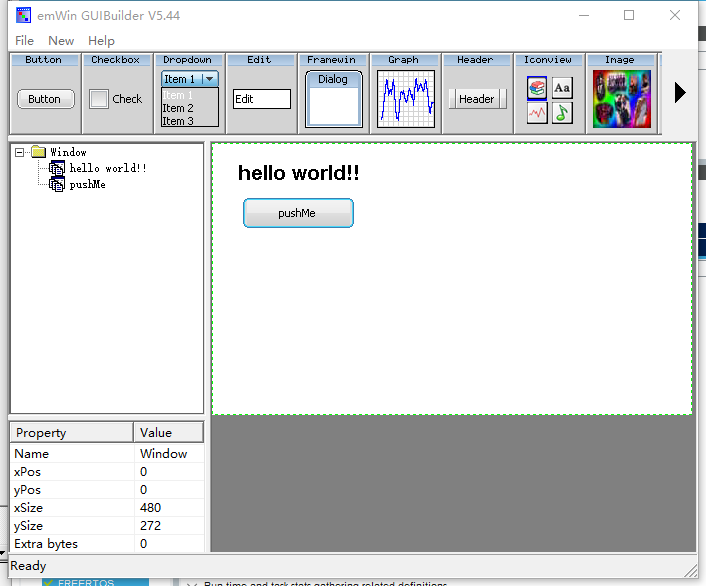
该笔记用于记录将gt911 电容触摸驱动加入freeRTOS管理的emwin GUI项目中

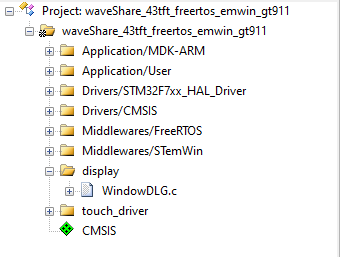
起始时间：2021年3月14日星期日

1. 确定使用guiBuild生成的显示框架可以运行

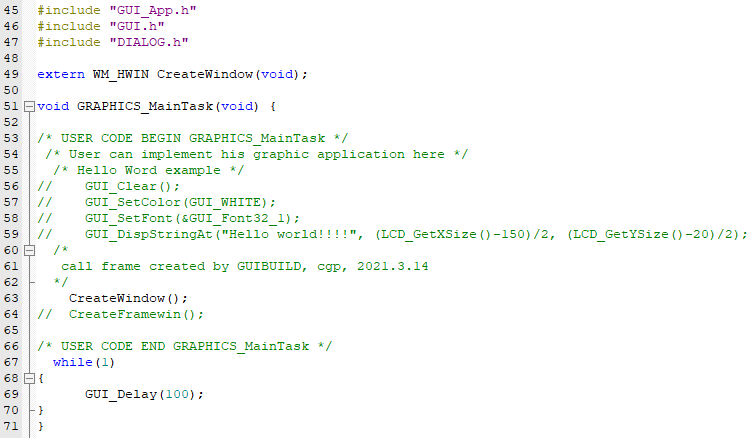


注意：1.使用window类作为顶层显示框架；2. Window的Size 为 480 \* 272

* 1. 将该显示框架生成的WindowDLG.c组织到mdk项目中，如下图示



* 1. 将WindowDLG.c入口函数引入到GUI\_App.c的GRAPHICS\_MainTask函数中



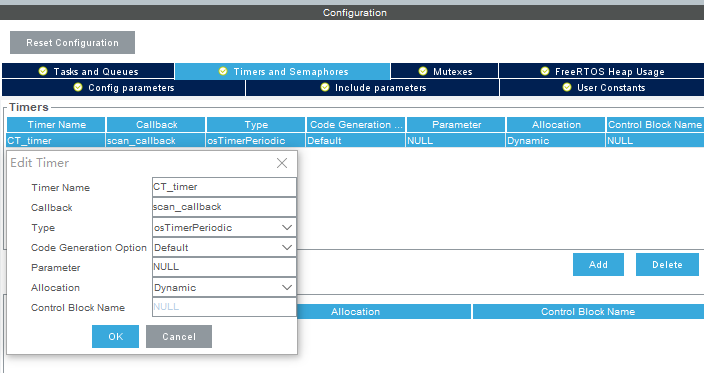
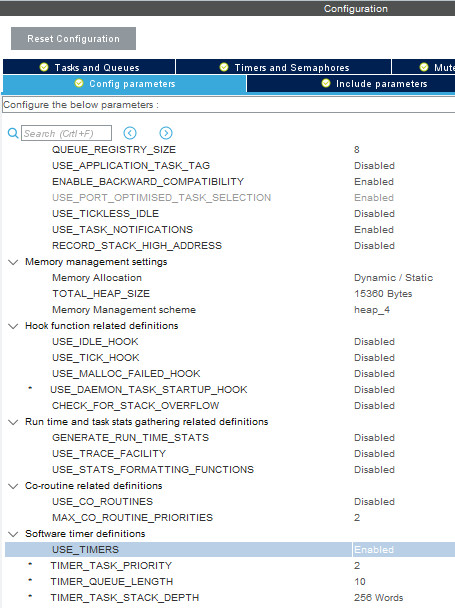
其中，第47，49，63行为新增代码。

1. 增加用于gt911定时扫描的软件定时器

原理上，需要约20ms调用一次gt911扫描过程，以判断是否有触摸动作。实现方法较多，因为我们以加入freeRTOS作为系统的管理层，因此，使用freeRTOS自己的软件定时器来调用较好。

2.1 开freeRTOS的用户软件定时功能

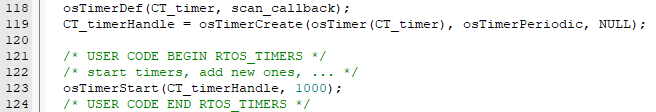
系统缺省该功能进打开，如下图



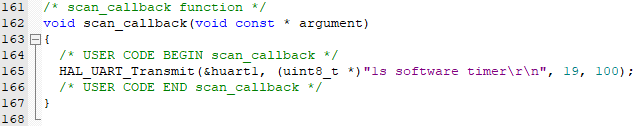
开启软件用户定时器功能 增加一个软件定时器

使用MX生成代码框架。

2.2 编写一个测试代码，在freertos.c代码中



增加123行，启动这个定义的软件定时器，按1000毫秒（1s）周期执行scan\_callback函数



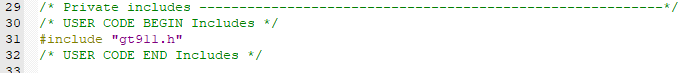
在回调函数中，增加一行调试代码，注意HAL\_UART\_Transmit的第三个参数为输出字符数，应包括\r\n和空格，第四参数为timeOut，可大致取100ms即可。

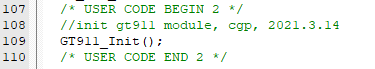
将一个usb-uart的Rx与核心板的PA9相接，确认能正确收到一秒一次的信息。

1. 增加由gt911有触摸动作的PD7中断响应函数

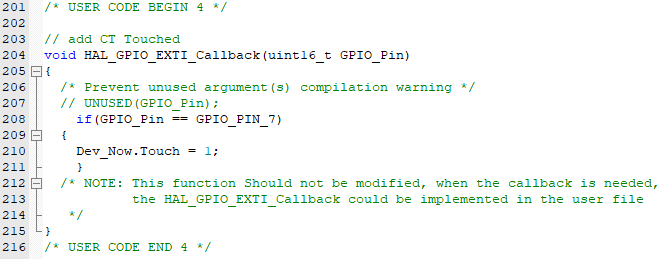
在增加gt911模块前，应先理解gt911工作原理和参考驱动的设计原理。

3.1 在适当的时间点初始化gt911

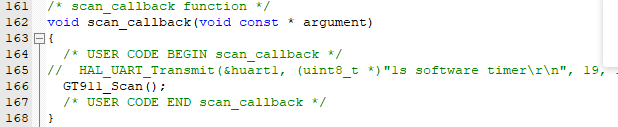




3.2 增加触摸动作的PD7中断响应



1. 调整freeRTOS的20ms软定时回调动作



1. 最后提醒一下，因为gt911的驱动中涉及printf做为调试输出，应对usart.c增加printf输出重定向。

