

K230 QEMU 建模

洛佳，泽文

qemu-system-riscv64 -nographic -M virt -bios <?>

- 1. default (opensbi, ?)
- 2. /path/to/binary



qemu-system-riscv64 -nographic -M k230

[K230_DEV_BOOTROM] = { 0x91200000, 0x00010000 },

目前的模拟



真板



解密, 闪存读取,
SD卡.....

DDR training/
DDR init (零阶段引导)

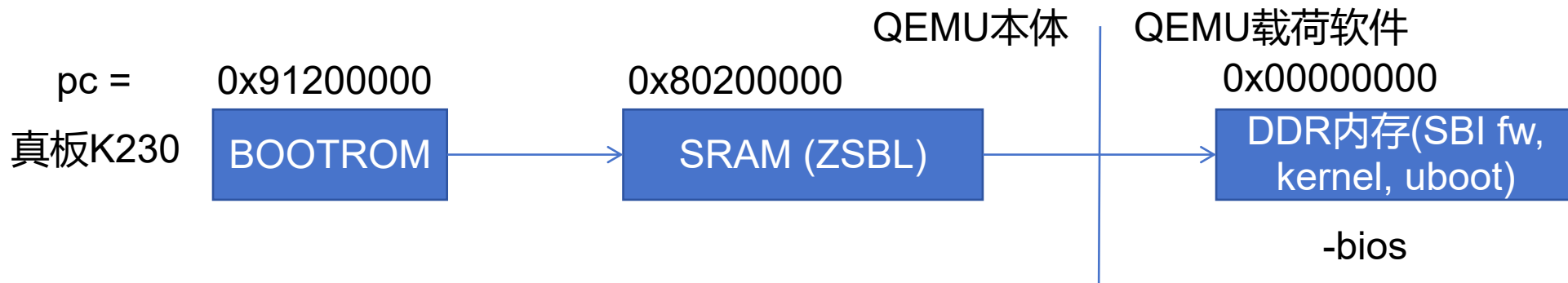
```
/* Mask ROM reset vector */
uint32_t reset_vec[] = {
    /* 0x91200000: auipc t0, 0x0          */ 0x00000297,
    /* 0x91200004: addi t0, t0, 36 # <trap> */ 0x02428293,
    /* 0x91200008: csrw mtvec, t0          */ 0x30529073,
    /* 0x9120000C: csrr a0, misa          */ 0x301012F3,
    /* 0x91200010: lui t0, 0x1            */ 0x000012B7,
    /* 0x91200014: slli t0, t0, 1          */ 0x00129293,
    /* 0x91200018: and t0, a0, t0          */ 0x005572B3,
    /* 0x9120001C: bnez t0, loop           */ 0x00511063,
    /* entry:                             */
    /* 0x91200020: addiw t0, zero, 1        */ 0x0010029b,
    /* 0x91200024: slli t0, t0, 0x1b        */ 0x01b29293,
    /* 0x91200028: jr t0 # uboot 0x80000000 */ 0x00028067,
    /* loop:                               */
    /* 0x9120002C: j 0x9120002C # <loop>    */ 0x0000006f,
    /* trap:                               */
    /* 0x91200030: j 0x91200030 # <trap>    */ 0x0000006f,
};
```



- 1. FIT format U-Boot (U-Boot)
- 2. Kernel, DTB (非U-Boot)

Address space mapping

Start Address	End Address	Function	Size
0x0000_0000	0x8000_0000	DDRC	2GB
0x8000_0000	0x8020_0000	KPU L2 Cache	2MB
0x8020_0000	0x8040_0000	SRAM	2MB
0x8040_0000	0x8040_0800	KPU Congfiguration	2KB
0x8040_0800	0x8040_0C00	FFT	1KB



- 复制原厂BootROM? 运行完整bootrom, 外设模拟量大
- 极简版BootROM (跳过ZSBL, 不初始化DDR)
 - 假设此时模拟环节从SRAM运行完成后开始; QEMU需要提供DTB (a1寄存器)



- 折中版做法 (从ZSBL开始模拟) 优点是QEMU K230具有辅助ZSBL开发的功能
 - (无法测试ZSBL的DDR初始化功能, 但是开发ZSBL够用)
 - 需要建模闪存和SD卡等外设

K230 QEMU建模的三种方法

- 极简版做法→折中版做法→复刻原厂BootROM
 - 难度逐渐提升，建模工作量逐渐增加，但是模拟逐渐变准确，功能逐步增加
- 折中版做法建模需求：SD卡、闪存等简单存储外设（可以是近期目标）
 - 功能：可以开发ZSBL，比极简版要好
 - 也不至于建模需求太大
- QEMU提供DTB的做法
 - 最终跳转到DDRC基地址，也就是0x08000000（PC寄存器的内容）
 - 需阅读SDK确认QEMU本体行为，每个核进哪个地址
 - 根据SBI规范，a0和a1寄存器必须设定为hartid和opaque
 - a0寄存器写入我们希望的QEMU本体的hartid，和SDK对齐
 - a1寄存器写入DTB的基地址（SDK中查询）假如SDK不提供那就不写了
 - 考虑是否支持Dynamic Firmware（a2 DynamicInfo，如果要支持，a0, a1, a2都得有）