尝试了一轮后发现原来的设计（v1）很复杂，在需求不是特别明确的情况下，为了不是特别重要的功能大动干戈。

现在回过头来尝试一个简化的方案。

缕一缕需求。预估时间到底是什么，有什么用。

在Program创建的时候，我们希望能给出一个大致的预估时间，这样可以让各方大致了解***实验室的工作负载***，以及所关注的***program需要执行的时间***。

这个时间其实很难精确，因为影响因素太多了。先不谈单个测试中，电池的初始状态，充放电条件的设置会对实验时间造成的影响。光是Assets的占用，和周末的停工，就使得预估时间很难准确。即使我获知了每个subprogram和test的执行时间，我仍然无法预估Program的执行时间，因为Assets充足的时候可能多个subprogram并行执行，不足的时候顺序执行甚至间断执行。***所以，统计Program的预估时间才有较大意义。***

*获取sub和tr的预估时间，不仅没有太大意义的，而且根据经验，通过充放电条件的设置来获取sub和tr的预估时间会需要很大的工作量（例如考虑动态电流放电的情况，甚至动态电流充电的情况）。*

在这个需求分析的基础上，我们再来看实现方法：

获取单个Program的执行时间：

在pro, sub, tr上都添加StartTime 和EndTime属性。

tr execute的时候，对tr.StartTime赋值。并检查sub.StartTime是否有值，如果没有，则sub.StartTime = tr.StartTime。再检查pro.StartTime是否有值，如果没有，则pro.StartTime = tr.StartTime。

tr commit的时候，对tr.EndTime赋值。如果sub完成，则sub.EndTime = tr.EndTime. 如果pro完成，则pro.EndTime = tr.EndTime

sub完成的条件：

如果testcount为one，则FirstTestRecords[length].Status ==Completed.

如果testCount为two，则FirstTestRecords[length].Status ==Completed && SecondTestRecords[length].Status ==Completed

pro完成的条件：

每个sub都完成

单次运行时间 = 结束时间 - 开始时间

开始时间 = 第一个Test Execute的时间

结束时间 = 最后一个Test Commit的时间

通过单个program的执行时间获取预估时间，主要有两种方式：

一．平均时间

预估时间 = 每次运行时间之和 / 执行次数

问题关键在于如何将pro归类。因为每一个pro都是一个独立的实体，相互之间没有关联。

为了将pro关联起来，在pro上添加ProgramGroup属性。执行次数即相同Group的pro个数。运行时间之和即相同Group中的pro的运行时间之和。

Save As的时候，如果不编辑BatteryType和Subprogram，则Group不变。否则归入新的Group。

Create的时候，归入新的Group。

二．上次时间

预估时间 = 上次执行时间

Create Program时，没有预估时间。Save As时，预估时间即为Copy源的执行时间。（即使Save As的时候对sub进行了编辑，也一样，反正是个大概）。如果换了BatteryType，则没有预估时间。

单次运行时间和预估时间，都是可以通过model层的数据计算出来的，所以不需要放到数据库中。而只是ViewModel里的东西。