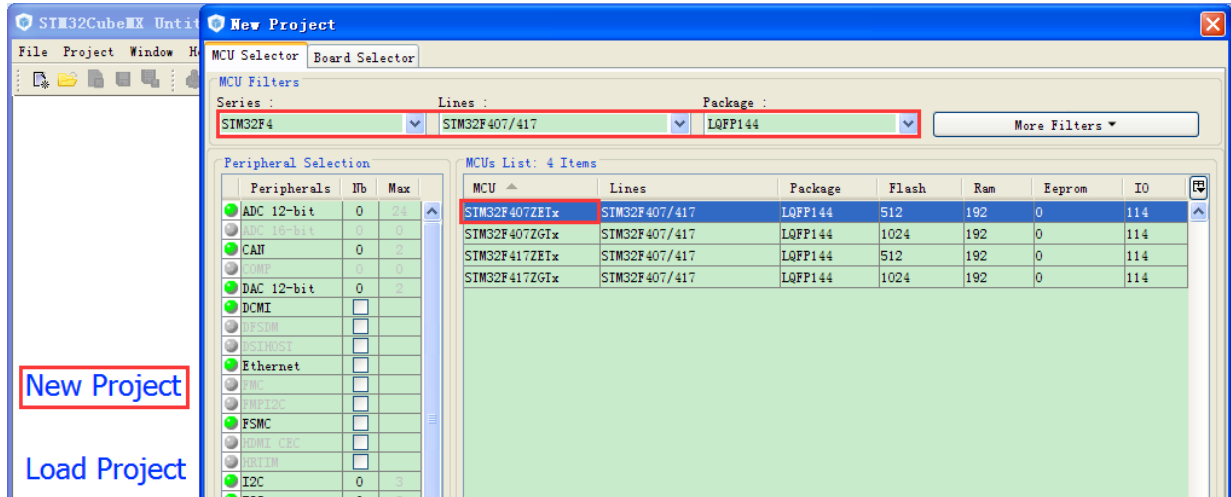


STM32Cube 学习之十一：DMA

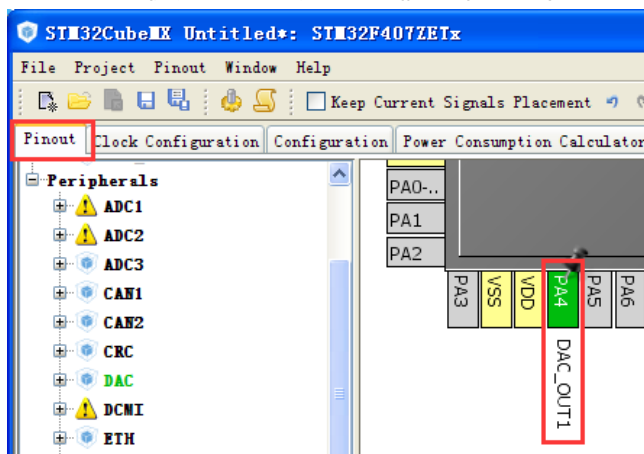
假设已经安装好 STM32CubeMX 和 STM32CubeF4 支持包。

Step1. 打开 STM32CubeMX，点击 “New Project”，选择芯片型号，STM32F407ZETx。

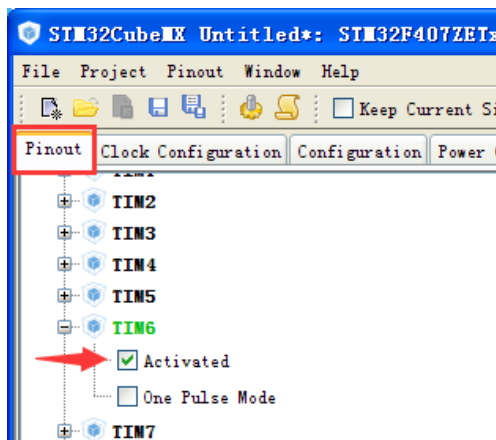


Step2. 在 Pinout 界面下配置引脚功能。

DAC：根据电路使用 PA4 作为 DAC 输出，因此将 PA4 的功能配置为 DAC_OUT1。

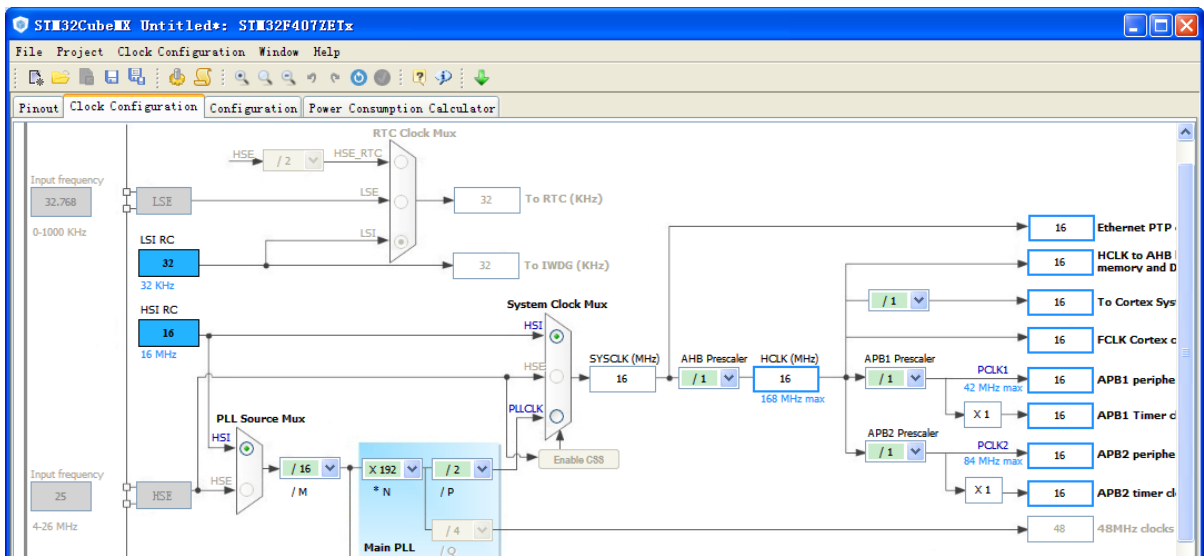


TIM6：激活 TIM6。



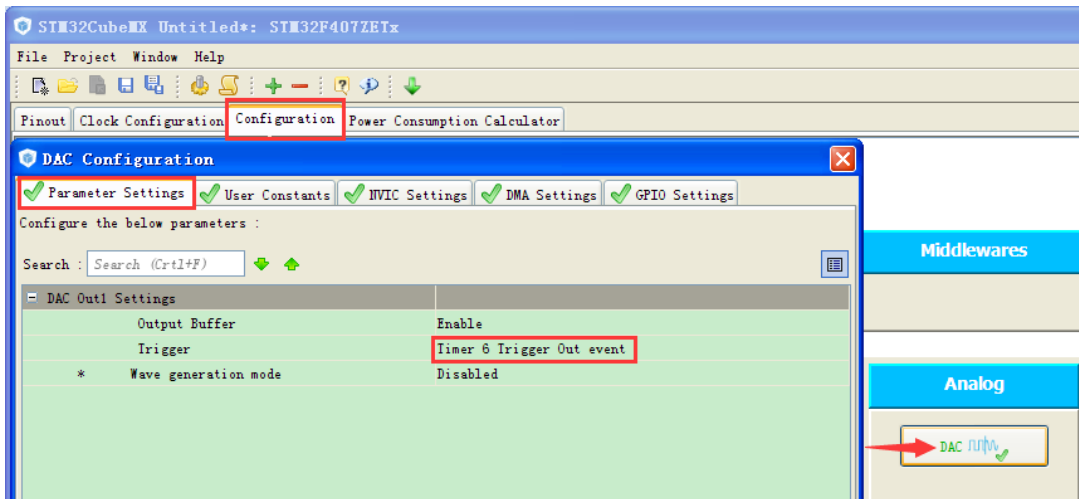
Step3. 在 Clock Configuration 界面配置时钟源。

配置时钟树，在此使用默认值，内部 16MHz。

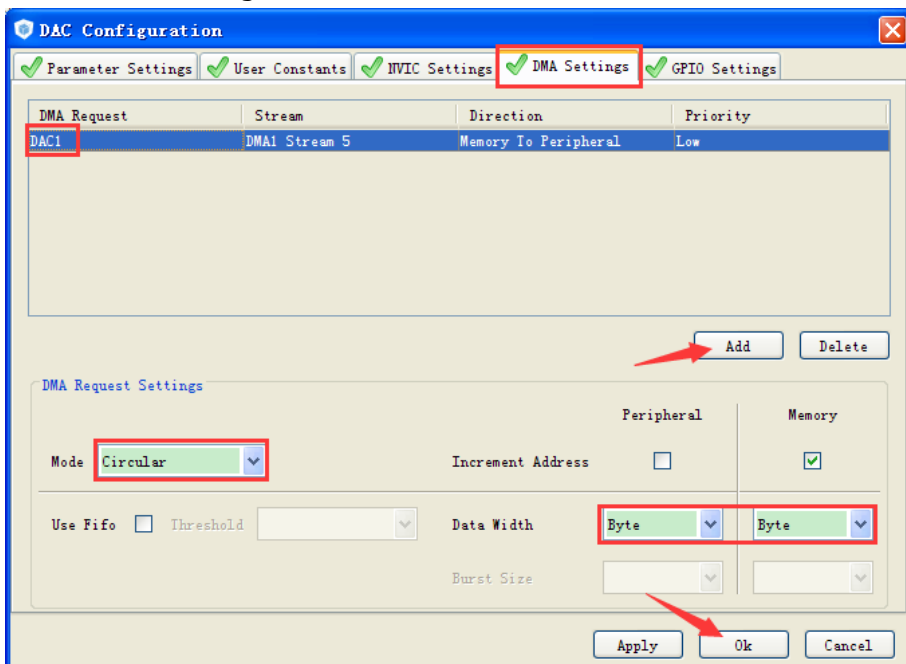


Step4.配置外设参数。

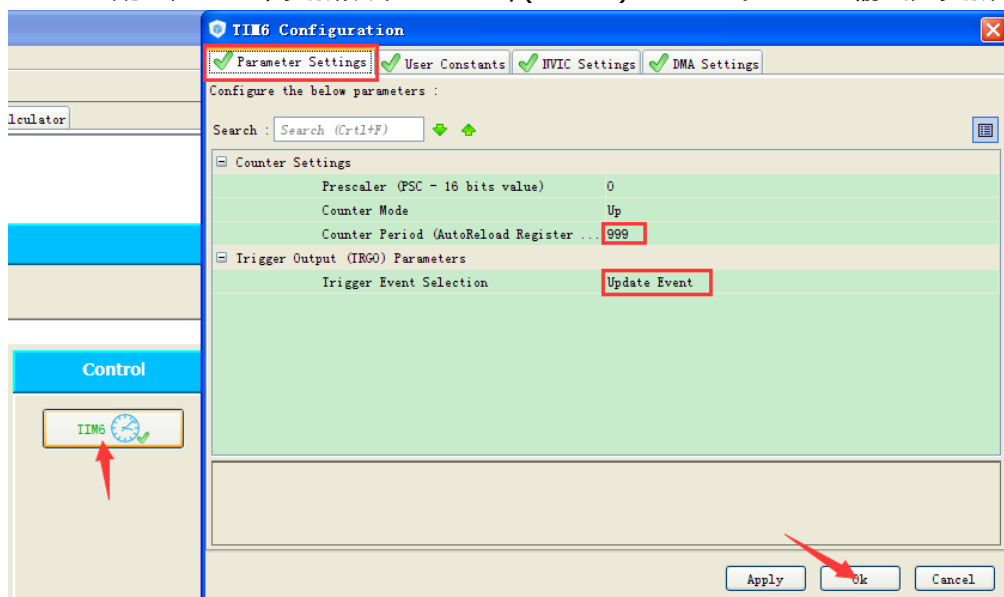
DAC：配置触发源为 TIM6 的触发输出事件。



切换到 DMA Settings 页面，添加 DAC1 的 DMA 功能，Circular 模式、字宽 Byte。

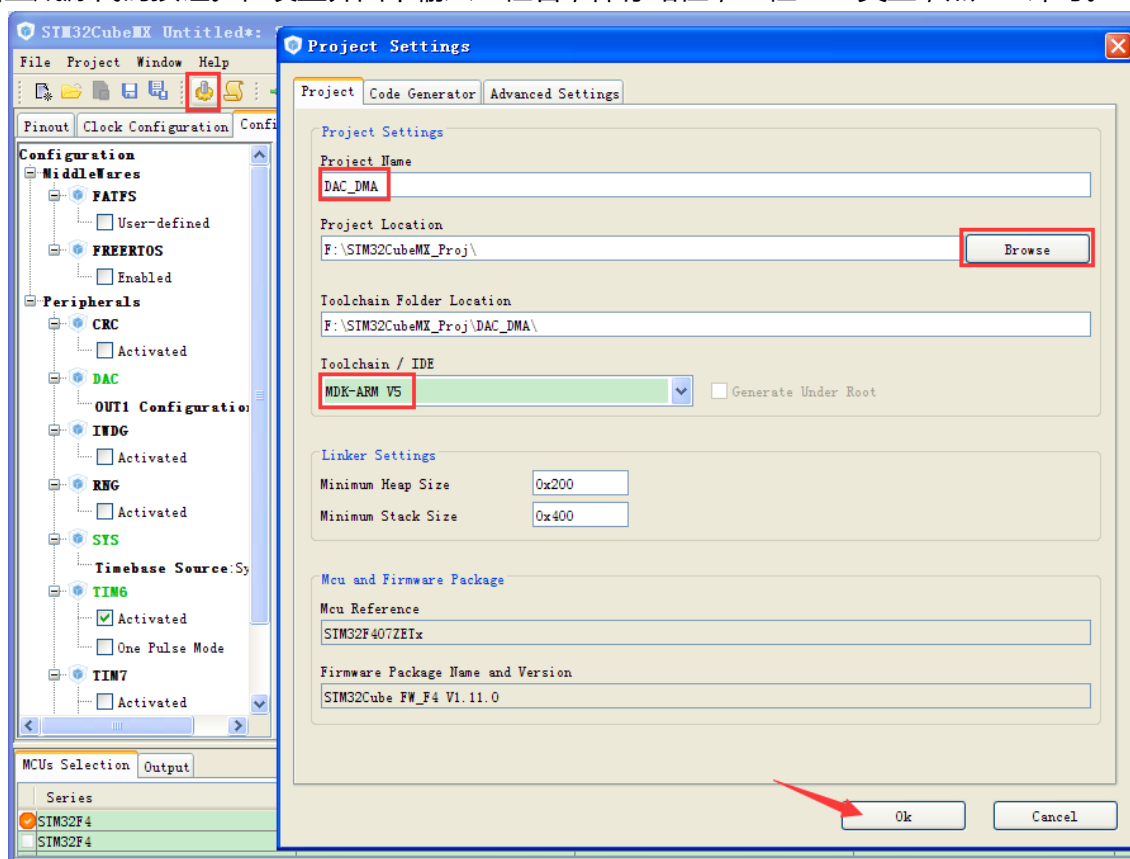


TIM6：ARR 配置为 999，更新频率=16MHz/(999+1)=16kHz。TRGO 输出为更新事件。

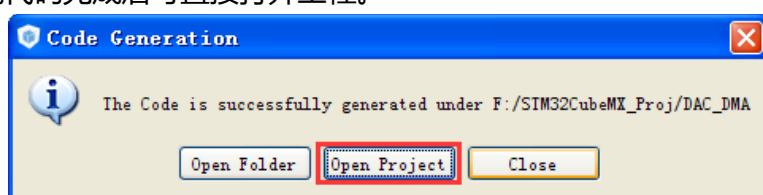


Step5.生成源代码。

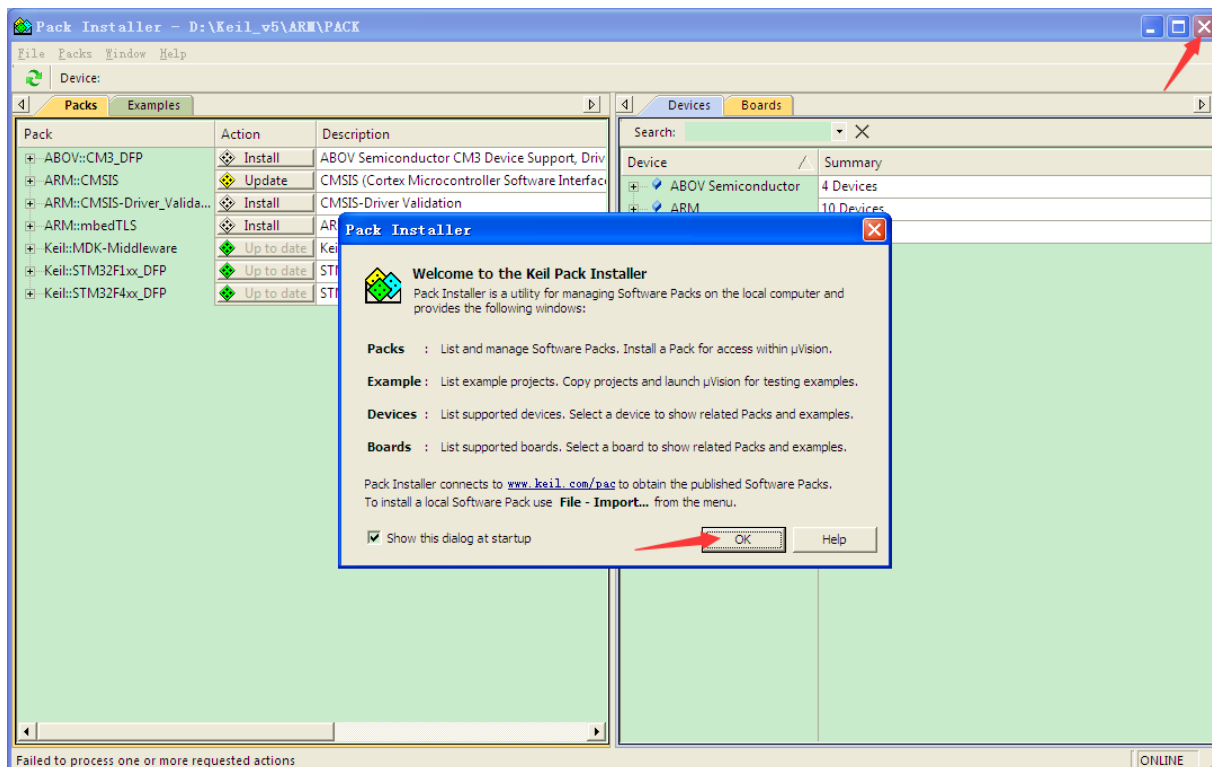
点击生成源代码按钮。在设置界面中输入工程名，保存路径，工程 IDE 类型，点 OK 即可。



生成代码完成后可直接打开工程。



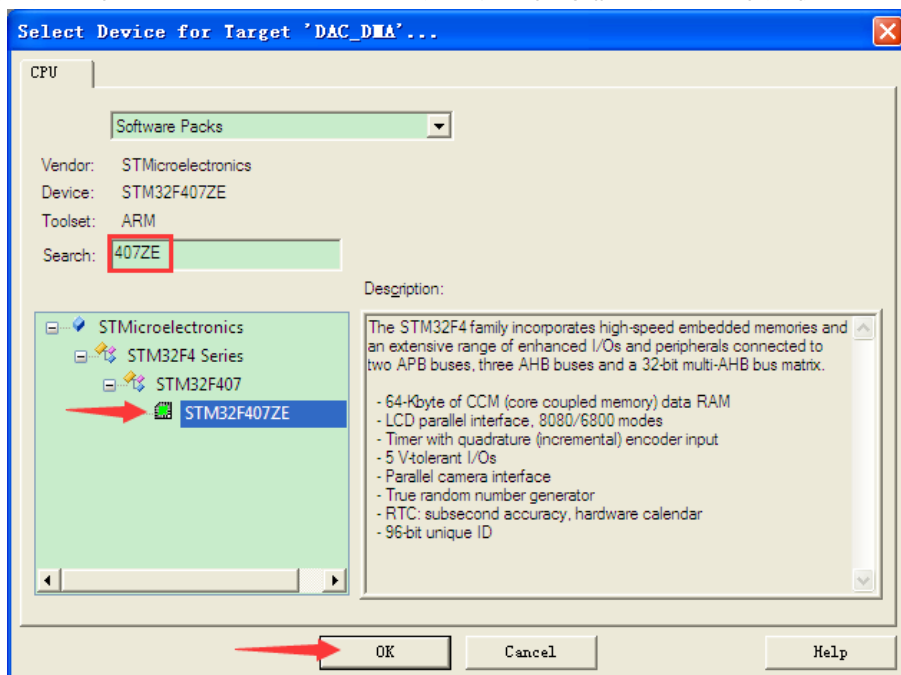
弹出如下对话框时，如果已经安装了 F4 的支持包，则点击 OK 关闭。如果没有安装，则点击界面中的 www.keil.com/... 链接，找到芯片的支持包，然后安装。



关闭后面的界面。



点击“是”，然后选择芯片型号。可以在搜索框中输入关键字，加快选择速度。



Step6.添加功能代码。

在主函数的用户代码区 0，定义一个数组。注意，数组类型为 uint8_t，原因是 Step4 设置的 DMA 传输字宽为 Byte。

```
63  /* USER CODE BEGIN 0 */
64  const uint8_t data_8bit[8] = {0x00, 0x20, 0x40, 0x60, 0x80, 0xA0, 0xC0, 0xE0};
65  /* USER CODE END 0 */
```

在用户代码区 2，使能 TIM6，启动 DAC 转换并使能 DMA 功能。

```
88  /* USER CODE BEGIN 2 */
89  HAL_TIM_Base_Start(&htim6);
90  HAL_DAC_Start_DMA(&hdac, DAC_CHANNEL_1, (uint32_t*)data_8bit, 8, DAC_ALIGN_8B_R);
91  /* USER CODE END 2 */
```

至此，工程完成。运行结果是在 PA4 引脚输出一个阶梯波。

本例演示的是外设 DMA 功能的使用方法。

基本流程是：

- 1.用 CubeMX 配置外设的 DMA 功能；
- 2.如果使用的触发源也是偏上外设(如 TIM)，则配置该外设；
- 3.使能触发源；
- 4.使能外设的 DMA 功能。

对于外设 DMA 功能，HAL 库都提供了相应的函数，形如 HAL_DAC_Start_DMA()，HAL_DAC_Stop_DMA()等。

官方例程请参考 stm32cubef4.zip 解压后

STM32Cube_FW_F4_V1.9.0\Projects\STM324xG_EVAL\Examples\DAC\DAC_SignalsGeneration 目录下的工程。

