

循序渐进，学习开发一个 RISC-V 上的操作系统



第 15 章 软件定时器

汪辰

- 软件定时器的分类
- 软件定时器的设计

➤ 软件定时器的分类

➤ 软件定时器的设计

- **硬件定时器：**芯片本身提供的定时器，一般由外部晶振提供，提供寄存器设置超时时间，并采用外部中断方式通知 CPU，参考第 12 章介绍。优点是精度高，但定时器个数受硬件芯片的设计限制。
- **软件定时器：**操作系统中基于硬件定时器提供的功能，采用软件方式实现。扩展了硬件定时器的限制，可以提供数目更多（几乎不受限制）的定时器；缺点是精度较低，必须是 Tick 的整数倍。

➤ 按照定时器设定方式分：

- 单次触发定时器：创建后只会触发一次定时器通知事件，触发后该定时器自动停止（销毁）
- 周期触发定时器：创建后按照设定的周期无限循环触发定时器通知事件，直到用户手动停止。

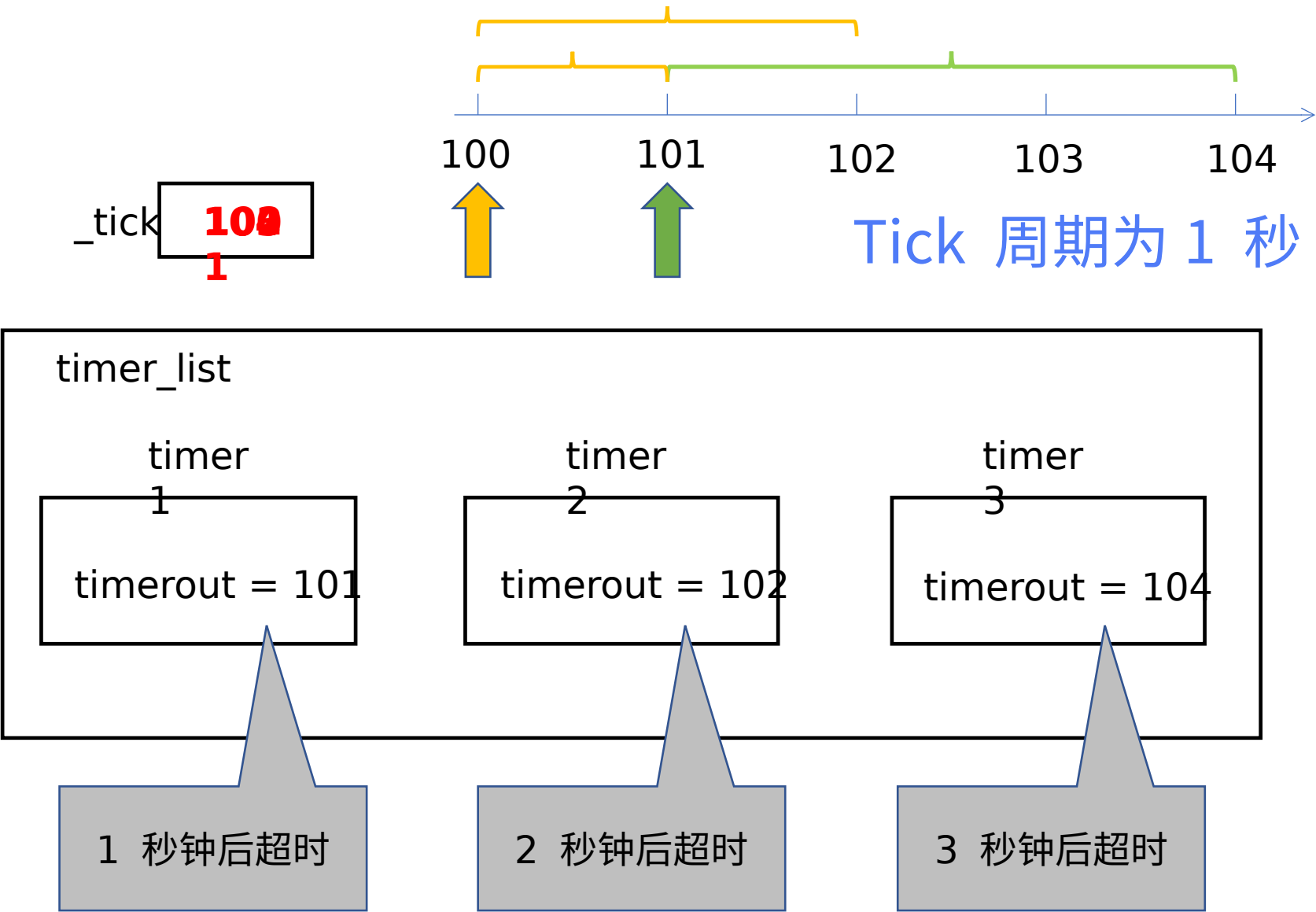
➤ 按照定时器超时后执行处理函数的上下文环境分：

- 超时函数运行在中断上下文环境中，要求执行函数的执行时间尽可能短，不可以执行等待其他事件等可能导致中断控制路径挂起的操作。优点是响应比较迅速，实时性较高。
- 超时函数运行在任务上下文环境中，即创建一个任务来执行这个函数，函数中可以等待或者挂起，但实时性较差。

➤ 软件定时器的分类

➤ 软件定时器的设计

```
/* software timer */
struct timer {
    void (*func)(void *arg);
    void *arg;
    uint32_t timeout_tick;
};
extern struct timer *timer_create(void (*handler)(void *arg), void *arg, uint32_t timeout);
extern void timer_delete(struct timer *timer);
```





谢谢

欢迎交流合作