

验证自动化流程设计说明

孙林涵，诸人豪，张煜

2025 年 6 月 9 日

摘要

本文档主要介绍本软件的设计思路、实现方法及相关注意事项。

1 介绍

本项目设计了一套由大语言模型（Large Language Model, LLM）驱动的 RTL 代码的生成与验证自动化软件，旨在提高 IC 设计的效率和准确性。

2 Agent 设计

软件通过设置项目管理、设计工程师和验证工程师三个 Agent，模仿一般的 IC 设计流程，使通用 LLM 生成可靠的 RTL 代码。下面，介绍各个 Agent 的设计。

2.1 项目工程师 Agent

项目工程师解读用户的需求，生成项目的技术规范（下文简称 Spec），并将其存储在文件中。

当设计完成时，项目工程师还会根据验证工程师的验证报告，检查设计是否满足规范要求。

2.2 设计工程师 Agent

设计工程师根据项目工程师提供的 Spec，生成 RTL 代码。设计工程师会将生成的代码存储在文件中，并在必要时进行修改。为避免 LLM 生成的代码存在语法错误，每次设计工程师提交代码时，都会使用 VCS 的 `vlogan` 工具进行语法检查。若检查出语法错误，设计工程师会根据错误信息进行修改，直到消除所有报错为止。

另外，设计工程师还会根据验证工程师的反馈，修改 RTL 代码以满足验证需求。

2.3 验证工程师 Agent

验证工程师根据项目工程师提供的 Spec 与用户提供的验证方案，生成验证计划，并编写测试用例。验证工程师会使用 VCS 编译仿真程序，并运行测试用例，生成验证报告。同样，当编译仿真程序时，若出现编译错误，验证工程师也会根据错误信息进行修改，直至成功编译得到 simv 仿真程序为止。接下来，验证工程师会检查 simv 仿真程序的输出，判断 RTL 代码是否存在问题。当 RTL 代码存在问题时，验证工程师会将问题反馈给设计工程师，并要求其修改 RTL 代码。当验证工程师认为 RTL 代码满足验证需求时，会将验证报告提交给项目工程师。

3 运行流程

接下来，介绍本软件是如何调用各个 Agent 以完成 RTL 代码的生成与验证的。

图1展示了本软件的运行流程。首先，项目工程师 Agent 会解读用户需求，生成 Spec 并存储在文件中。接着，设计工程师 Agent 会根据 Spec 生成 RTL 代码，并进行语法检查。若存在语法错误，设计工程师会进行修改。然后，验证工程师 Agent 会根据 Spec 与验证方案生成验证计划，并编写测试用例。验证工程师会编译仿真程序并运行测试用例，生成验证报告。当 RTL 代码存在问题时，验证工程师会将问题反馈给设计工程师，设计工程师会修改 RTL 代码并重新提交。最终，当验证工程师认为 RTL 代码满足验证需求时，会将验证报告提交给项目工程师。项目工程师确认 RTL 代码满足规范要求后，整个流程结束。

为了避免设计工程师的错误误导验证工程师，设计工程师和验证工程师将分别根据项目工程师提供的 Spec 与用户提供的验证方案生成各自的代码和测试用例。整个流程中多次出现的语法检查和编译步骤，确保了生成的代码和测试用例都是正确的。

4 总结

本项目设计了一套由 LLM 驱动的 RTL 代码的生成与验证自动化软件，实现了 IC 设计流程的自动化，能够快速产出原型设计。然而，在设计中，我们发现，一般的通用大模型在数字电路相关任务上、尤其是验证任务上，存在相当大的局限性。要进一步减少本项目的迭代次数，提升设计效率，必须对大模型进行针对性的训练，以使其能够更好地理解数字电路的设计与验证流程。本项目产出的原型设计，也应由专业的设计工程师和验证工程师进行进一步的修改与验证，以确保其满足实际的设计需求。

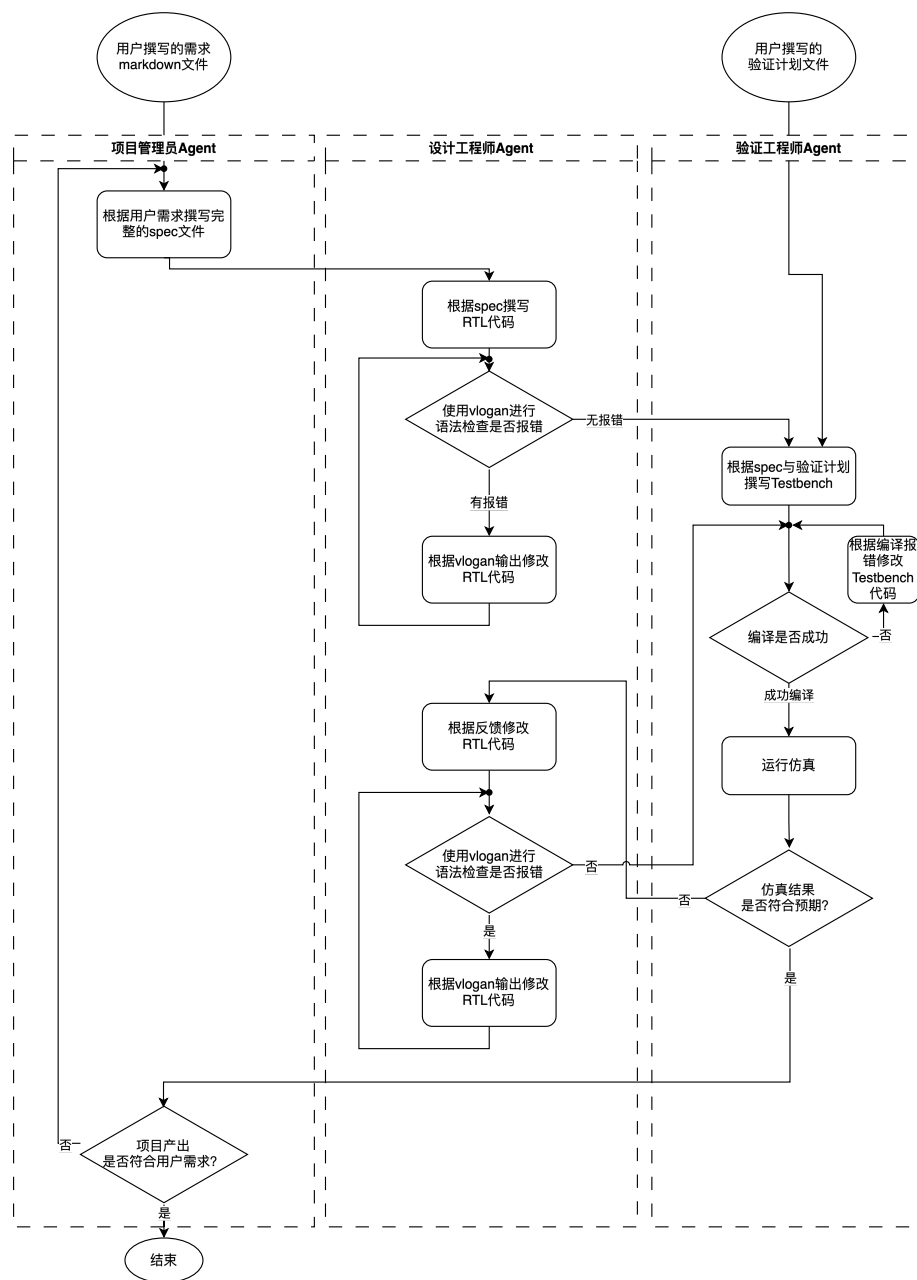


图 1: 程序流程图