

ucore 操作系统使用说明

(v0.01)

ucore 是一个清华大学教学用的操作系统，其只使用到了一些基本的 MIPS 指令，有关它的介绍请见大赛资源发布包 *ucore_thumips/docc/ucore_thumips.pdf*。至于如何使用 pmon 启动内核，请参见 *FPGA_test/FPGA_soc_test/pmon 运行并load 内核启动的方法.pdf*。

在此，非常感谢清华大学提供出来的 ucore 资源和移植过程中给予的帮助。

1 ucore 系统运行基本要求

为了在开发板上使用自实现 MIPS 微系统中运行 ucore 系统，需要具备以下条件：

- (1) 较完备的 CPU：除了大赛要求实现的指令集外，还需实现 TLB 指令和 TLB 模块，Cache 指令和 Cache 模块可不实现。
- (2) 较完备的 SoC：集成 DDR3 控制器以连接 DDR3 颗粒作为内存；集成串口控制器以进行交互；其他可选模块可扩展 SoC 的功能，如网口、VGA 等。
- (3) 交叉编译工具链，以编译 ucore，可使用大赛资源发布包里提供的工具。

2 ucore 系统编译

进入 Linux 环境，在 *ucore_thumips/*下输入 make 命令编译。结束后，obj/ucore-kernel-initrd 就是所要的 ucore 内核文件。

需要注意的是，obj 目录下已经有一个能正常使用的内核 ucore-kernel-initrd。除非需要修改内核源代码，否则这个编完的文件足以供大赛使用。

3 ucore 系统运行

可以先运行 PMON，再通过网口加载 ucore 至内存上运行，该方法参见 *FPGA_test/FPGA_soc_test/pmon 运行并load 内核启动的方法.pdf*。该方法已在 SoC_up 上验证通过，但需要 SoC 很完备，且需要先调通 pmon 的运行。

由于 ucore 编完的文件很小，也可以考虑直接放在 xilinx ip 的 ram 中进行运行。该方法需要将 ucore 编完的 elf 文件转换为 ram 的初始化文件 coe 文件，转换方法可以参考性能测试或功能测试里的 Makefile 的写法。该方法对 SoC 要求更低，且不需要先调通 PMON，但目前我们未进行尝试。

预赛并不要求启动操作系统，如果参赛者有兴趣，可以自行尝试移植 ucore 系统。