

项目申请书 - 为RT-Thread提供通用蓝牙协议栈能力

我是来自华中科技大学软件学院研究生二年级的严立能，本次申请的**项目名称**为：为RT-Thread提供通用蓝牙协议栈能力。

对于该项目，我的**解决方案**如下：

- 1. 对接 NimBLE HCI层：主要使用RT-Thread 中UART设备驱动模块对接 NimBLE HCI层，完成对大部分的RT-Thread 上的bsp开发板的适配，且只要支持UART就能支持蓝牙协议栈 搭配蓝牙 controller卡片。
- 2. 添加 NimBLE 对外部蓝牙controller芯片的初始化支持：主要参考BTstack中初始化架构完成构建，完成部分外部蓝牙controller芯片的初始化支持；这一部分尽可能解耦 HCI传输接口 与 初始化实际数据内容，实现用统一的初始化 HCI 传输调用过程，便于后期扩展其他芯片初始化功能。
- 3. 完善 NimBLE app功能：为蓝牙协议栈提供更多可以跑的 example 例程，主要是学习 NimBLE app编程，并完成在 RT-Thread 下的功能运行测试。

我的**项目开发时间计划**如下：

1. 前期（7.1 - 7.14）（两周）

此阶段主要是项目了解、知识学习阶段，了解 HCI Transport 的 H4 相关知识，进一步学习 NimBLE 的 HCI 接口。学习 BTstack 的外部蓝牙芯片支持架构，了解 AP6212 蓝牙芯片的初始化流程。了解 NimBLE App 编程。上述内容形成学习文档。

时间	任务
7.01 - 7.08	学习H4 相关知识，了解 NimBLE HCI 接口。
7.08 - 7.15	学习 BTstack 蓝牙control 初始化架构，了解AP6212初始化流程。文档整理。

2. 中期（7.15 - 8.14）（四周）

此阶段主要是使用RT-Thread UART设备驱动 完成 NimBLE HCI层 对接；搭建外部蓝牙Controller芯片初始化支持架构；完成 AP6212 芯片的初始化支持；视情况扩展其他芯片初始化支持。

时间	任务
7.15 - 7.22	使用RT-Thread UART设备驱动，完成 NimBLE HCI层对接。
7.23 - 8.14	建立Controller初始化框架，实现外部蓝牙 Controller 芯片支持，初始化 AP6212 芯片。

3. 后期（8.15 - 9.30）（六周）

此阶段的工作内容包括：完善 NimBLE app samples 在RT-Thread下的运行与测试；测试软件包 在 RT-Thread Studio上的正常编译运行；测试 NimBLE 的 Controller 初始化功能；将代码整合到主分支。

时间	任务
8.15 - 9.14	完善 NimBLE app samples
9.14 - 9.22	使用 RT-Thread Studio 正常编译运行 NimBLE；测试 NimBLE的 Controller初始化。
9.23 - 9.30	整理代码，提交到github仓库主分支。

项目过程中以及项目结项后的对个人的一些期望

首先养成记录学习过程的习惯，尝试搭个博客，分享个人学习成果；无论在后面学习或是工作的时间中，也能坚持积极参加开源活动，活跃在开源项目以及开源社区中；拓展视野，锻炼能力，更好地提升自我。