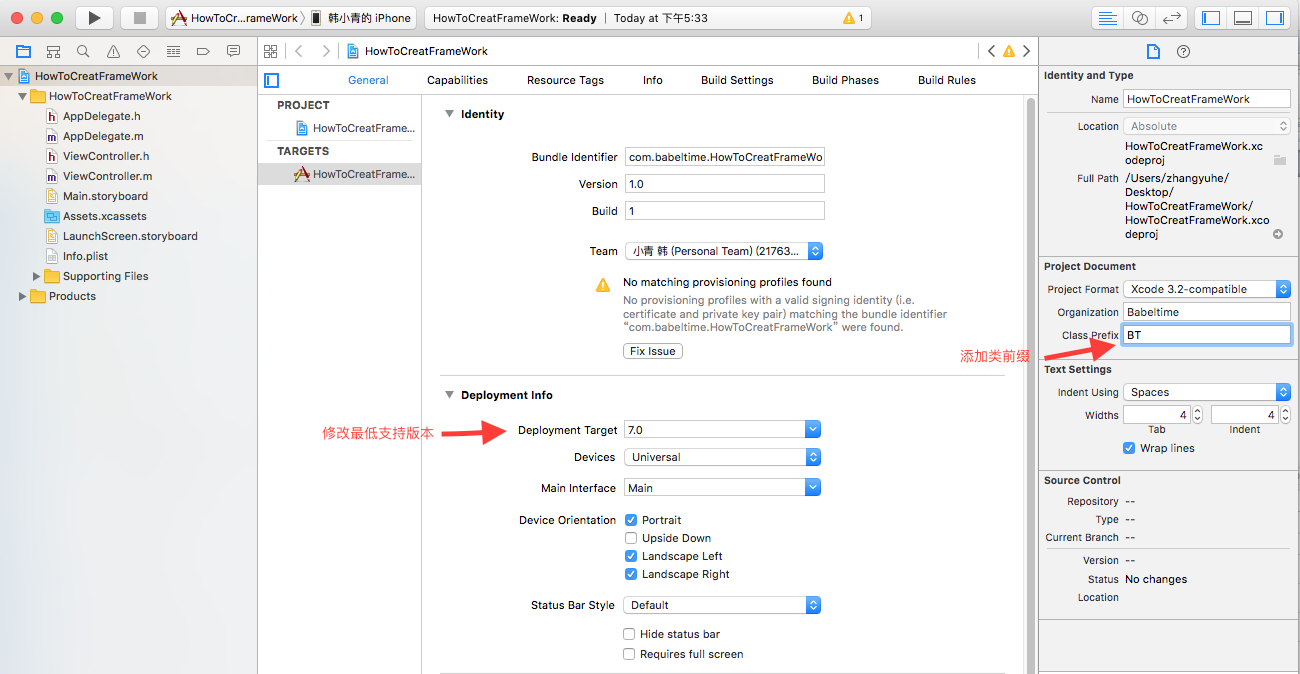
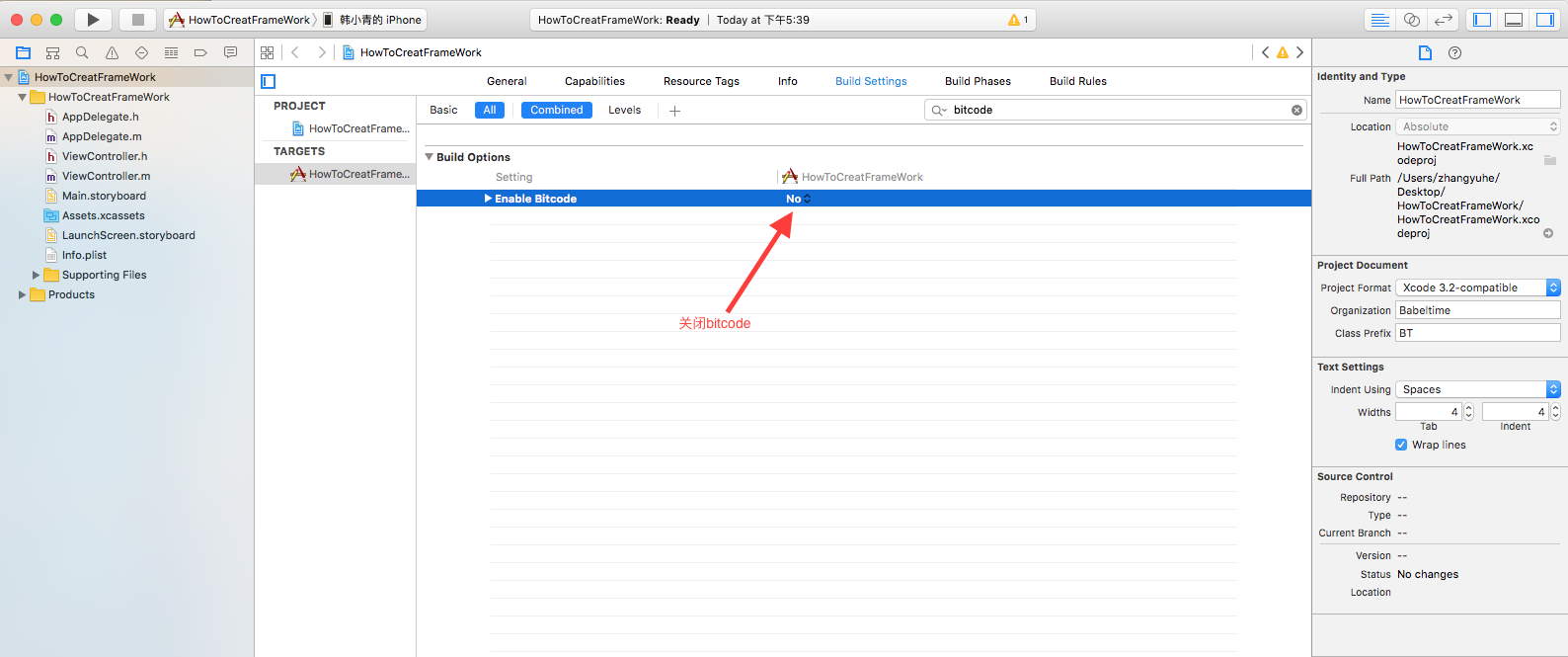
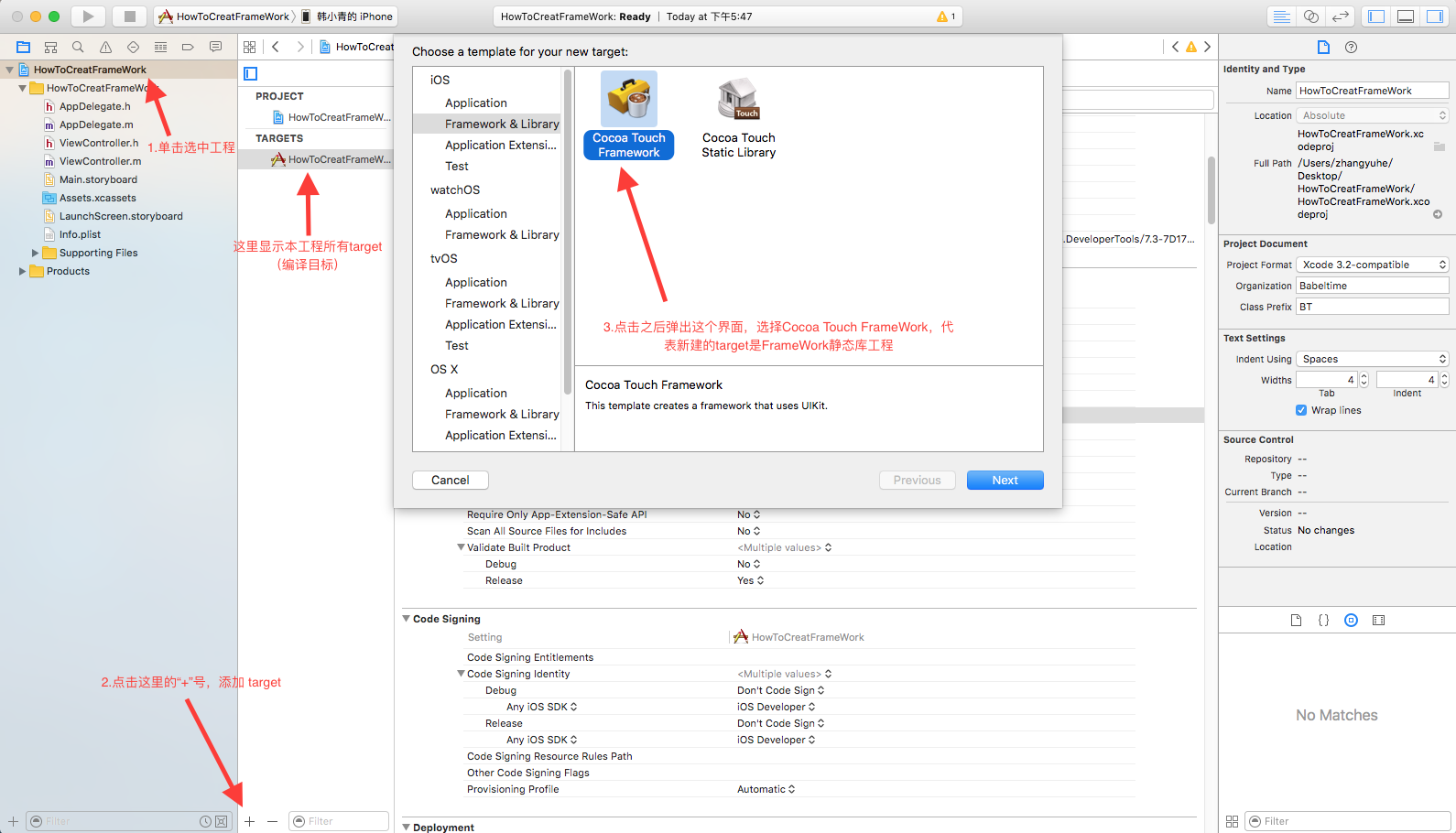
1. 新建一个名字HowToCreatFrameWork的IOS single view工程

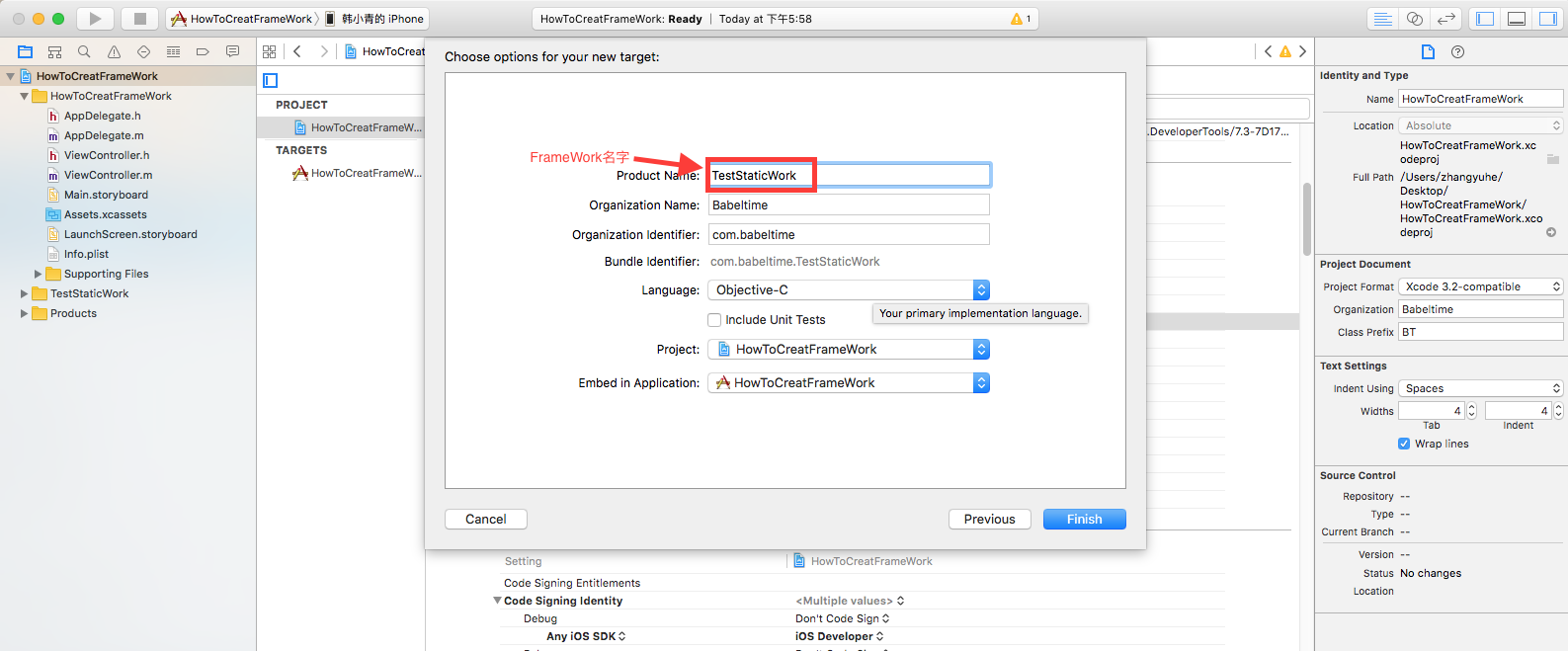


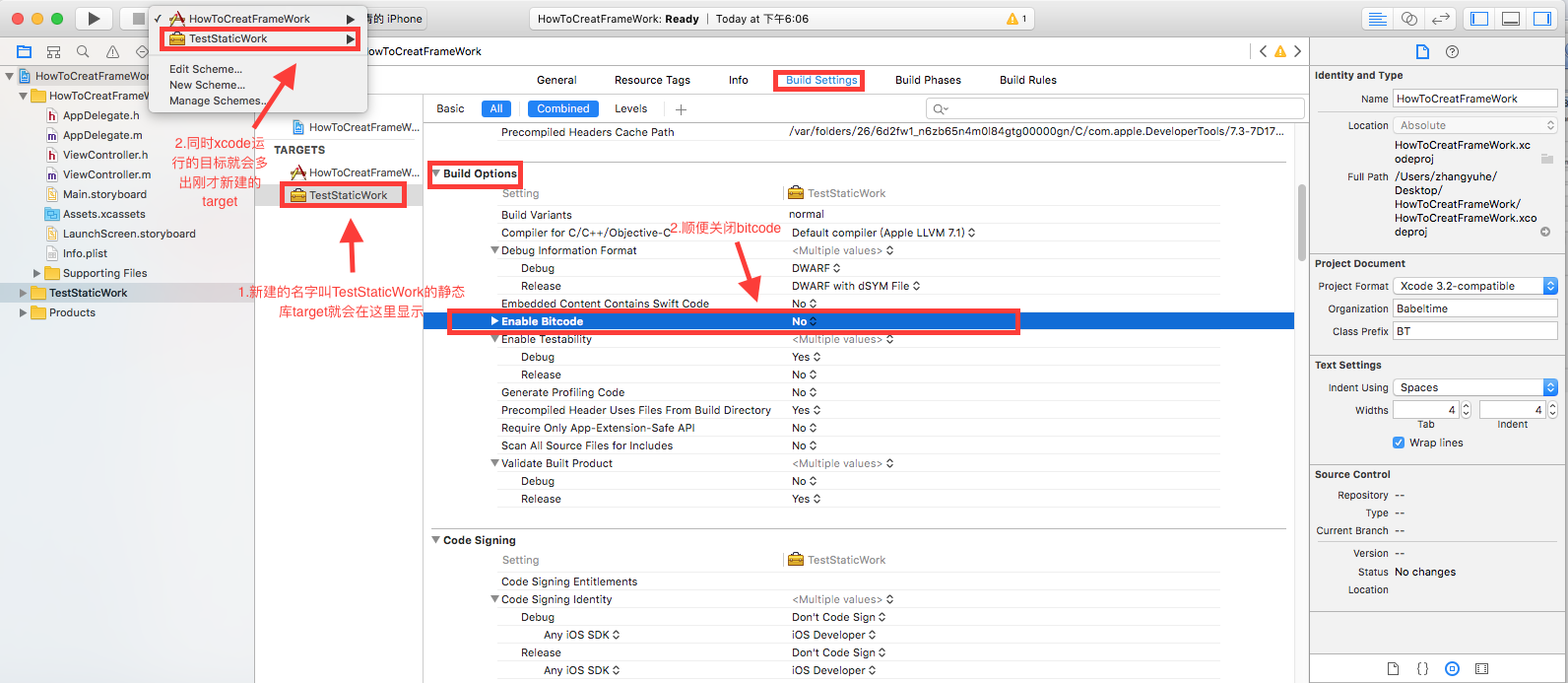
2.build settins 关闭bitcode

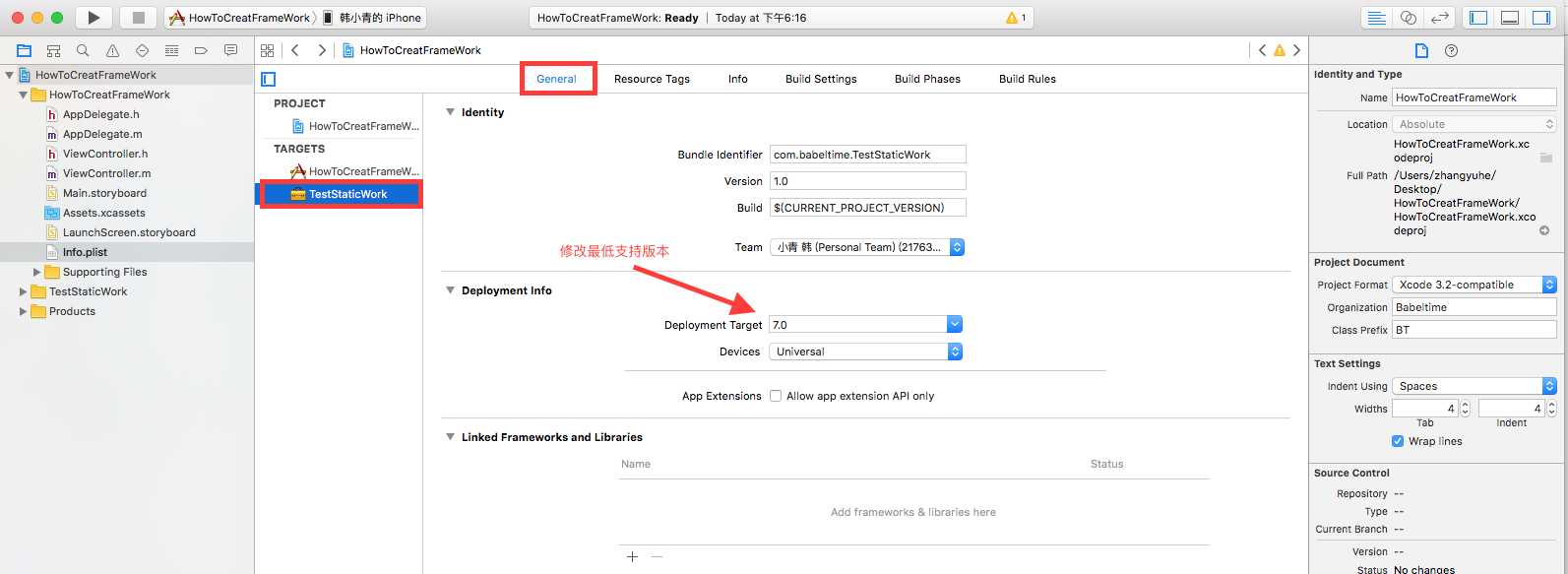


3.开始创建一个farmework 静态库

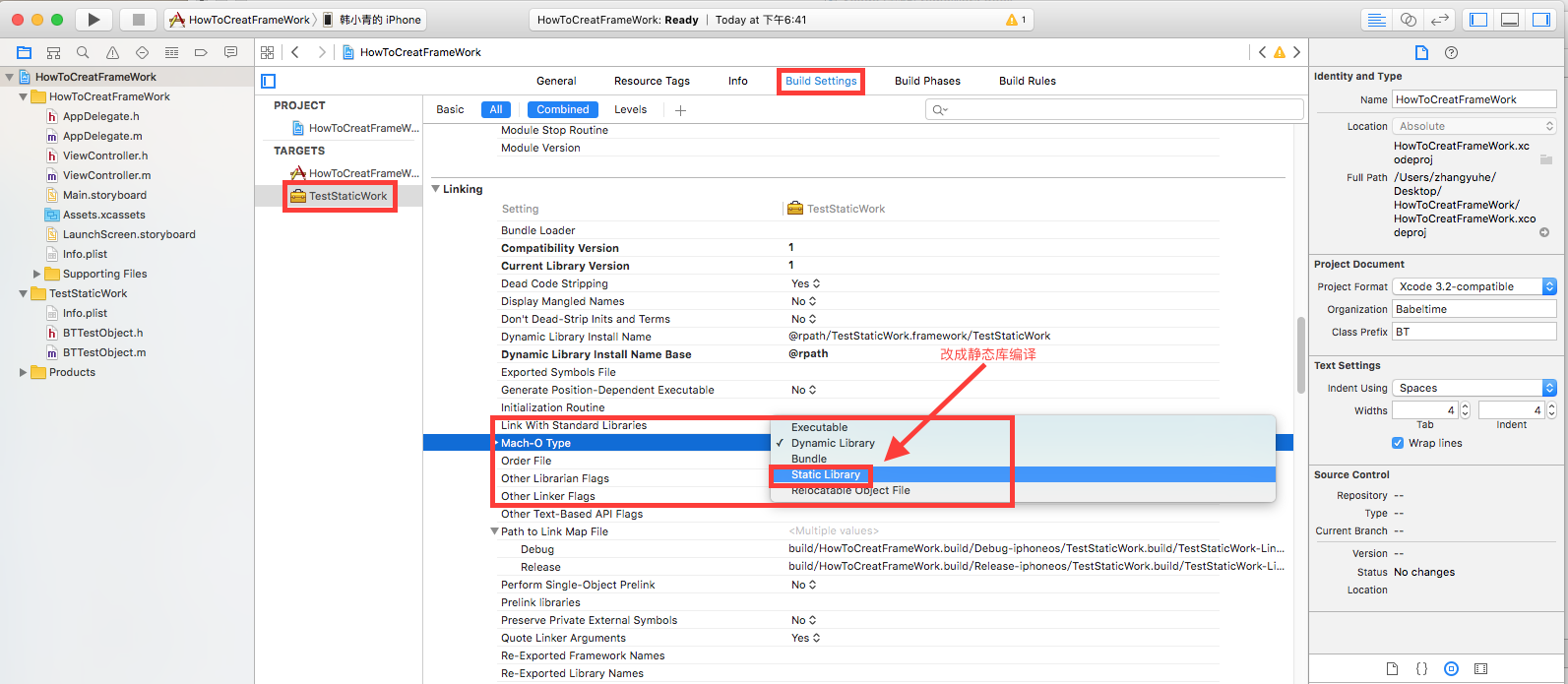




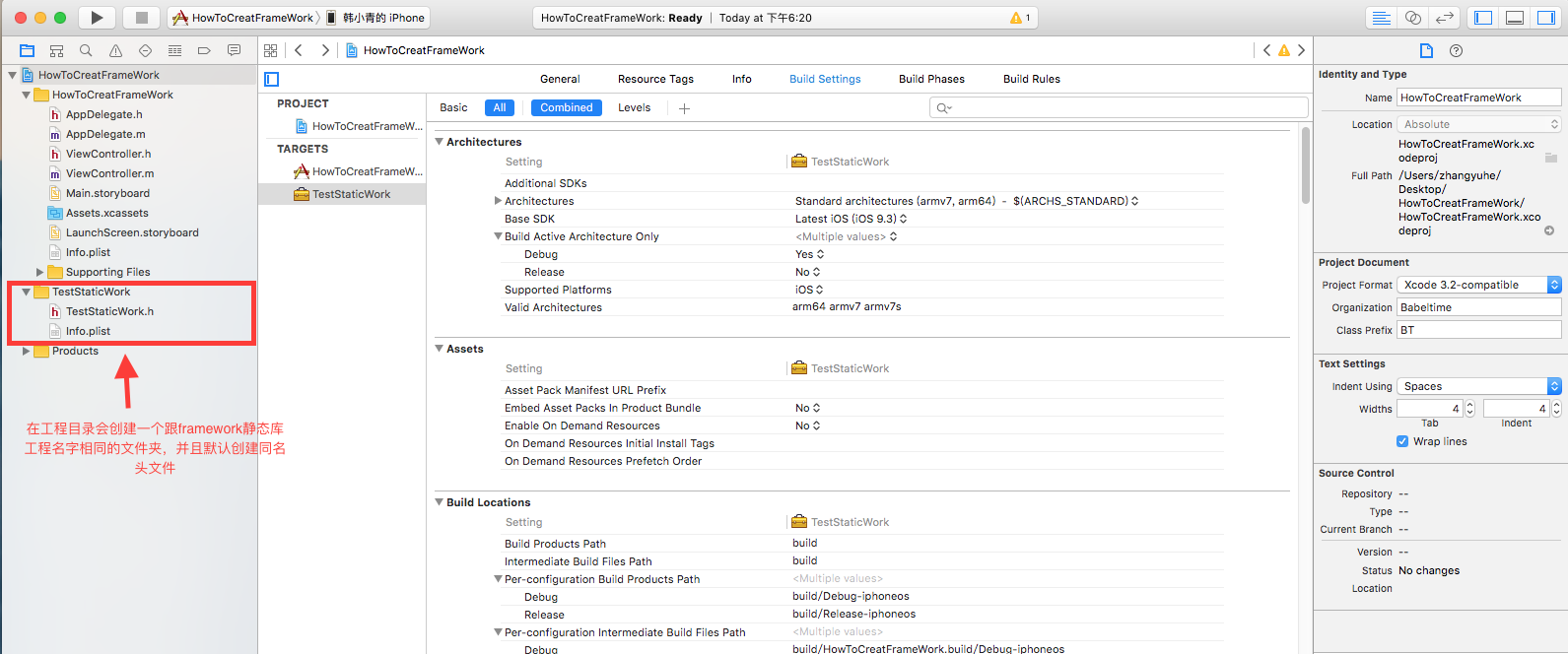




有一个重要的操作！！！系统默认创建出来的framework target是动态的，因此需要手动修改为静态的

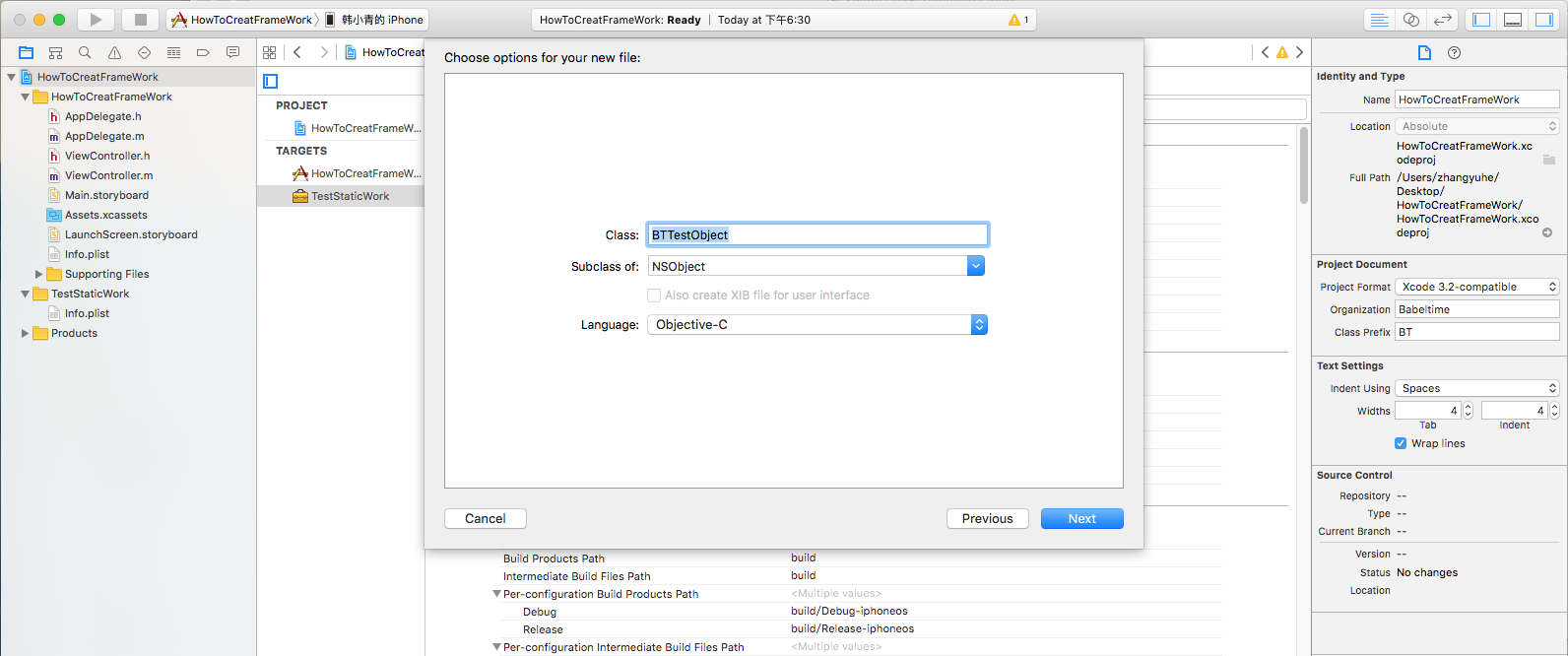


系统会为新的静态库工程target新建相关的目录，如下图



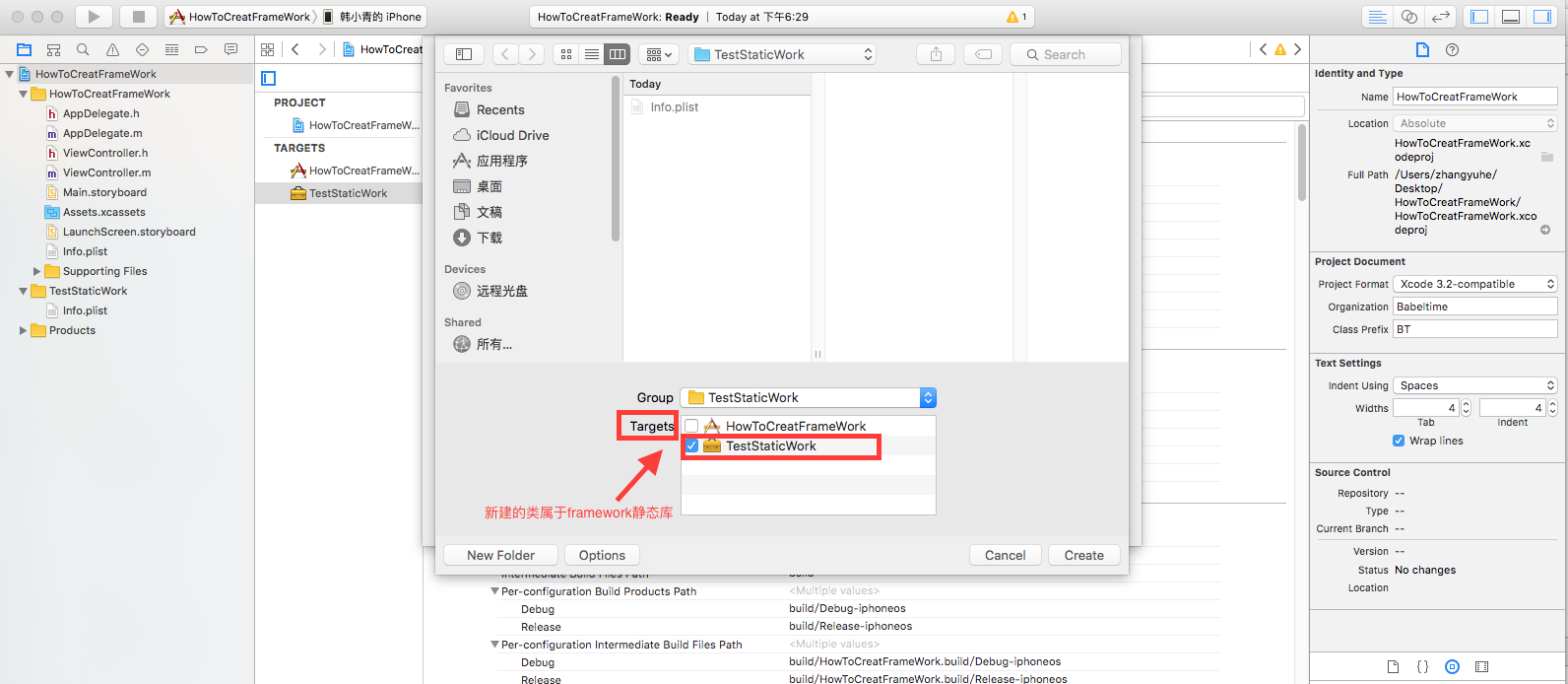
可以删除系统自动创建的头文件，自己创建想要的类，实现对应的功能

本例创建一个名字为BTTestObject的类作为测试

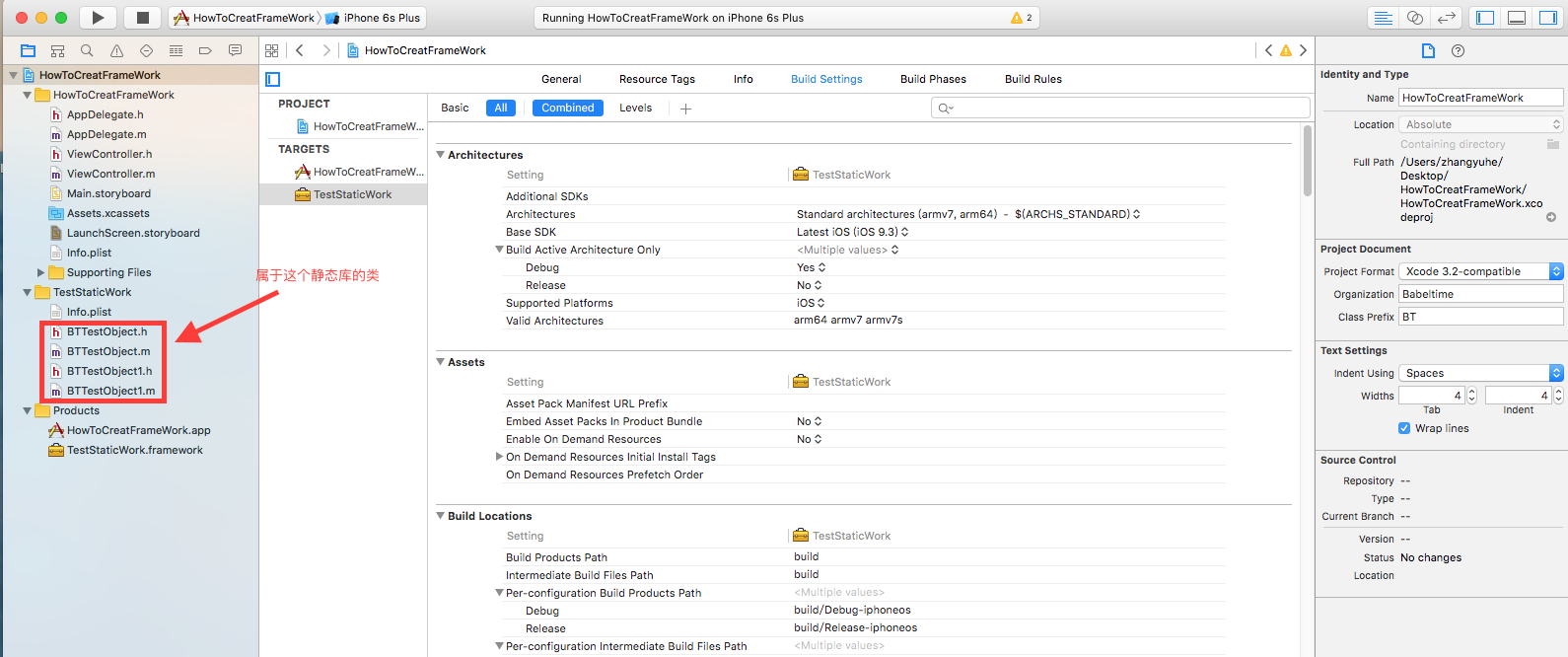


点击next继续 --注意这里新建的类是包含在这个静态库里面的，因此要选择好Tatget

如下图：



同样的方式再创建一个BTTestObject1的类

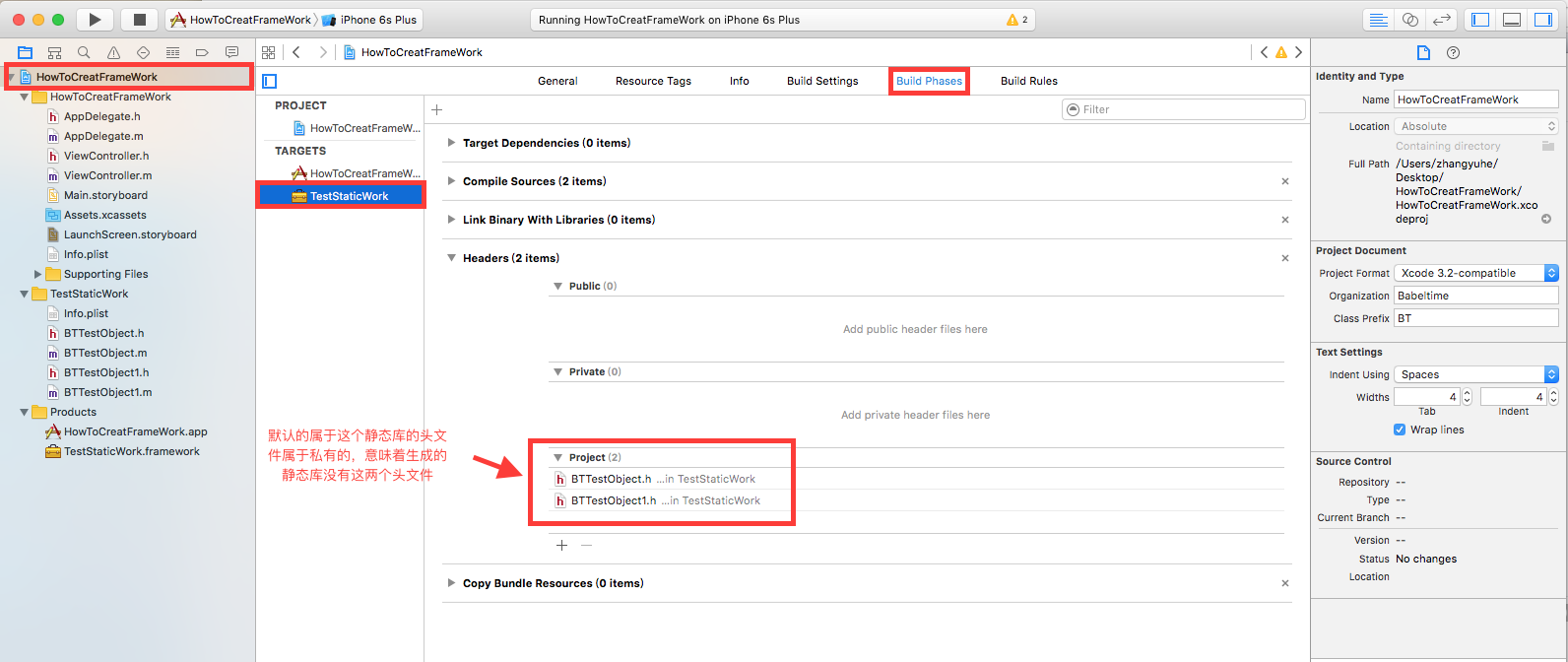


既然静态库工程的目标是生成一个二进制文件，只暴漏一些头文件给被人使用

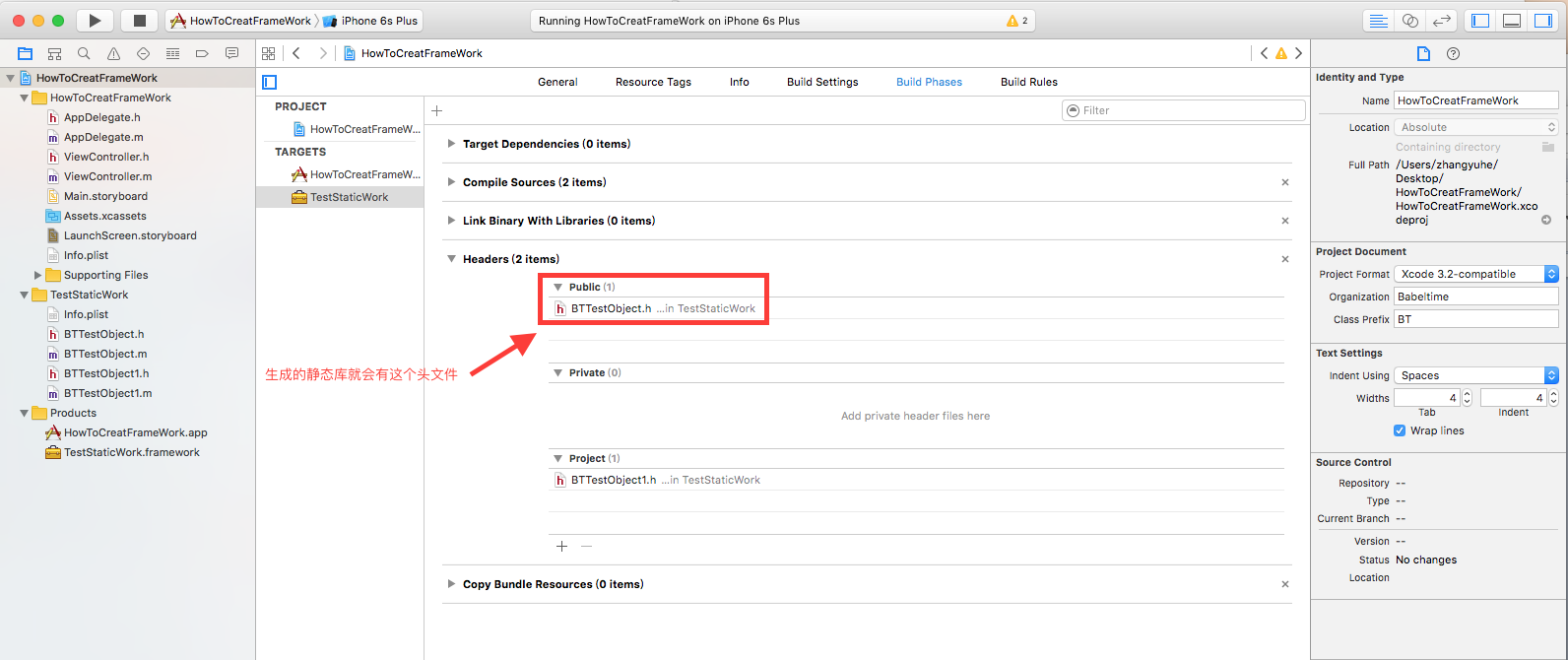
所以需要去配置哪些头文件可以暴露，哪些可以隐藏。

接下来，找到这个静态库对应的 Bulid Phases选项

看到默认情况下所有的头文件都在 project栏目下，私有的。意味着生成的静态库没有这两个头文件

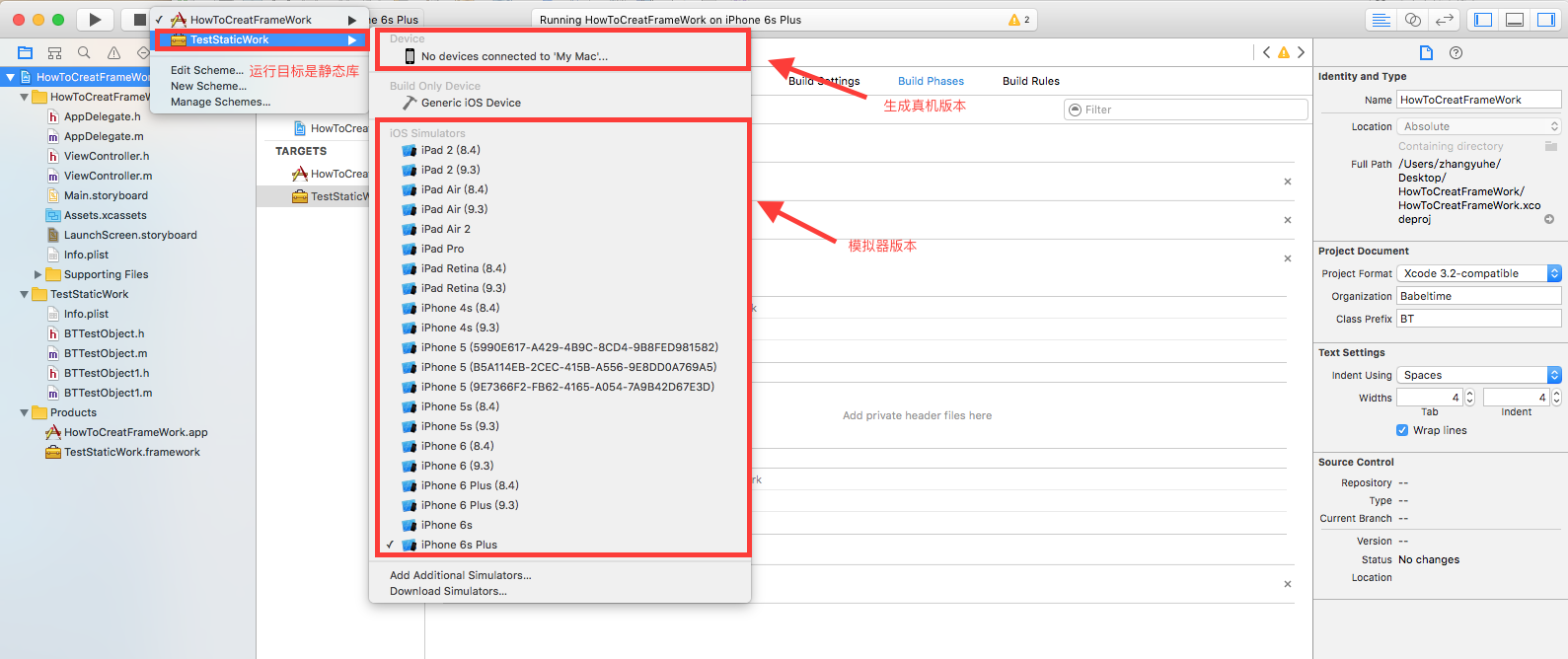


可以选择要暴露的头文件，把它拖到public栏目下：



到此为止，准备工作以及配置都已经完成，接下来就可以生成我们想要的framework静态库了！

把运行目标选择为静态库名字TestStaticWork，command +r运行就可以生成了了。值得注意的是，这里是可以选择运行真机和模拟器，就意味着生成的framework静态库能用到真机还是模拟器



同一工程下引用新建的framewok静态库 不能在buildSetting –>embedded Binnaries里面添加，否则会运行报错。

只要确保Liked Framewoks and Libraries添加自己的framework就行

如果你的静态库里面使用了oc的分类，就要在引入的静态库的工程的build settings中的Other Linker Flags添加-ObjC选项。

原因说明：

Objective－C不会为每一个objc函数生成链接符号，而是会为每一个类生成链接符号。通过category对一个已有的类进行扩展，那么链接器就不知道如何把原代码与category的代码实现关联起来，导致生成的对象无法响应属于category的消息。

解决办法：build settings中的Other Linker Flags添加-ObjC选项。这个选项会促使链接器加载库中所有定义了Class和Category的文件。但这些额外的代码会使目标文件变大。但在64位和iOS应用程序中，linker有一个bug，这个bug会导致objc选项无效，无法从静态库中加载只定义了Category的文件 ，这就需要添加－all\_load或-force\_load选项