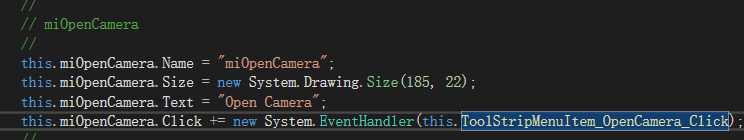
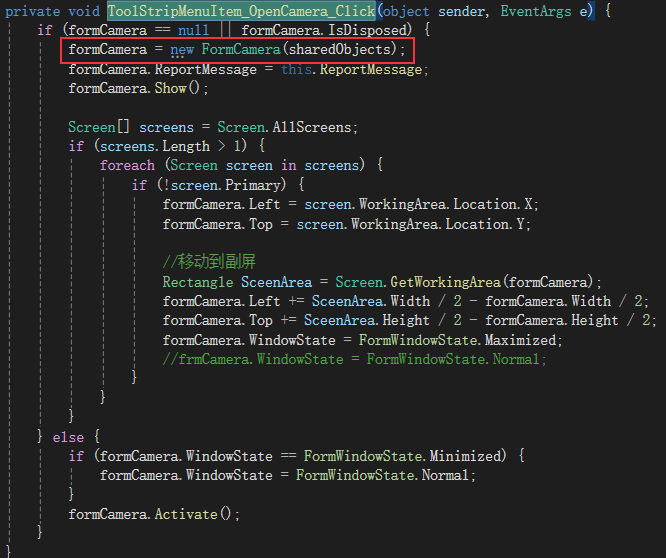
1. 创建

关联到UI上的Open Camera菜单项



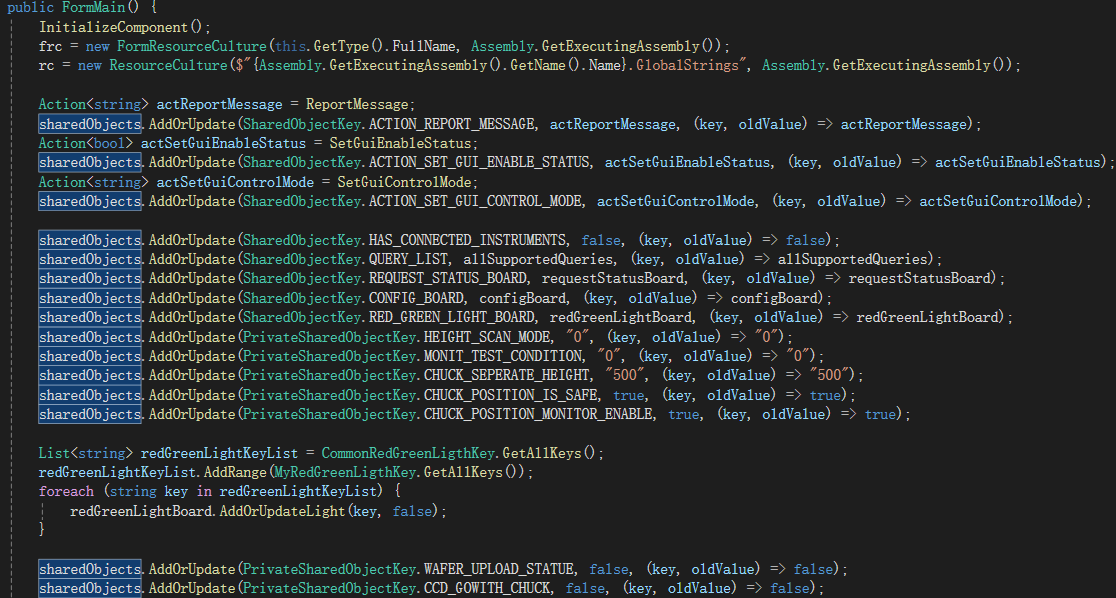


传入的参数是FormMain的私有字段：

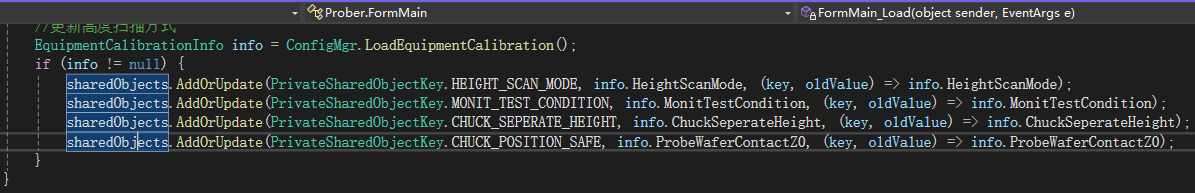
private readonly ConcurrentDictionary<string, object> sharedObjects = new ConcurrentDictionary<string, object>();

*搜索solution会发现这个字段在别的project里面也很常见，看来是一个很常用的数据结构。*

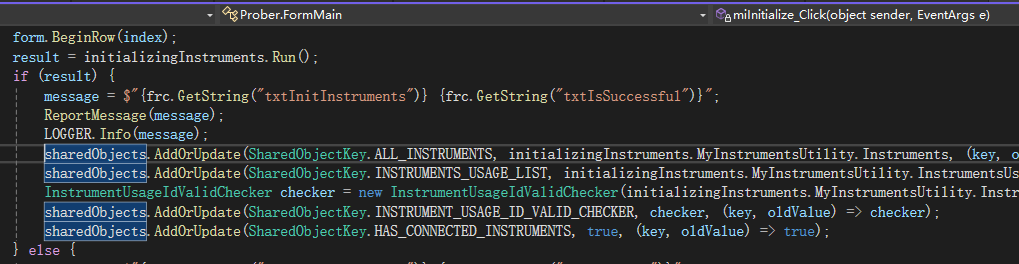
*在FormMain工程中的初始化，主要在FormMain的构造函数中完成：*

**

*在加载函数中也有：*

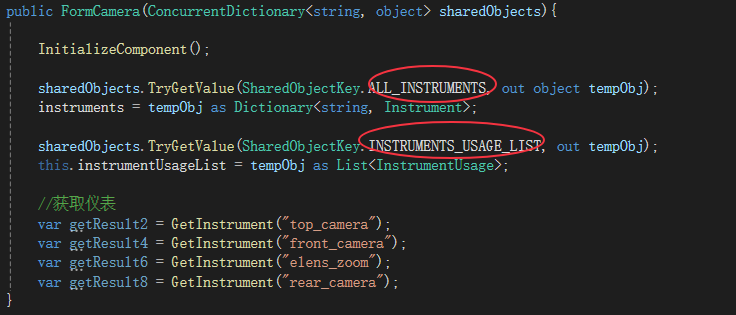
**

*初始化函数中也有：*

**

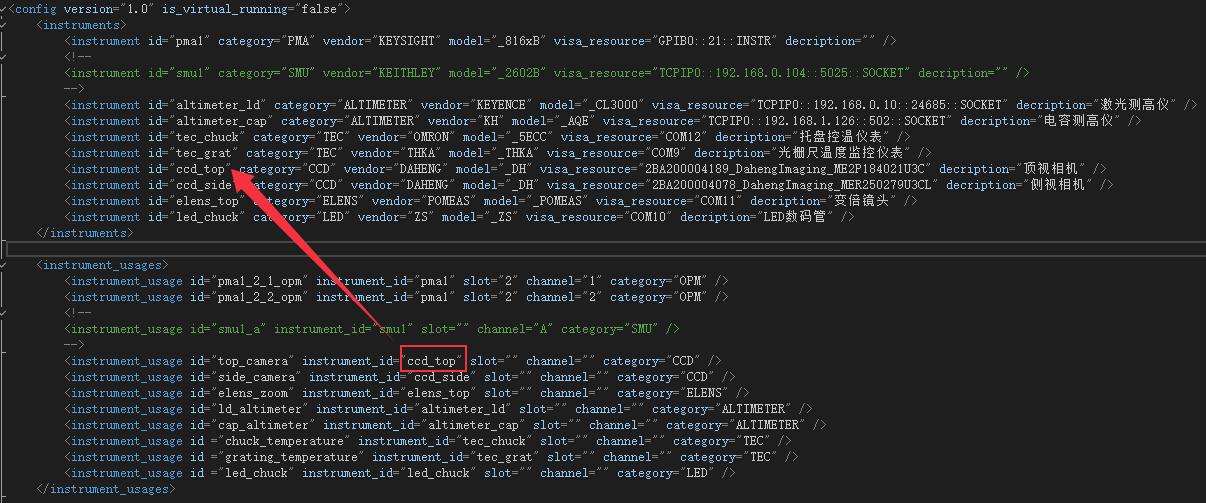
*猜测跟初始化Instruments这里有关。*

再回来看FormCamera的构造函数：

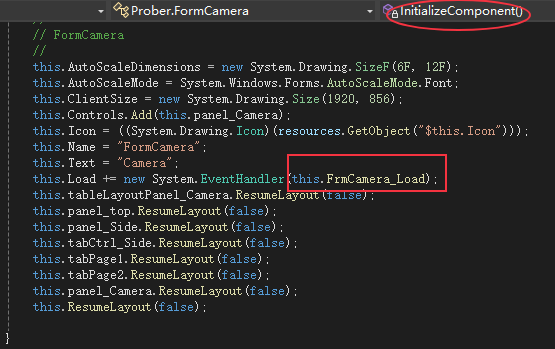


果然是从sharedObjects中提取了ALL\_INSTRUMENTS和INSTRUMENTS\_USAGE\_LIST中的内容。

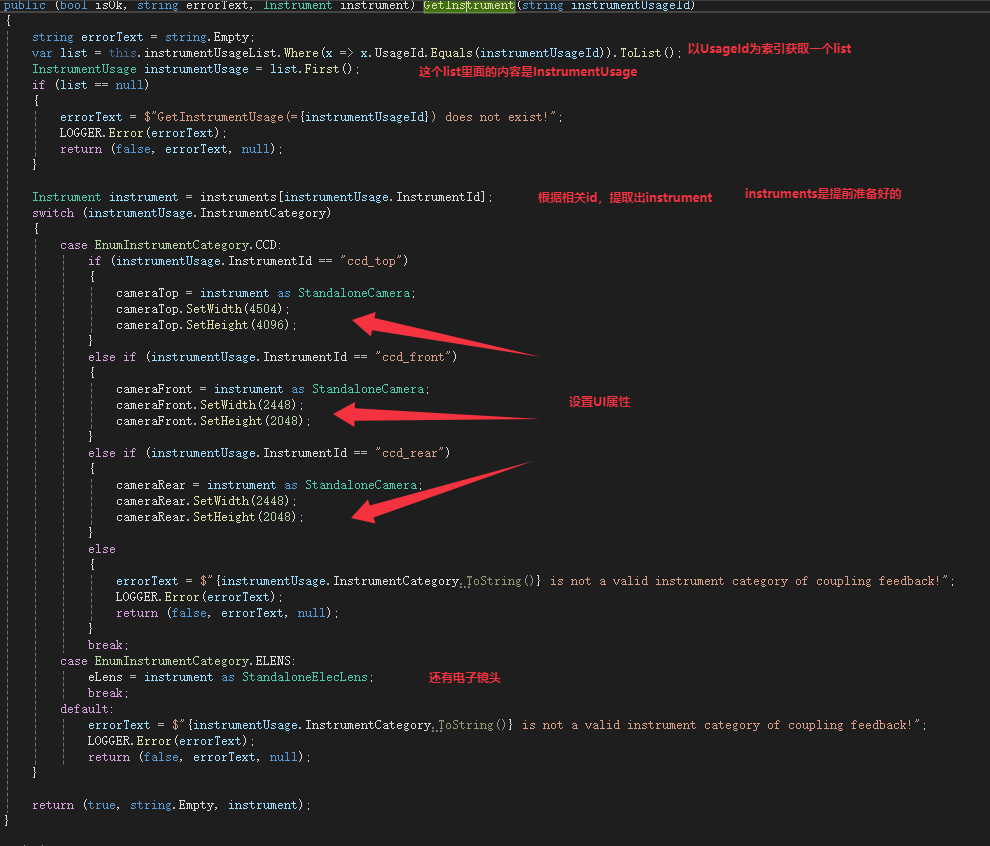
这里的instruments和instruments\_usage，是从config\_instruments.xml文件中来的



另外注意到InitializeComponent中定义了Load：



接下来看看GetInstrument做了什么：

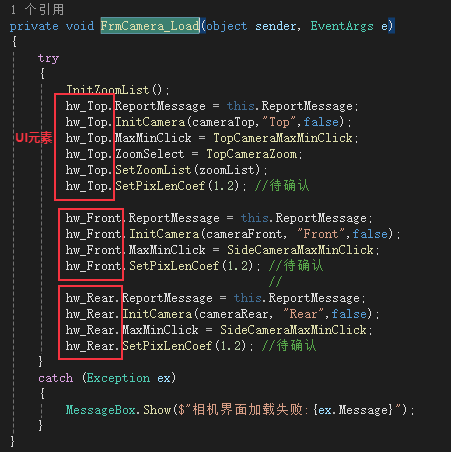


cameraFront\cameraRear\cameraTop变量在这里被赋值了。但是实际的初始化过程，应该是在前面的initialize阶段完成的，这里只是把句柄传递过来。

至此，FormCamera创建完成。

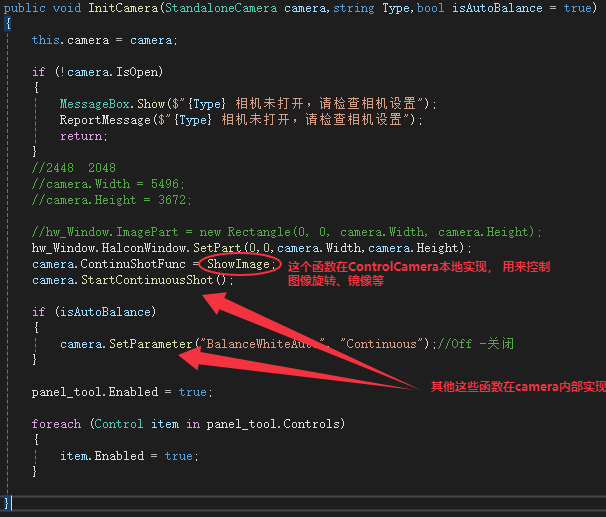
2. Load

将UI元素与instrument关联：



注意这里将ControlCamera的具体操作，以代理的方式，放到FormCamera这边来处理。

再来看看InitCamera函数：



所以camera封装了相机的基础操作，但是把跟UI相关的开放出来了，交由ControlCamera来实现。

3. UI事件处理

