

身为产品经理，若要保证所有必要的状态转换和事件都完整、准确的在产品功能需求中描述出来，状态转换图是必不可少的强力工具。



绝大多数智能硬件产品是一个包含状态转换、数据操作和功能执行的综合系统（这个系统的状态是有限的），它在任何时刻都处于众多状态中的某一种状态。只有当某个特定的事件发生或某个被定义的标准被满足时，系统的状态才会发生转换，由一种状态向另一种状态发生转变。比如，当处于“关闭”状态的电灯的开关系统接收到用户的指令时（用户按开关即是“事件”），电灯从“关闭”状态变为“开启”状态。

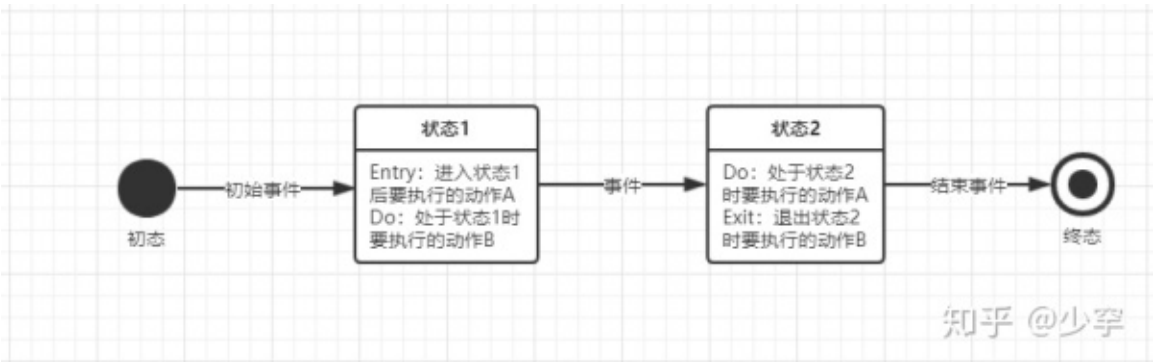
一、为什么需要绘制状态转换图？

当我们用文字来描述一系列复杂的状态转换逻辑时，很可能会忽略某些关键的状态变化过程，也可能导致某些状态变化的重复和新增一些本不该出现的状态变化。这就使需求不清晰，从而导致文档的读者（通常是工程师）无法全面的理解系统的行为变化。

这时候，我们就需要引入 UML 中的状态转换图来展示系统状态关系的全貌，状态转换图英文全称 State transition diagram，缩写 STD，简称状态图。它属于事件驱动模型，表示系统对外部事件的响应方式，能清晰的描述系统状态之间的转换顺序和状态之间的关系，在节省大量文字描述的情况下帮助工程师更好的理解需求和讨论设计思路，避免开发时出现状态转换逻辑错误，并且系统实现后还要用状态模型来论证系统的结构和操作。状态

图还明确的定义了状态发生转换时必要的触发事件和影响状态转换的关键因素，有利于在开发过程中避免非法事件的进入。通过绘制状态图，还可以帮助我们检测系统设计中是否存在缺陷。状态图既可以表示系统循环运行过程，也可以表示系统单程生命期。所以，绘制状态转换图这项技能对产品经理是至关重要的。

二、状态转换图有哪些组成要素？



状态图样例

状态图（STD）表示系统状态和引起状态改变的事件，它包含三种关键元素：

01.可能的系统状态

系统在任何时候都会处于某种状态中，所有系统都有状态，一个状态代表系统的一种行为模式。比如手机具备“待机”、“通话中”、“关机”等状态，处于“关机”状态的手机无法接听电话（消息），而处于“待机”状态的手机则可以接听电话。这说明系统所处的状态决定了系统对事件的响应方式或所接受的消息。

在状态图中定义的状态主要有初态(即初始状态)、终态(即最终状态)和中间状态。在一张状态图中只能有一个初态，而终态则可以有 0 至多个。

系统状态一般用圆角矩形表示，如上图所示。其中可能还包括处于该状态时将要执行的非原子动作（可中断）的简单描述，以 Do 引出，动作完成后状态就结束，然后一个从当前状态出发的转换被触发。

状态还可能包括进入动作和退出动作，进入动作指进入状态时执行的原子动作（不可中断），以 Entry 引出，比如手机进入“充电”状态后屏幕就显示充电图标。退出动作指退出

状态时执行的原子动作，以 Exit 引出，比如手机退出“充电”状态后屏幕就不再显示充电图标，进入“待机”状态。

02.允许的状态转换

即状态之间的转换关系，比如电脑可以从“待机”状态转换“睡眠”状态。

状态的转换一般用连接两个圆角矩形的箭头表示，如上图所示。

03.导致状态发生转换的事件

事件使得系统从一个状态转换为另一个状态，比如用手指点击一个处于“开启”状态的智能音箱的电源，使智能音箱变为“关闭”状态，在智能音箱状态的变化是被“用手指点击开关”这个动作引起的，所以“用手指点击开关”就是一个事件。

事件主要分为四类：信号事件、调用事件、变化事件、时间事件。事件一般用状态转换箭头上的文字标签来表示，如上图所示。

在绘制状态图（系统建模）时，图形符号的使用通常是非常灵活的，不必严格遵守符号的形式和细节，比如描述系统状态，不管是圆角矩形还是圆形都是可以的，但需要注意的是，对同一事物的描述要使用相同的符号，以保证表述的一致性。

三、绘制状态图的步骤有哪些？

为了便于理解，我们以一个简化了的智能洗衣机控制系统来分析一下绘制状态图的方法。这款智能洗衣机具备：一个用来显示按钮和设备设置的触控屏；一个用来选择洗涤模式的按钮，可以选择强力洗涤和超快洗涤两种方式；一个用来设置水量的数字键盘；一个能控制开始/停止的按钮。还具备安全锁功能，在没关闭洗衣机仓门时洗衣机不会工作，工作中打开盖子洗衣机会暂停工作，且工作完成后洗衣机会发出提示音提示用户来取衣物。

假设该智能洗衣机的操作步骤如下：

选择洗涤模式，强力洗涤或超快洗涤；用数字键盘设置本次洗涤所需水量；点击开始按钮，使用相应洗涤模式和水量开始洗涤。

产品逻辑梳理清楚之后，我们下面开始绘制状态转换图。

步骤 1：列出产品/系统的所有状态

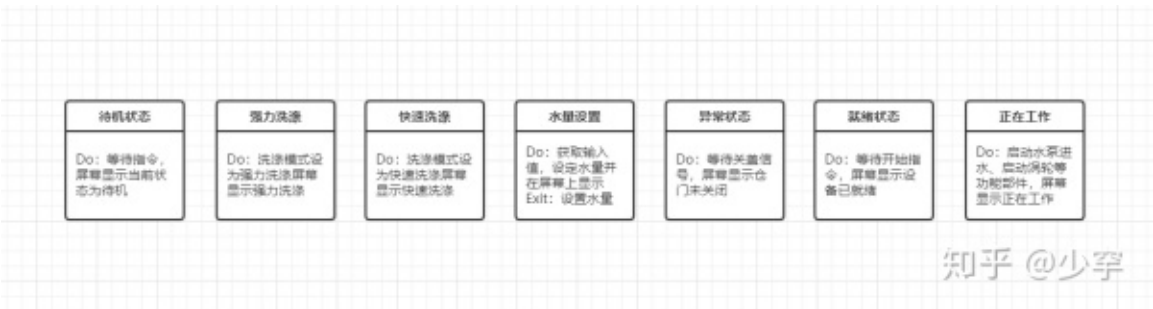
梳理产品逻辑，列出产品/系统可能出现的所有状态，比如：待机状态、强力洗涤、快速洗涤、水量设置、可工作状态（就绪）、不可工作状态（异常）、正在工作等。



罗列状态

步骤 2：列出每个状态须执行的动作

在状态名称下方列出该状态下所包含的所有动作，即用 Entry，Do，Exit 标注进入动作、执行动作和退出动作。比如：待机状态下执行等待指令动作、强力洗涤状态下设置洗涤模式为强力洗涤等。



罗列动作

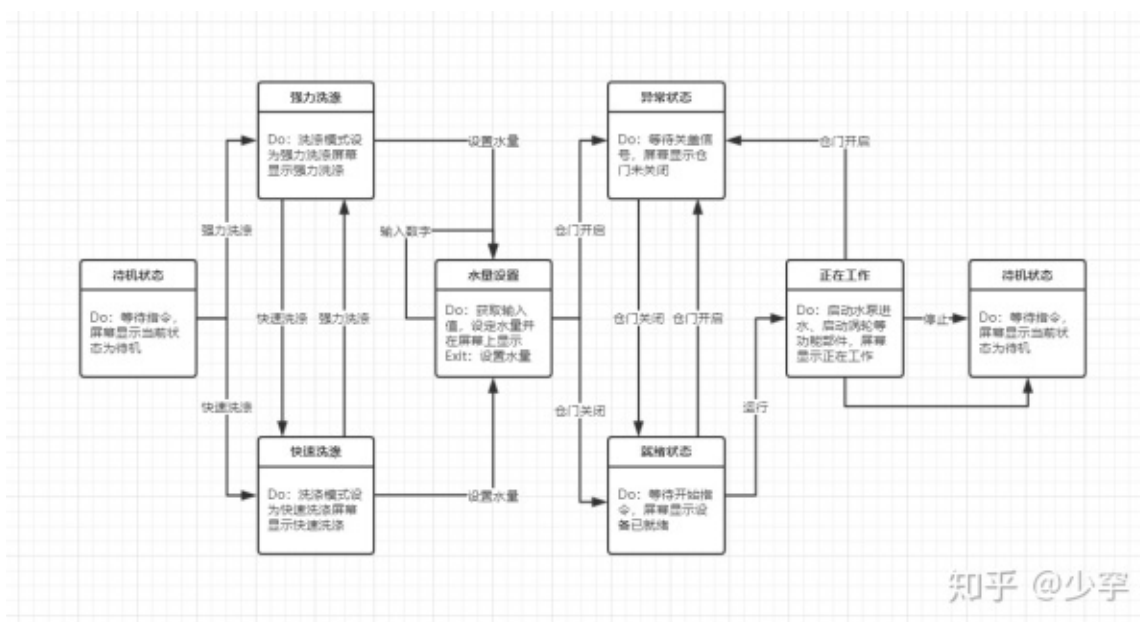
步骤 3：确认并绘制出引起状态发生转换的事件

事件可以通过状态表来梳理，状态表是用矩阵的形式表示不同状态之间存在的所有转换，通过使用状态表和分析矩阵中的每一个单元格，能确保遍历所有的状态转换事件。首先，在表格的首行和首列分别列出系统的所有状态。单元格表示列状态到行状态之间的转换是否有效，如果是有效的转换状态，在单元格中写出引起转换的事件，如果是无效的转换，则可以使用 “” 或 “无” 表示。这样的状态表能保证我们对所有的状态转换没有遗漏，它可以帮助阅读者直观的理解可能存在的转换顺序。

	待机状态	强力洗涤	快速洗涤	水量设置	就绪状态	异常状态	正在工作
待机状态	\	强力洗涤	快速洗涤	\	\	\	\
强力洗涤	\	\	快速洗涤	设置水量	\	\	\
快速洗涤	\	强力洗涤	\	设置水量	\	\	\
水量设置	\	\	\	输入数字	仓门关闭	仓门开启	\
就绪状态	\	\	\	\	\	\	运行
异常状态	\	\	\	\	仓门关闭	\	\
正在工作	停止/完成	\	\	\	\	仓门开启	\

状态转换表

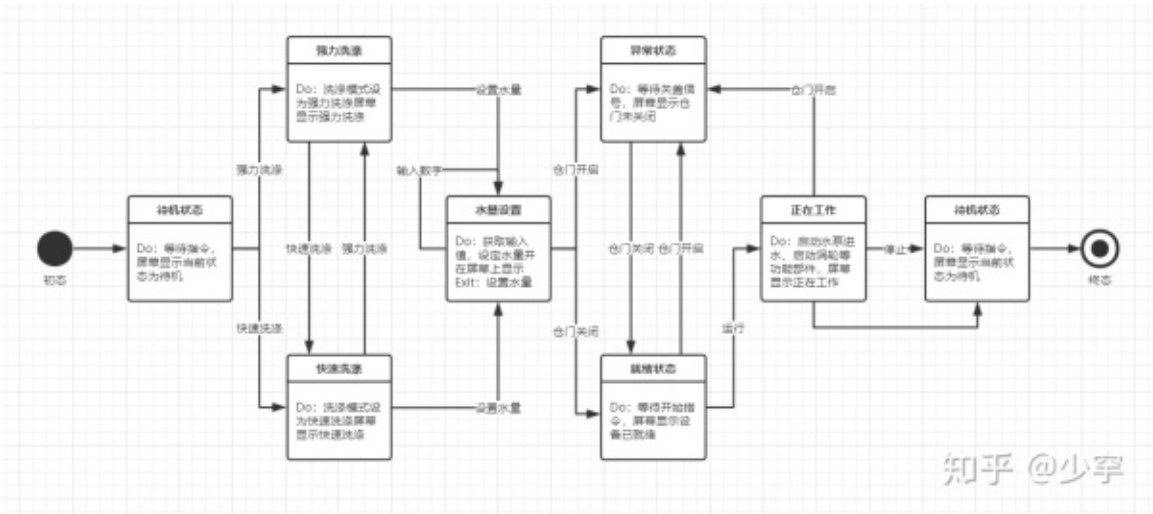
使用状态表梳理出所有引起状态发生转换的事件后，就可以通过箭头连接可以发生转换的状态，并在箭头上标注出引起该状态转换的事件或条件。比如：系统处于待机状态时，用户可以点击强力洗涤按钮进入强力洗涤状态或点击快速洗涤按钮进入快速洗涤状态，当用户改变想法时可以点击另一个洗涤模式按钮。设置好水量并且关闭仓门后，点击开始按钮，智能洗衣机就会开始一个工作周期，工作周期完成后系统回到待机状态。



绘制并标注事件

步骤 4：标注初态和终态并细化状态图

在步骤 3 的基础上标注出初态和终态，然后补充更多系统状态和系统事件方面的细节，可以用表格的形式来展现，下面的两个表格也可以分别在进行步骤 2 和步骤 3 时创建，这样一份完整的状态转换图就完成了！



完整的状态转换图

状态	描述
待机状态	智能洗衣机等待指令输入，屏幕显示“待机”
强力洗涤	智能洗衣机的洗涤模式设置为强力洗涤，屏幕显示当前设置为“强力洗涤”
快速洗涤	智能洗衣机的洗涤模式设置为快速洗涤，屏幕显示当前设置为“快速洗涤”
水量设置	洗涤任务的水量由用户输入，屏幕显示所设置的水量，当更改水量时更新显示
就绪状态	智能洗衣机当前可用，屏幕显示“已就绪”
异常状态	智能洗衣机当前不可用，屏幕显示“仓门未关闭”，提示关闭仓门
正在工作	智能洗衣机工作，涡轮旋转。屏幕显示“正在工作”，当洗涤完成，声音模块发出3秒提示音提示用户工作完成，屏幕显示“洗涤完成”

知乎 @少亨

状态描述表

事件	描述
强力洗涤	用户点击强力洗涤按钮
快速洗涤	用户点击快速洗涤按钮
设置水量	用户按下某个水量选项
输入数字	用户按下数字键输入水量
仓门开启	智能洗衣机的仓门开关开启
仓门关闭	智能洗衣机的仓门开关关闭
运行	用户点击开始按钮
停止	用户点击停止按钮

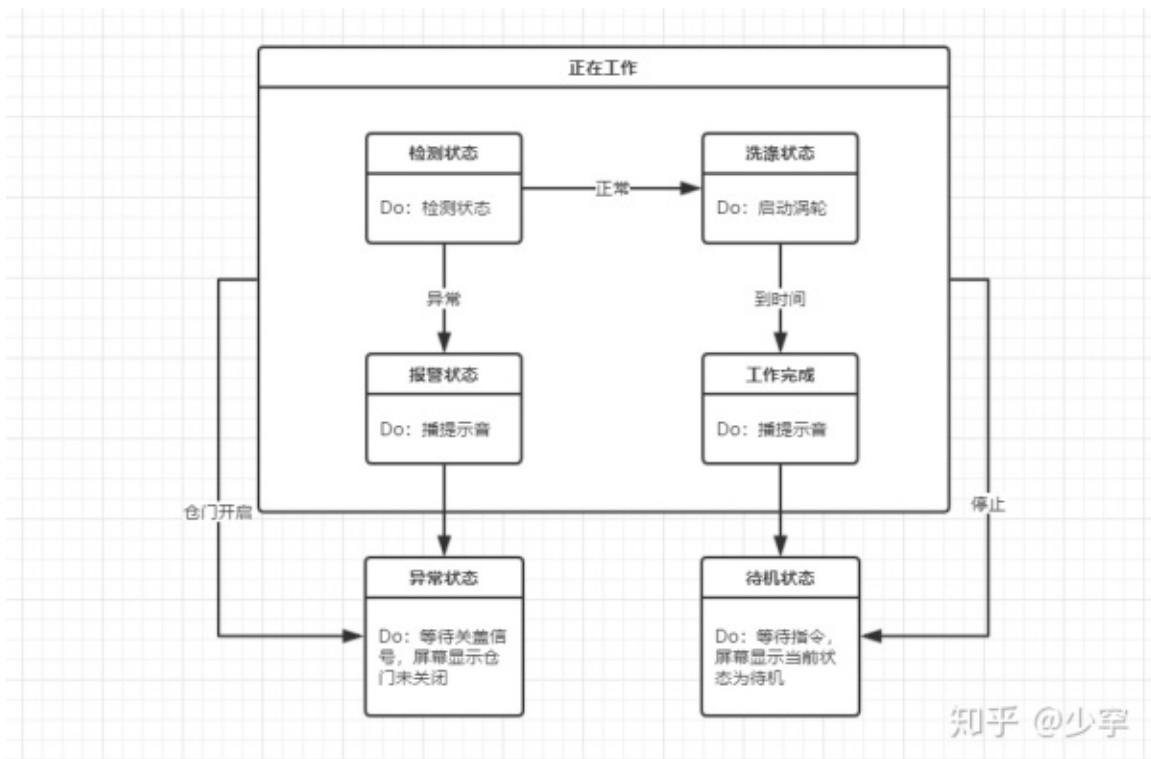
知乎 @少亨

事件描述表

四、超态与子状态

由于智能硬件的系统越来越复杂，系统的状态数量急速增加。因此，对于比较复杂的系统模型，我们需要隐藏一些系统执行处理的细节，仅展示系统状态的变化。此时就需要引入超态（superstate）和子状态（substate），它们是一种嵌套关系，超态中嵌套了两个或两个以上的子状态。这两个概念的引入，可以帮助我们更好的绘制和解释复杂系统的状态转换关系。

依然拿智能洗衣机为例，它在工作开始前会检测系统状态，如果一切正常则启动涡轮开始洗涤，洗涤完成后声音模块播放提示音，然后设备进入待机状态；如果检测到任何异常，则声音模块播放报警提示音，然后设备进入异常状态。通过下图可以看到，正在工作是超态，而检测状态、洗涤状态、报警状态、工作完成都是其子状态。



超态与子状态

五、绘制状态转换图的工具

工欲善其事必先利其器，可以绘制状态转换图的工具有很多，除了原型绘制工具外，还可以尝试一些专业绘图工具。

下面推荐一些简洁易用的绘图工具：

Diagram Maker | Online Diagram Software Online Diagram and Flowchart Software | Cacao Diagramming Software & Team Collaboration Tools | Gliffy
ProcessOn – 免费在线作图，实时协作 流程图制作软件，图表制作软件 – Microsoft Visio

六、结语

状态转换图无须展示系统数据处理的细节，它只需完整的展示系统运行可能导致的状态的变化。状态转换图提供了一种横跨多个用户故事或用例的更高层视图，且每个用户故事或用例都可能涉及系统状态的转换。它帮助项目干系人快速理解系统状态联系，更快理解系

统的预期行为。状态转换图涵盖了所有状态可能发生变化的关键路径，测试人员可以由状态图衍生出测试用例，使早期的测试工作更顺利的展开。

作者：少宇，知乎：少宇

题图来自 Unsplash，基于 CC0 协议