**自动咖啡机的嵌入式**

**控制软件开发总结报告**

完成人员：万学远 潘捷 谢鑫磊 张世舜 王振鸿

二零一七年十一月四日

# 一. 制定工作计划

拿到任务后，与小组成员讨论协商，在老师与助教的帮助下，制定了如下工作计划：

## 已选择任务及人员分配

|  |  |
| --- | --- |
| **确定任务** | **参与人员** |
| **《软件需求规范文档》；《**概要设计说明书》 | 共同参与 |
| Cup\_Storage模块 | 王振鸿 |
| Feeder模块；软件说明文档 | 张世舜 |
| Conveyor\_Belt模块；Mix模块；物理模块；  整体调试；总结报告 | 万学远 |
| Hot\_Water模块；PIL测试报告 | 潘捷 |
| Doser模块；SIL测试报告 | 谢鑫磊 |

## 

* + 1. **控制软件开发工作计划**

|  |  |  |
| --- | --- | --- |
| **时间区间** | **具体工作内容** | **预期成果** |
| 10月15至  10月21日 | 软件团队与咖啡机产品经理（老师和助教）沟通软件开发需求，形成软件设计规格书，清晰的定义控制软件的各项功能、硬件输入和输出接口。在Matlab/Simulink环境中，实现咖啡机的物理建模，同时完成控制软件的概要设计，划分成若干个软件模块，确保模块覆盖设计规格书的所有功能，清晰定义各个模块的接口。 | 交付《软件设计规格书》  咖啡机物理模型文件  《概要设计说明书》 |
| 10月22至  10月28日 | 基于simulink/stateflow建模方法，设计各个软件模块的控制逻辑和算法，并与咖啡机物理模型进行协同仿真，实现整个咖啡制作流程。基于模块的stateflow模型，手动编写或用自动代码生成工具，完成C代码设计工作。 | 交付 控制软件模块stateflow模型文件  模块C代码及代码自动生成报告 |
| 10月29至  11月5日 | 基于MATLAB工具，对模块C代码进行Software in Loop测试或Processor in Loop测试，得到测试报告。 | 交付《SIL/PIL测试报告》  项目总结和软件说明文档 |

# 二. 第一周总结

第一周大家通过各种渠道，自学了stateflow的相关建模方法，讨论写出了软件需求文档：

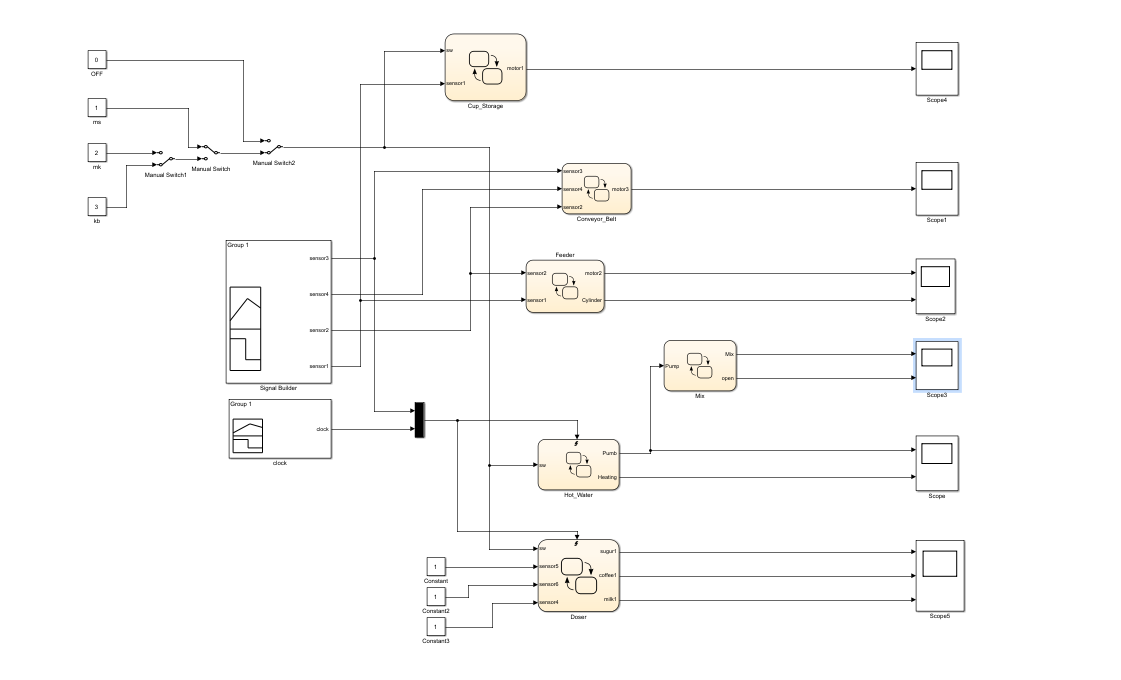


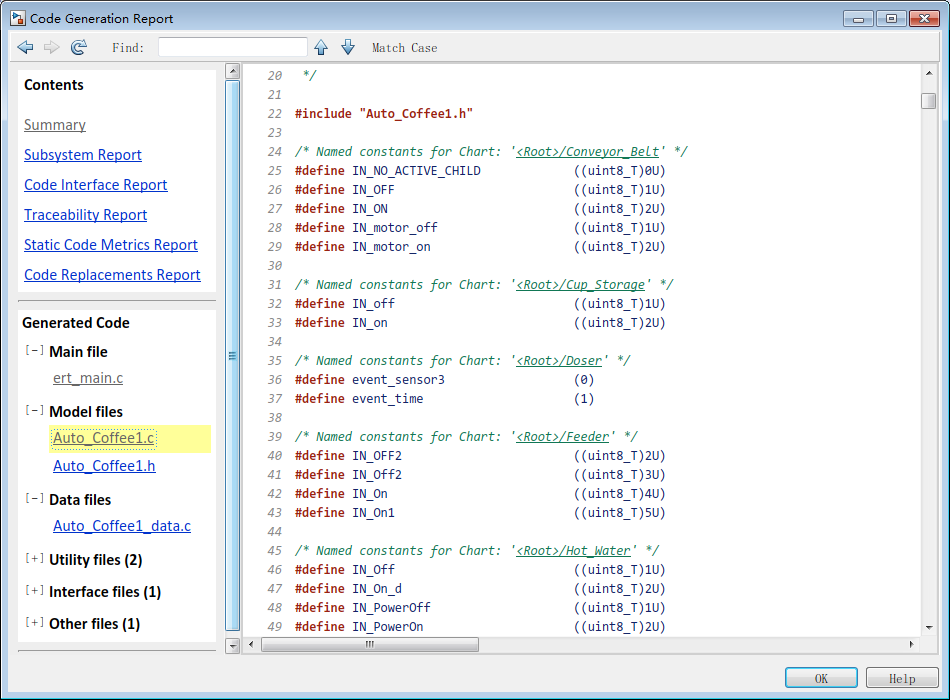
在GitHub上建起了团队，将大家都拉进，以便文件的提交与修改。各自完成了自己所负责的模块的stateflow建模，然后根据自己的模块介绍，完善了《概要设计说明书》



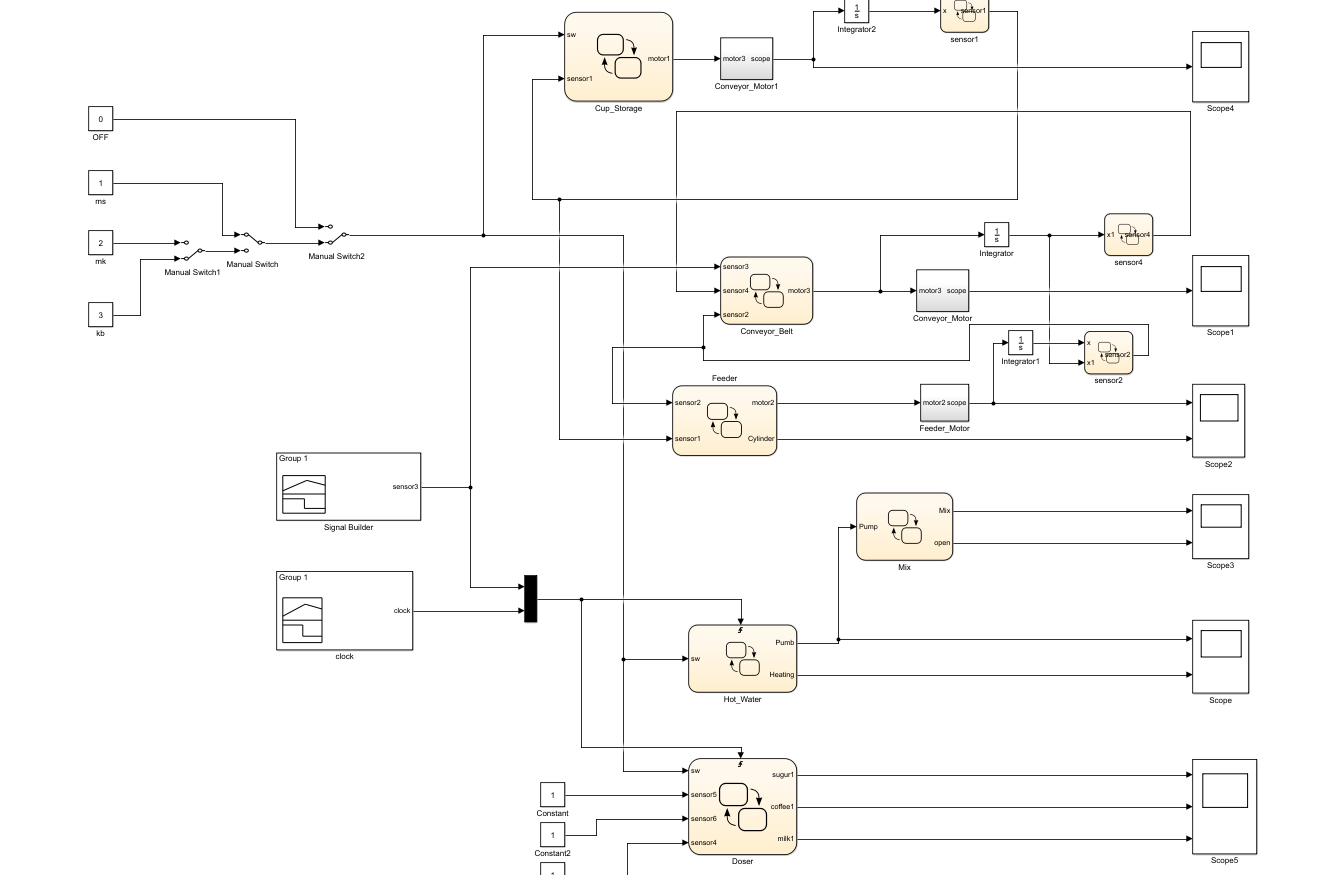
# 三. 第二周总结

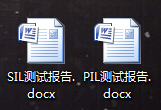
这周接着上周的工作，将分散的各个模块组合到一起，然后进行调试，最终得到了自动咖啡机的stateflow模型：



经过老师与助教的帮助下，将逻辑调通，并且按照要求自动生成了c代码：

# 四. 第三周总结

这周完成了任务的收尾工作；由于上周提交的模型缺少物理模块，所以这周建了咖啡机的物理模块，与之前的stateflow结合起来，形成了更加真实的仿真：

然后通过模型，进行了Software in Loop测试Processor in Loop测试，验证了各个模块的可行性：

# 五. 个人小结

**万学远**

这个为期三周的大作业让我收获很大，作为队长，我锻炼了自己的组织能力，给每个组员分配的任务要保证每个人都能学到知识，每个人都能有事情做；每周组织一次讨论总结会，大家一起交流遇到的问题与新的想法。  
 同时也学到了simulink的物理建模与stateflow建模，也有了状态机这个概念，遇到复杂的问题，可以把问题分解成一些简单一点的小问题；认识了单元测试的重要性与必要性，因为问题越早发现解决的成本越低，等到快完结的时候发现基础架构出了问题那将会导致灾难性的后果；也尝试了一下单元测试的工具和流程，然后就是自学能力与资料查阅能力也得到了培养，任务中很多东西是需要自己查找文献然后自学的。这些都要感谢老师和助教的帮助！

**谢鑫磊**

实时软件控制的课程进入了尾声，说实话，这个课程是大四的课程中最累的一门，特别是我基础不够好，所以我只能够说应付了作业就得费很大的劲。

这门课和大四的所有课程一样，是一个管中窥豹的内容，如果想要认真的学好和掌握，可能要么以前做过类似的项目，要么得花很多时间，甚至把这个作为自己研究生的方向。

最后的大作业是用matlab做基于模型的软件开发，小组的分工等内容要明确，其中一定要重视概念设计，虽然麻烦，但是可以避免之后很多的弯路。经过这次的课程，只能说自己对软件工程有了更为深层次的理解，但由于基础并不好，所以到最后也没有用到发的板子，比较遗憾，但通过几次焦头烂额的学习和助教的几次讨论，自己还是成长和收获了不少

**潘捷**

这次大作业，以MATLAB软件为基础，用stateflow搭建状态控制，用simulink做物理模型，基本完成陈冰老师布置的自动咖啡机模型。

这次作业强调了团队协作，各自都有自己的任务，在实践中也去学习新知识。我负责的是Hot\_Water模块。

模块的搭建要立足于实际生活，不能为了某个功能而凭空设置状态；在阅读官方文档帮助时，也更加明白英语的重要性，好的英语能让你更快的找寻自己所需；队友之间勤交流，说出自己想法，完善模型。总之，在这个大作业中，学到了很多。

**张世舜**

三周前我们结束了实时控制软件设计课程，这周我们组完成了课程大作业，整个课程学习算是告一段落了，但我觉得之后的学习还远远没有结束，在这门课上学到的很多软件工程的构建思想和方法还将在接下来的学习、工作中不断地延续下来。在这门课的学习过程中我感到受益良多，所收获的，不仅是知识，更重要的是在思考问题和解决问题上的能力，同时也对一名“职业”的工程师应具备的品质有了更加深刻的认识。

我们学习了基于模型的设计，stateflow状态机建模，基于模型的测试，SIL和PIL测试等，在课堂上，老师为我们实际演示了stateflow状态机控制器建模的过程，在课后，我们也结合mathworks官网上的教程，学习了基于simulink/stateflow控制逻辑和算法的设计。在学习了软件的基本操作和一些工具的使用之后，我们便开始了团队大作业的进程。这次大作业，共分为六个阶段：软件设计规格书编写→咖啡机物理模型构建和概要设计说明书编写→控制软件模块stateflow模型构建→模块C代码自动生成→SIL测试和PIL 测试→项目总结和软件说明文档编写。我们组在做团队大作业过程中总体来说还是做的不错的，从一开始的设计方案讨论、小组成员分工到后面各自就所负责模块进行的模型、控制逻辑设计和功能实现，再到之后的物理模型的构建、代码生成、sil和pil测试我们都较好地完成了，但在这个过程中也存在着一些不足，比如有些阶段没有按时完成，在一个阶段的最后才提交；小组成员对自己的任务不够明确；组员之间的交流还不够多等等。这也给我们之后进行团队合作提供了很多经验，希望之后能在之后有所加强。

最后，要感谢陈老师、助教学长和小组成员们。陈老师非常鼓励我们进行实践和阅读，示范了我们最好的学习方式，强调“做中学”，除课堂讲授外，还让我们通过编程练习和项目实践相结合的方式，掌握软件工程的原则和方法，并推荐给我们《构建之法》、《嵌入式实时操作系统》等书籍，让我们能够进行更加深入的学习。还要感谢助教洪星学长，在课后认真负责地为我们批改作业，并不厌其烦的回答我们的问题，帮助我们解决各种困难。最后要感谢小组成员，我们一起合作、交流，解决了一个个问题，同时我也在跟组员们讨论的过程中学习到了很多思考问题和解决问题的方法。

**王振鸿**

本次团队作业中，最主要的收获是熟悉了Simulink下基于模型的嵌入式软件设计开发流程。  
主要过程有：  
建立Simulink下实物的模型——代替实物测试，减少成本，提高设计测试效率；  
使用stateflow表达控制逻辑——比直接码代码更直观高效的图形化操作；  
基于模型的调参——实现要求的输入与输出之间的函数关系，比实物调参更准确、便捷、高效；  
进行SIL测试——图形化逻辑转为高级语言模块，测试转化代码的功能是否正确；  
进行PIL测试——配置处理器，生成可执行的机器语言模块，烧入处理器测试执行，与模型结果对比测试。还可以测试处理器执行效率（本次未做到）。