

基于CubeMX 5.6.1的TouchGFX移植教程

笔记本：TouchGFX

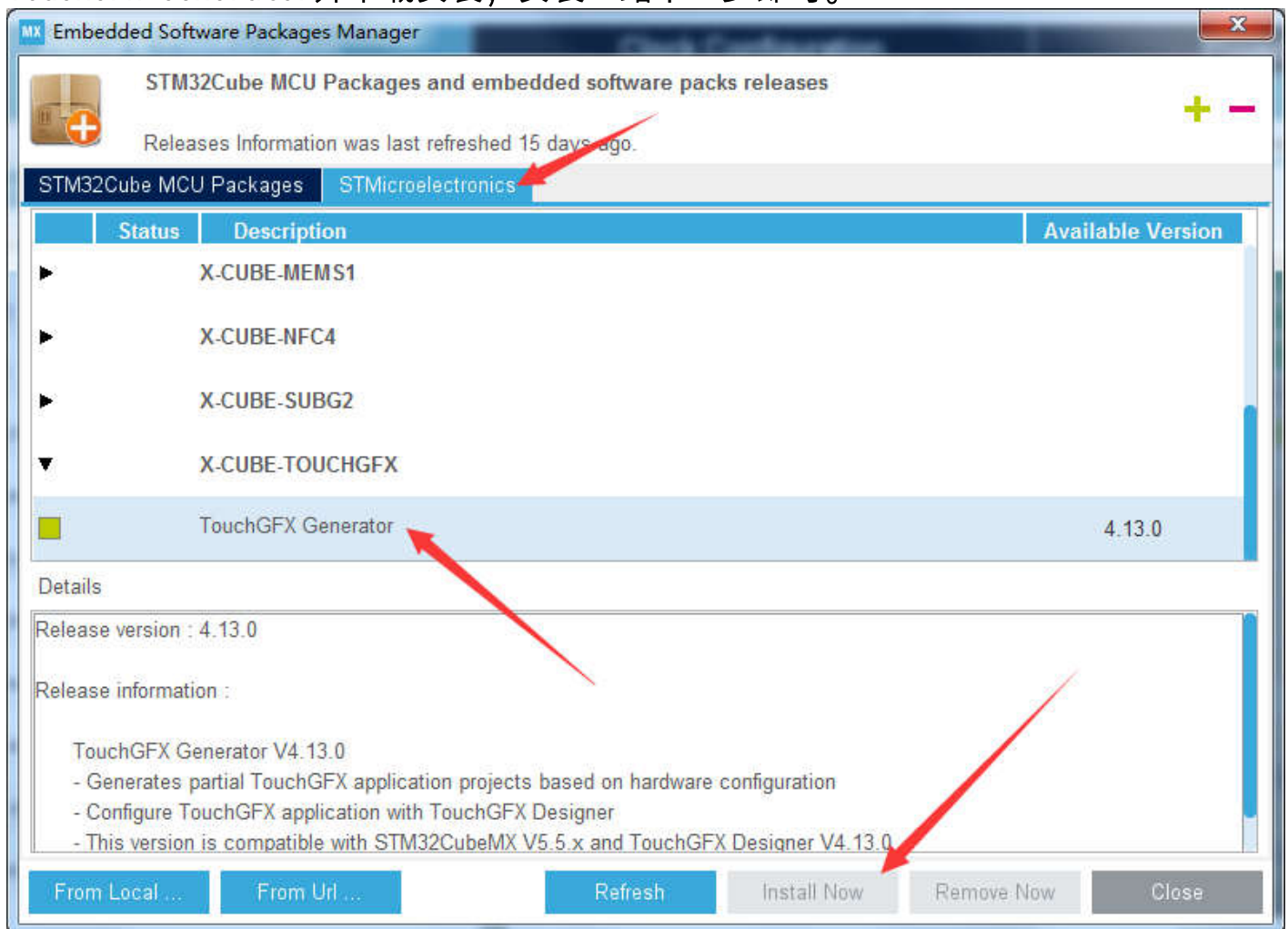
创建时间：2020/5/14 22:33

更新时间：2020/5/14 23:58

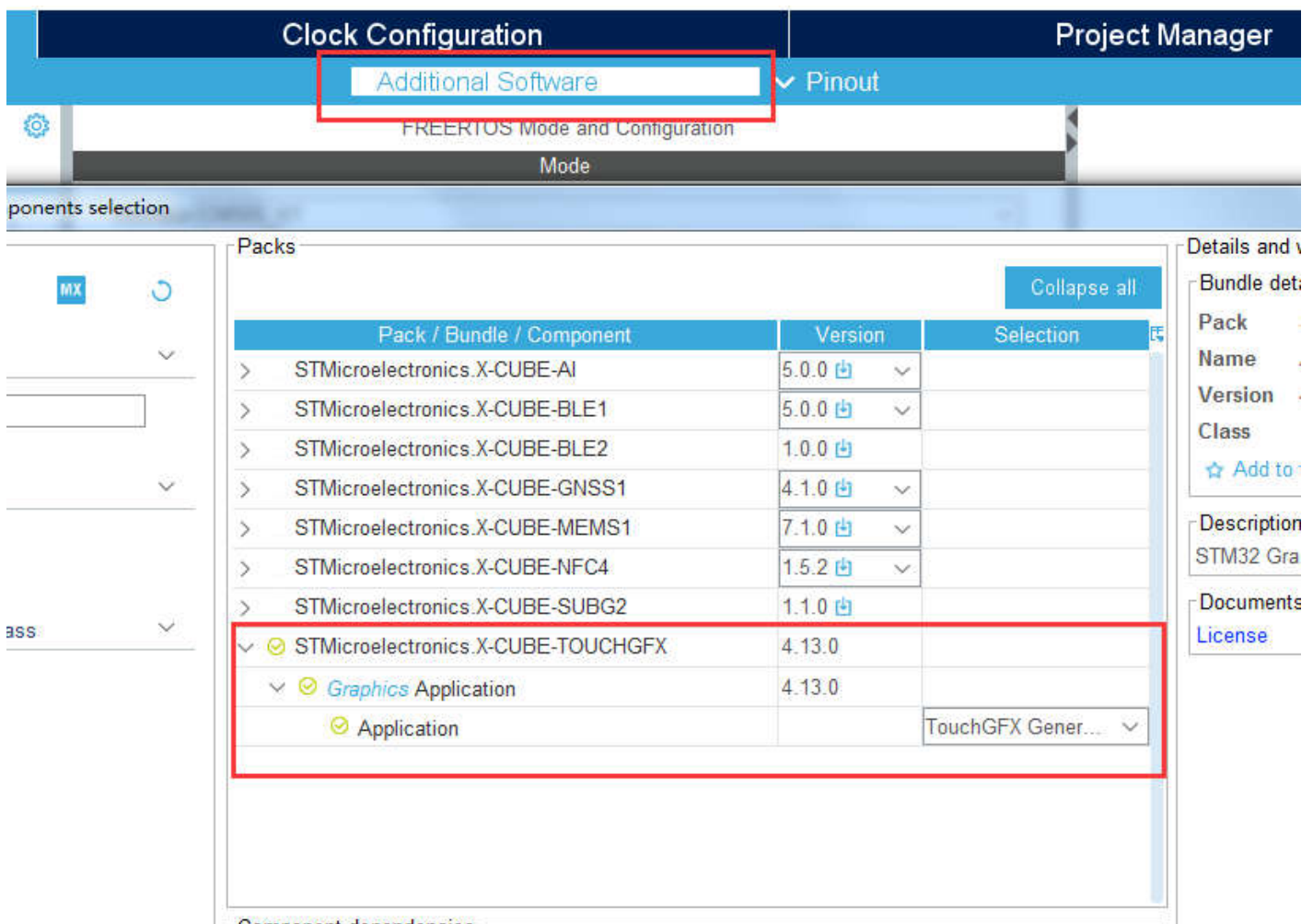
作者：阿里

准备工作：CubeMX升级到5.6.1

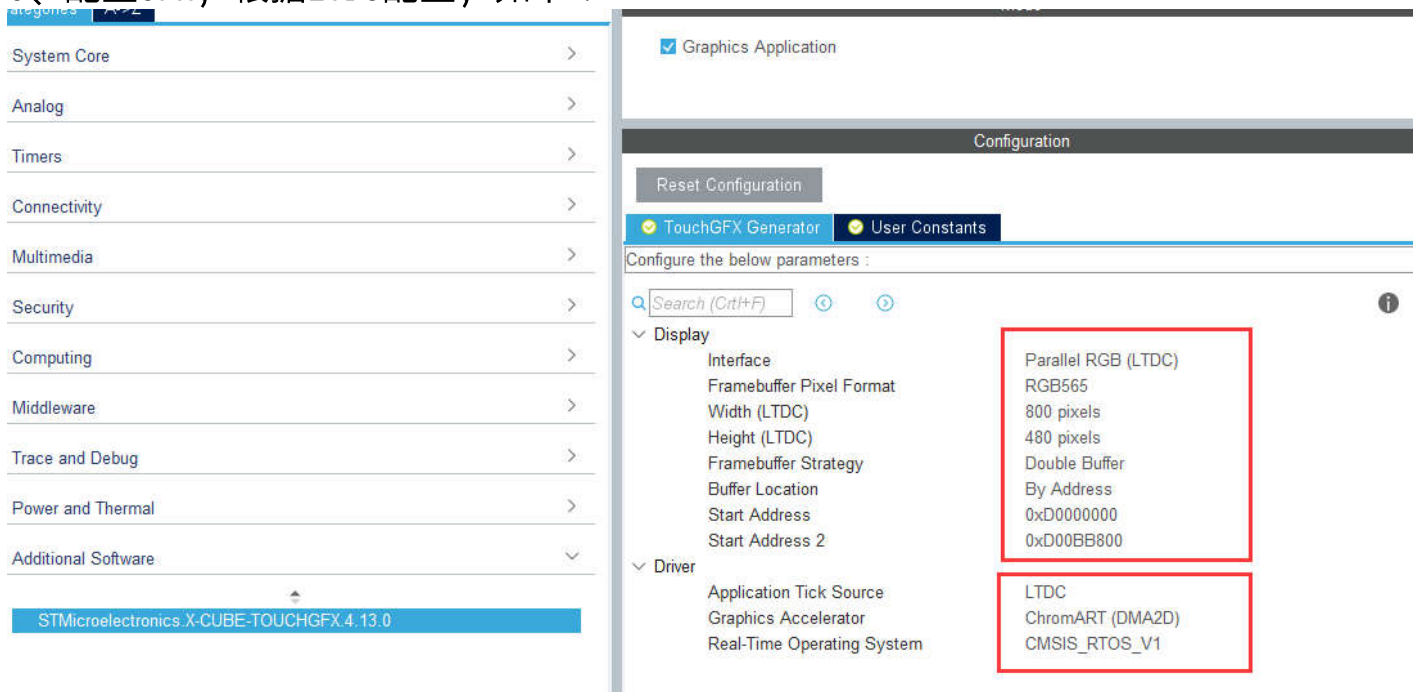
1、安装TouchGFX Designer 4.13，打开CubeMX，打开Help→Embedded Software Packages Manager→切换到STMicroelectronics，拉到X-CUBE-TOUCHGFX，勾上TouchGFX Generator并下载安装，安装一路下一步即可。



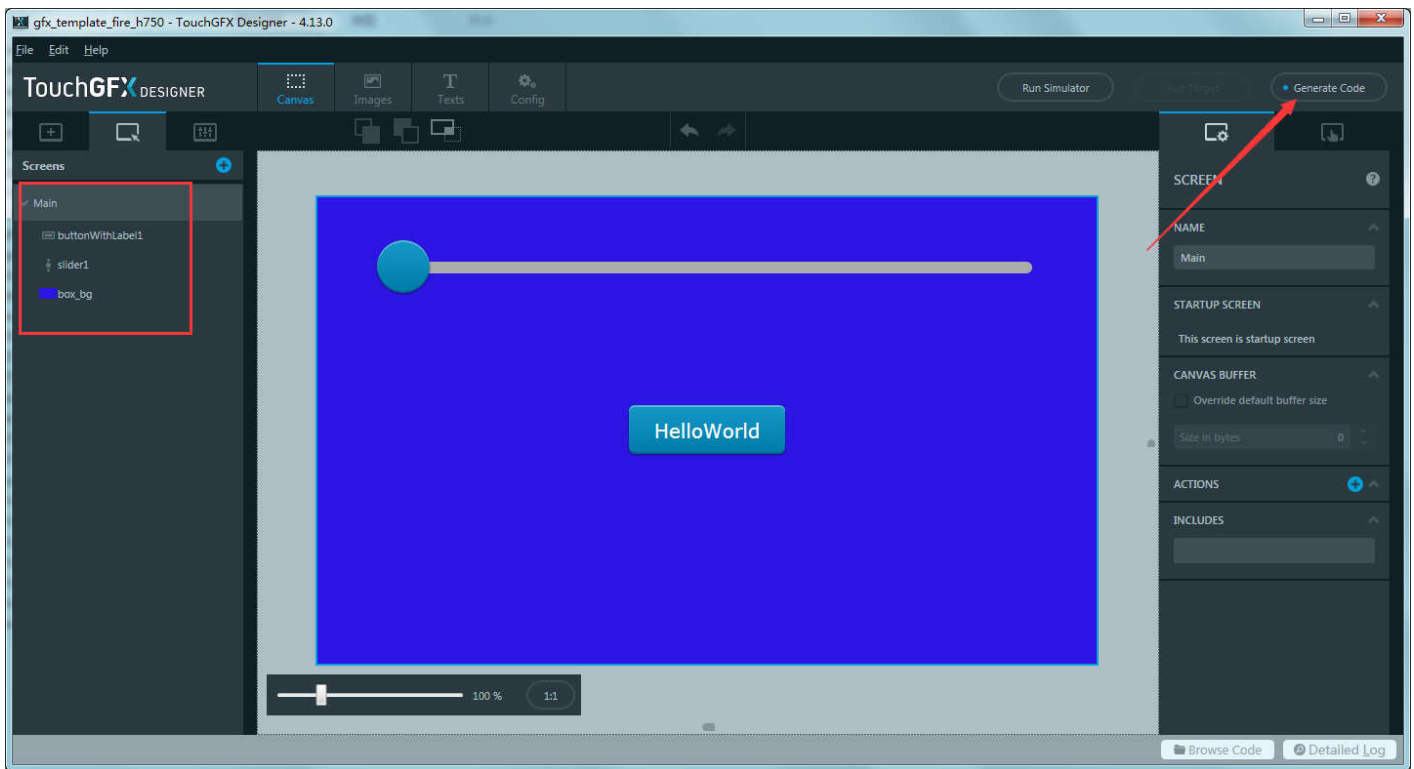
2、打开CUBEMX新建GFX工程，需要配置的有：打开CRC，勾选DEBUG方式，配置SDRAM，配置LTDC与DMA2D，注意GFX只支持RGB565或RGB888，配置LTDC的Layer层时需要注意，勾选LTDC与DMA2D的中断，加入FreeRTOS，其中GFX所在的任务内存需配置大一点，512Words以上，添加GFX，如找不到可按如下方式添加。



3、配置GFX，根据LTDC配置，如下：



4、生成工程，再新建GFX界面，找到Src文件下的 ApplicationTemplate.touchgfx.part，并双击打开，新建一个界面打生成代码 Generate Code：



5、修改工程，在freertos.c的GUI任务加入MX_TouchGFX_Process即可：

```

101  * @brief Function implementing the taskGui thread.
102  * @param argument: Not used
103  * @retval None
104  */
105  /* USER CODE END Header_StartTaskGui */
106  void StartTaskGui(void const * argument)
107  {
108      /* USER CODE BEGIN StartTaskGui */
109      MX_TouchGFX_Process();
110      /* infinite loop */
111      for(;;)
112      {
113          osDelay(1);
114      }
115      /* USER CODE END StartTaskGui */
116  }
117
118  /* Private application code -----

```

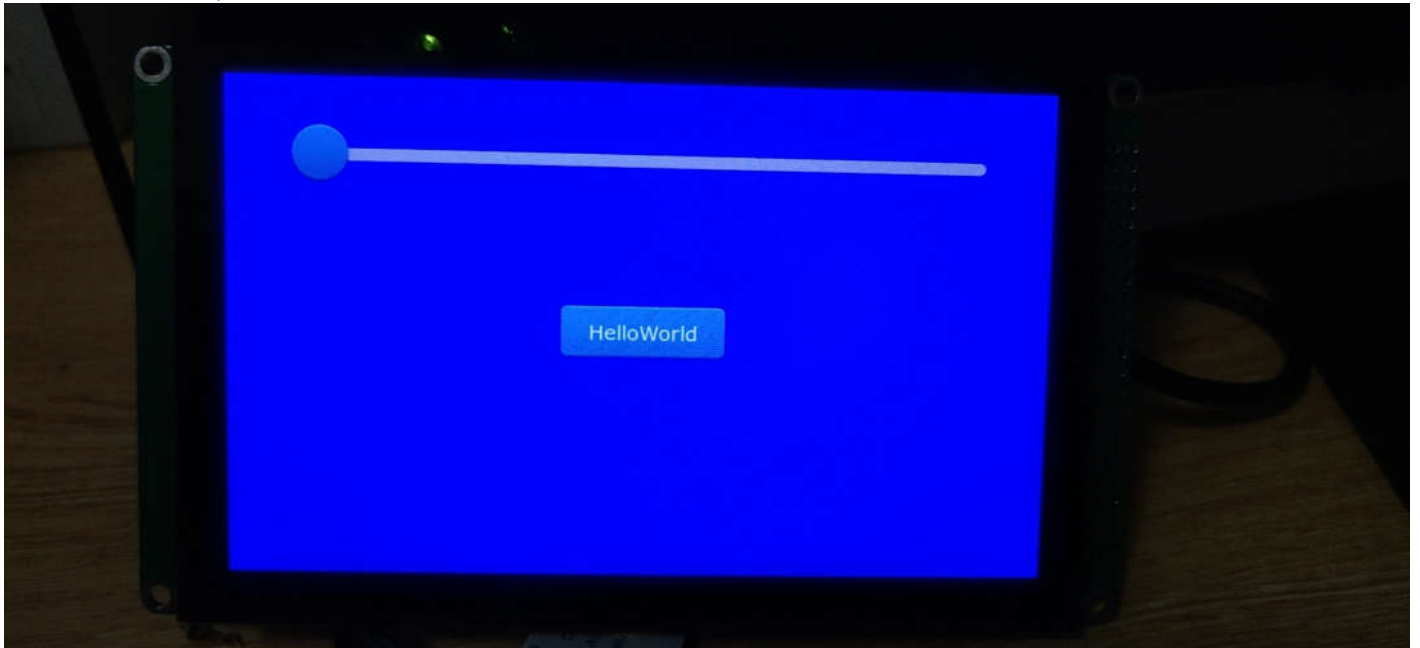
6、如果开启了DCache，需要修改4个地方，在STM32DMA.cpp有3处，在TouchGFXHAL.cpp有1处，找到SCB_CleanInvalidateDCache();这样的函数，全部取消屏蔽：

```

46
47 static void DMA2D_XferCpltCallback(DMA2D_HandleTypeDef* handle)
48 {
49     /* USER CODE BEGIN DMA2D_XferCpltCallback */
50     // If the framebuffer is placed in Write Through cached memory (e.g.
51     // to flush the Dcache prior to letting DMA2D accessing it. That's d
52     // using SCB_CleanInvalidateDCache().
53
54     // SCB_CleanInvalidateDCache(); 取消屏蔽
55     /* USER CODE END DMA2D_XferCpltCallback */
56
57     touchgfx::HAL::getInstance()->signalDMAInterrupt();
58 }
59

```

7、编译下载，即可看到画面了：



8、触摸移植，只要修改STM32TouchController.cpp文件，可以添加触屏初始化，添加读取坐标函数，成功赋值x y并返回true即可，重新编译下载即可触摸了。

```

#include <STM32TouchController.hpp>
#include "ft5306.h"

void STM32TouchController::init()
{
    /**
     * Initialize touch controller and driver
     *
     */
    FT5306_Init();
}

bool STM32TouchController::sampleTouch(int32_t& x, int32_t& y)
{
    /**
     * By default sampleTouch returns false,
     * return true if a touch has been detected, otherwise false.
     *
     * Coordinates are passed to the caller by reference by x and y.
     *
     * This function is called by the TouchGFX framework.
     * By default sampleTouch is called every tick, this can be adjust
     *
     */
    if ( FT5306_Read_TouchPoint() != 0 )//读取到如果有触摸
    {
        static uint16_t sx = 0;
        static uint16_t sy = 0;
        FT5306_Read_XY(&sx, &sy);//读取触坐标
        x = sx;
        y = sy;
        return true;
    }
    return false;
}

```

一些注意事件

- 1、新的CUBEMX生成的GFX工程基本移植好了，如编译不通过，注意是否添加头文件。
 - 2、如下载后没有图像，注意是否开启CRC，SDRAM与LTDC配置是否正确，未添加GFX前是否可以正常显示，是否开启了背光，是否打开了LTDC与DMA2D的中断，GFX所在的任务内存是否改大了，如程序运行在QSPI时是否进行了中断偏移SCB→VTOR = QSPI_BASE，如开了DCache是否有启动SCB_CleanInvalidateDCache();
 - 3、如添加触摸后无效，检测触摸驱动是否能读到坐标。
- 关于TouchGFX的定位，个人认为是定位高端的单片机，因为GFX的界面基本都是靠贴图片来设计的，最好是有LTDC和SDRAM加QSPI。