
目录

1. 准备工具	1
1.1 软件	1
1.2 硬件	1
2. 硬件连接	1
2.1 供电	1
2.2 调试器连接	1
3. 导入工程	2
4. 编译工程	2
5. 下载与调试	2
5.1 配置调试器	2
5.2 直接运行	2
6. 验证效果	2
7. 常见问题	3
7.1 编译提示找不到 SystemInit	3
7.2 下载失败，提示无法连接目标	3
7.3 程序下载后 LED 不亮	3
7.4 程序跑飞或时钟异常	3
8. 下一步	3

SP-CORE-F103 快速入门

本指南帮助你从零开始，在 SP-CORE-F103 上运行第一个 LED 流水灯程序。

1. 准备工具

1.1 软件

- STM32CubeIDE: ST 官方免费 IDE，集成 ARM GCC 工具链和调试器。
- 下载地址: <https://www.st.com/en/development-tools/stm32cubeide.html>

1.2 硬件

- SP-CORE-F103 开发板 ×1
- USB Type-C 线 ×1 (供电)
- ST-Link V2 / DAP-Link / J-Link 调试器 ×1
- 杜邦线若干

2. 硬件连接

2.1 供电

通过 USB Type-C 线连接电脑或充电器，板载红色 LED (D5) 旁边的电源指示灯不会自动亮，需程序控制点亮。

2.2 调试器连接

开发板	调试器
3V3 (可选)	VCC
SWDIO	SWDIO
SWCLK	SWCLK
RST	RST
GND	GND

提示: 开发板可通过 USB 自供电，因此调试器的 VCC 可以不接，只接 SWDIO/SWCLK/RST/GND。

3. 导入工程

1. 打开 STM32CubeIDE。
2. 选择 File → Import...。
3. 在弹出的对话框中选择 Existing Projects into Workspace, 点击 Next。
4. 点击 Browse..., 选择本 SDK 中的目录: 02-Firmware/SPCORE-F103
5. 勾选 SPCORE-F103 工程, 点击 Finish。

4. 编译工程

1. 在 Project Explorer 中右键点击 SPCORE-F103。
2. 选择 Build Project。
3. 等待编译完成, Console 窗口应显示类似输出: text Finished building: SPCORE-F103.elf

5. 下载与调试

5.1 配置调试器

如果首次使用, 需要确认调试配置:

1. 点击工具栏下拉箭头, 选择 Debug Configurations...。
2. 展开 STM32 Cortex-M C/C++ Application, 选择 SPCORE-F103 Debug。
3. 在 Debugger 标签页确认: - Debug probe: ST-LINK (或 DAP-Link/J-Link 根据实际选择) - Interface: SWD - Frequency: 默认即可
4. 点击 Debug 开始下载。

5.2 直接运行

1. 点击工具栏绿色 Run 按钮 (或按 Ctrl+F11)。
2. 程序将自动编译 (如有改动)、下载并开始运行。
3. 观察开发板: - 4 颗绿色 LED (D1~D4) 会依次点亮。 - 最后红色 LED (D5) 点亮。 - 然后循环往复。

6. 验证效果

如果一切正常, 你应该看到:

D1 亮 → D2 亮 → D3 亮 → D4 亮 → D5 亮 → 回到 D1...

每颗 LED 点亮约 150ms，形成流水灯效果。

7. 常见问题

7.1 编译提示找不到 `SystemInit`

现象：undefined reference to `SystemInit`

解决：确保 `Src/system_stm32f1xx.c` 已加入工程并编译。本 SDK 已包含该文件，如仍报错，请检查 `.cproject` 的 `sourceEntries` 是否包含 `Src`。

7.2 下载失败，提示无法连接目标

现象：No target connected 或 Cannot connect to target

解决： - 检查 SWDIO/SWCLK/GND 是否接好。 - 检查调试器驱动是否安装。 - 按住复位键，点击 Debug，松开复位键（必要时使用 Connect Under Reset）。 - 检查 BOOT0 是否为低电平（从 Flash 启动）。

7.3 程序下载后 LED 不亮

解决： - 确认已通过 USB Type-C 给开发板供电。 - 确认编译的是 `Debug` 配置。 - 确认下载成功，点击 Run 让程序运行。 - 用万用表测量 PB12~PB15/PC13 是否有高低电平变化。

7.4 程序跑飞或时钟异常

解决： - 检查 8 MHz 晶振是否焊接正常。 - 如果不需要 72 MHz，可简化 `system_stm32f1xx.c`，直接返回，使用默认 8 MHz HSI。

8. 下一步

- 阅读 [用户手册](#) 了解完整硬件资源。
- 查看 [引脚定义](#) 规划你的外设连接。
- 修改 `Src/main.c`，尝试控制更多 GPIO 或添加串口输出。