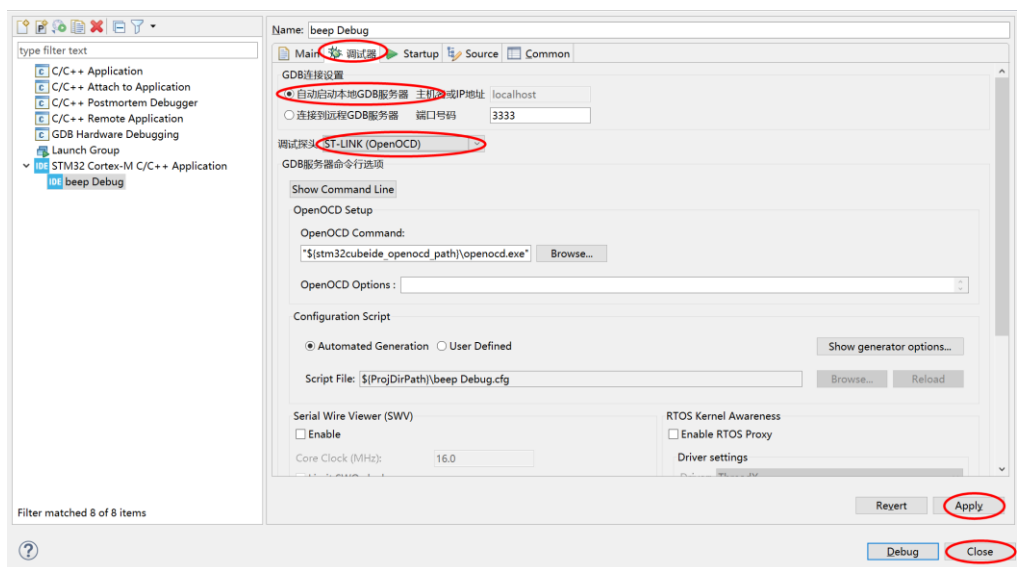
如果没有 debug 配置文件，就直接创建一个新的配置文件，创建方法如下图。



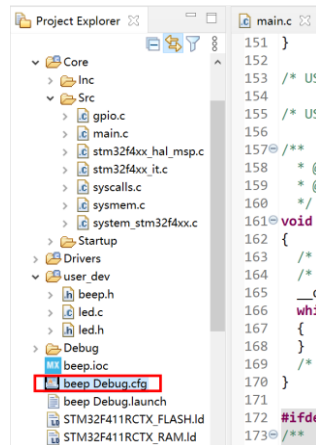
- (2) 新的 beep Debug 中，“Main”选项卡，如果“C/C++ Application:”处为空的话，输入“Debug/beep.elf”，如下图所示。



- (3) 在“调试器”选项卡中，按照下图操作，生成配置文件。



(4) 在 STM32CubeIDE 界面左侧，能看到刚刚生成的 beep Debug.cfg 调试器配置文件，存放位置如图所示。



(5) 在本项目的存放路径，找到 beep Debug.cfg。复制或另存该文件，为一个新的 .cfg 文件，后缀保持不变，可在文件命名时加上 DAP 以示区分，如下图所示。切记一定要复制或另存！否则无法烧录。

Data (D:) > beep >				Data (D:) > beep >			
名称	修改日期	类型	大小	名称	修改日期	类型	大小
.settings	2023/11/11 21:11	文件夹		.settings	2023/11/11 21:11	文件夹	
Core	2023/11/11 21:11	文件夹		Core	2023/11/11 21:11	文件夹	
Debug	2023/11/11 21:46	文件夹		Debug	2023/11/11 21:46	文件夹	
Drivers	2023/11/11 21:11	文件夹		Drivers	2023/11/11 21:11	文件夹	
user_dev	2023/11/11 21:34	文件夹		user_dev	2023/11/11 21:34	文件夹	
.cproject	2023/11/11 21:30	CPROJECT 文件	26 KB	.cproject	2023/11/11 21:30	CPROJECT 文件	26 KB
.mxproject	2023/11/11 21:30	MXPROJECT 文件	7 KB	.mxproject	2023/11/11 21:30	MXPROJECT 文件	7 KB
.project	2023/11/11 21:12	PROJECT 文件	2 KB	.project	2023/11/11 21:12	PROJECT 文件	2 KB
beep Debug	2023/11/11 22:39	CFG 文件	1 KB	beep Debug - DAP	2023/11/11 22:39	CFG 文件	1 KB
beep Debug.launch	2023/11/11 22:39	LAUNCH 文件	9 KB	beep Debug	2023/11/11 22:39	CFG 文件	1 KB
beep	2023/11/11 21:42	STM32CubeMX	6 KB	beep Debug.launch	2023/11/11 22:39	LAUNCH 文件	9 KB
STM32F411RCTX_FLASH.ld	2023/11/11 21:30	LD 文件	5 KB	beep	2023/11/11 21:42	STM32CubeMX	6 KB
STM32F411RCTX_RAM.ld	2021/8/23 13:09	LD 文件	5 KB	STM32F411RCTX_FLASH.ld	2023/11/11 21:30	LD 文件	5 KB
				STM32F411RCTX_RAM.ld	2021/8/23 13:09	LD 文件	5 KB

(6) 用 notepad++ 或者 vscode 等文本工具软件，打开刚刚复制并重命名的 beep Debug-DAP.cfg 文件，也可以双击打开“beep Debug-DAP.cfg”文件，按照下图所示，进行修改，然后保存退出，这样操作就可以支持 DAP 下载模式了。

```
# This is an genericBoard board with a single STM32F411RCTx chip
#
# Generated by STM32CubeIDE
# Take care that such file, as generated, may be overridden without any early
notice. Please have a look to debug launch configuration setup(s)

source [find interface:stlink-dap.cfg]  将此改为 cmsis-dap.cfg

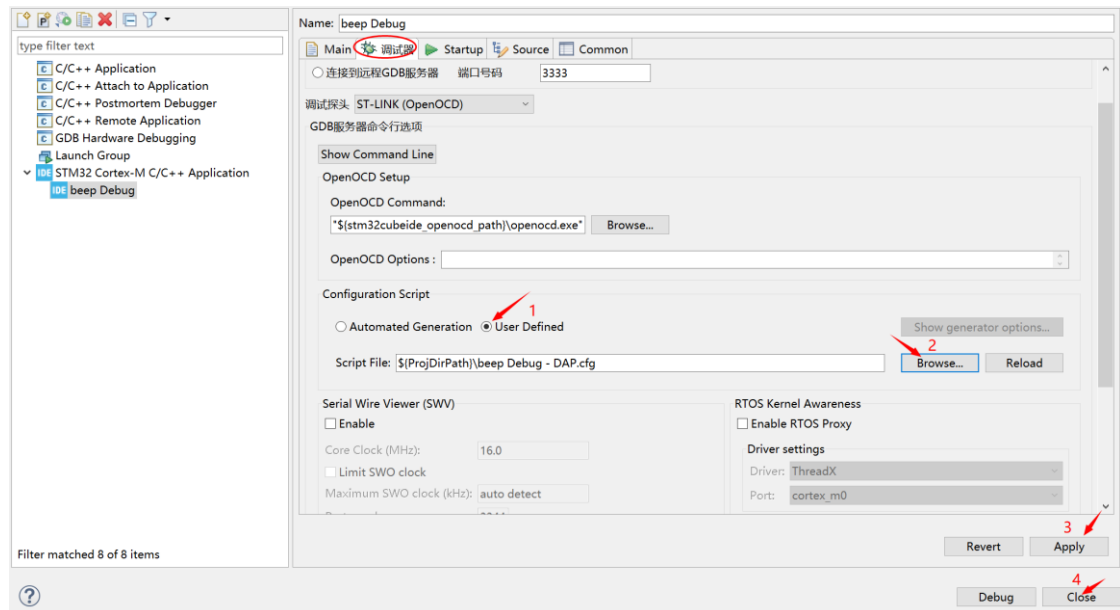
set WORKAREASIZE 0x8000


transport select "dapdirect_swd"  将此改为 swd

set CHIPNAME STM32F411RCTx
set BOARDNAME genericBoard

# Enable debug when in low power modes
set ENABLE_LOW_POWER 1
```

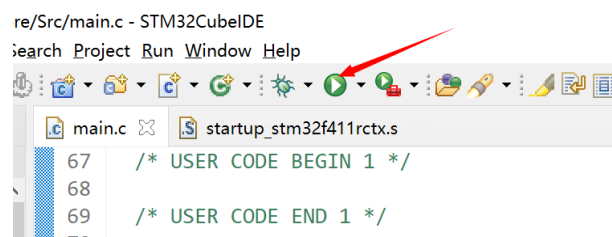
(7) 然后，再次进入 debug Configurations 界面，按照下图中标注的顺序依次设置。**重点：把刚才修改的“beep Debug-DAP.cfg”文件添加进来**，整个设置工作就做完了。



(8) 然后再点击  下载工具，OK，烧录成功，**板子上的蜂鸣器发出声音**。如果蜂鸣器发出短促的声音，尝试调整 beep_on()函数的延时时间，观察效果。

CHECK:

1. 按照“DO”中步骤 1 复制项目，否则会出错。
2. 如果不是新建案例，而是打开已经存在的“beep”案例，存在无法下载的情况，需要按照“DO”中步骤 7 进行操作。
3. 如果仍然无法正常下载，再点击一次工具栏上的运行按钮，如图所示。



ACTION:

1. 实际工程项目中，蜂鸣器一般只有 1 个，因此驱动模块仅用 1 个.h 文件，可以无需.c 文件。