

Timer & RTC

汇报人：王俊鹏

汇报时间：2026.10.21

01 为什么需要Timer



北京大学
PEKING UNIVERSITY

CPU 轮询:

程序不断检查时间是否到达，CPU 会被重复检查占用；程序复杂时，检查间隔就可能变化，定时任务的时间也会受到影响



Timer 定时:

Timer 按设定周期独立计数，到期后发出中断，通知 CPU 执行相应任务。等待期间，CPU 可以处理其他工作。

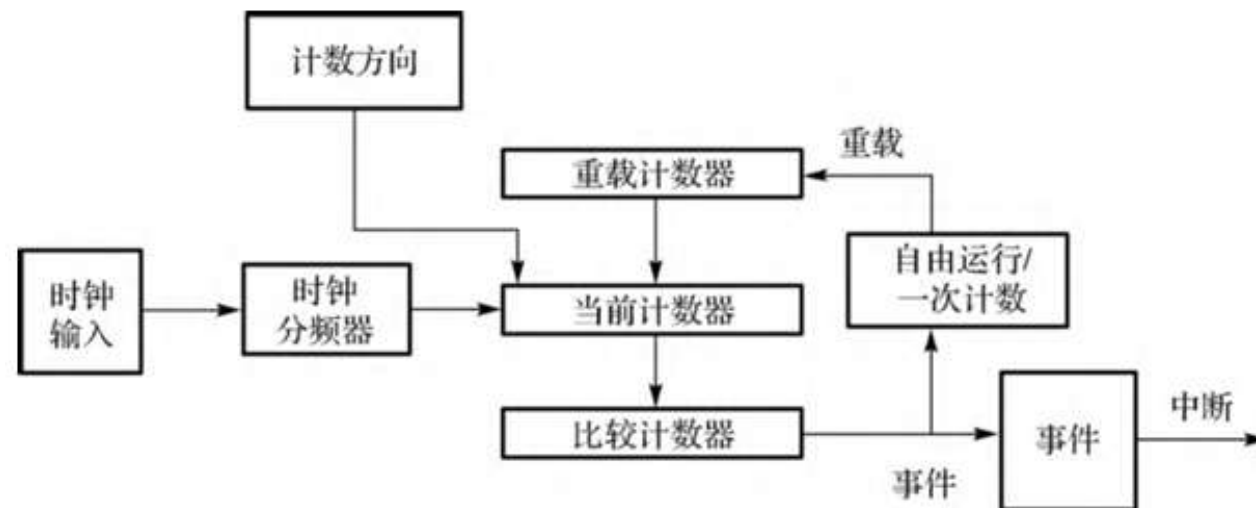


02 Timer的硬件原理



北京大学
PEKING UNIVERSITY

- **时钟源与分频器：** 把输入时钟转换成计数器的计数节拍。
- **当前计数器：** 按设定方向递增或递减。
- **比较与触发：** 计数值达到设定条件后，产生定时事件。
- **事件与重载：** 事件可以触发中断；周期模式重载后继续计数，单次模式则停止。



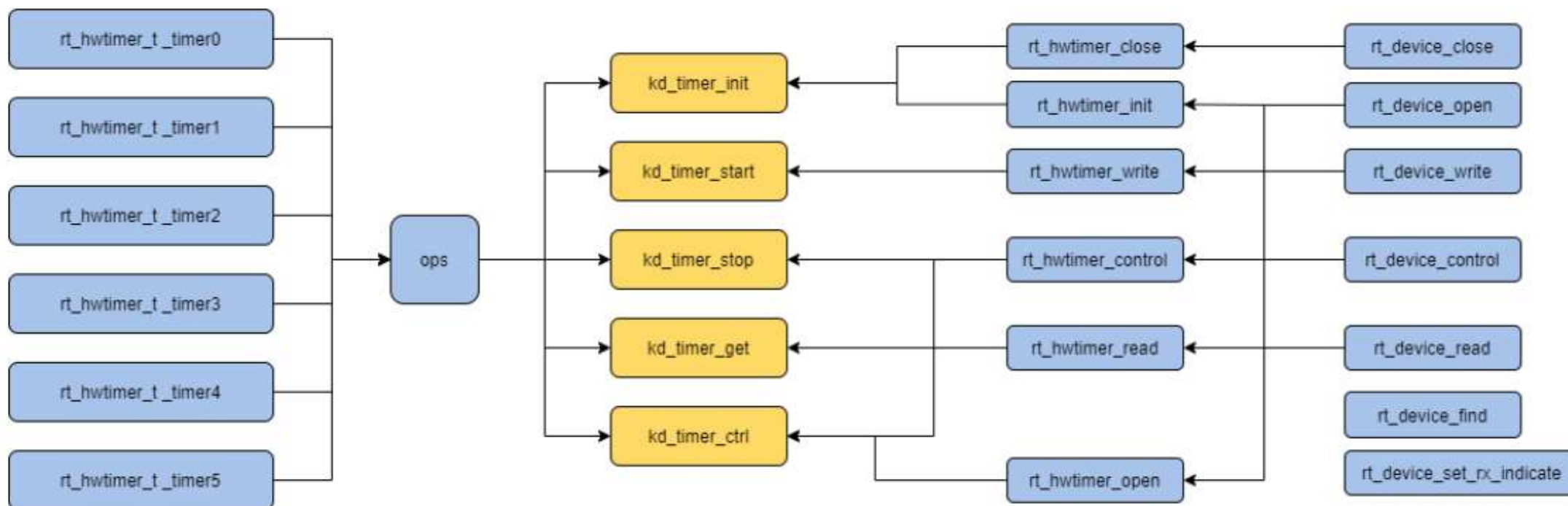
常见的定时器结构框图

03 K230上的Timer



- **硬件设备**: timer0 ~ timer5 表示K230有的6个硬件定时器设备 (还有一个Timer(-1)的软件定时器)
- **通用接口**: 应用通过 rt_device_* 接口查找、打开、配置、读写和关闭 Timer。
- **驱动分发**: 将通用操作经 ops 分发到对应的 kd_timer_* 驱动函数。
- **硬件控制**: 驱动完成定时器初始化、启动、停止、计数读取和控制; 超时后可通过回调通知应用。

TIMER 驱动结构如下图所示:



04 K230与K210 Timer API对比



北京大学
PEKING UNIVERSITY

K230 API	参数字母	含义与示例
machine.Timer(id)	id	定时器编号。当前文档中 I=-1 表示软件定时器，I=0...5 表示硬件 Timer。你现有例程用 Timer(-1)。
tim.init(mode, freq, period, callback)	mode	mode, 工作模式: Timer.ONE_SHOT 单次触发; Timer.PERIODIC 周期触发。
	period	period, 定时周期, 单位为毫秒; 例如 P=1000 表示 1000 ms。
	callback	callback, 超时回调函数, 函数需接收一个 Timer 参数, 例如 callback=fun。
	freq	freq, 可选的触发频率, 单位 Hz。若同时设置 freq 和 period, 当前文档规定 freq 优先。
tim.deinit()	无	停止定时器并释放资源。

05 K230与K210 Timer API对比



北京大学
PEKING UNIVERSITY

K210 API	参数字母	含义与示例
plic_init()	无	初始化 PLIC 中断控制器。
sysctl_enable_irq()	无	开启 CPU 全局中断；注册了 Timer 中断后，CPU 才能响应。
timer_init(Dev)	Dev	Timer 设备编号
timer_set_interval(Dev,Ch,Time)	Dev	Timer 设备编号。
	Ch	Timer 通道编号，Ch对应通道 0...3。
	Time	定时间隔，单位纳秒。例如 T=500000000 表示 500 ms。函数返回实际设置的间隔，
timer_irq_register(Dev,Ch,S,PRI,Callback,CTX)	Dev, Ch	要注册中断的 Timer 设备和通道。
	S	是否单次触发：S=1 为单次，S=0 为重复触发；你的示例里这两个值分别用于单次和周期模式。
	PRI	中断优先级，例如示例中的 1。
	Callback	中断回调函数，类型为 int callback(void *ctx)。
	CTX	传给回调函数的上下文指针；回调中可通过它识别当前 Timer/通道。
timer_set_enable(Dev,Ch,Enable)	Enable	使能开关：E=1 启动，E=0 停止。
timer_irq_deregister(Dev,Ch)	Dev, Ch	注销指定设备和通道的中断回调。

06 为什么需要RTC



北京大学
PEKING UNIVERSITY

RTC (Real Time Clock)

维护年月日、时分秒，可通过 Tick 或闹钟中断响应时间事件，适用于时间显示、定时唤醒和事件记录。

TIMER (定时器)

测量时间间隔，到达设定周期或计数值时触发中断，适用于延时、周期采样和定时任务。

● RTC：现在是几点？

维护年月日和时分秒；可由 Tick 或闹钟事件触发中断

● TIMER：过了多久？

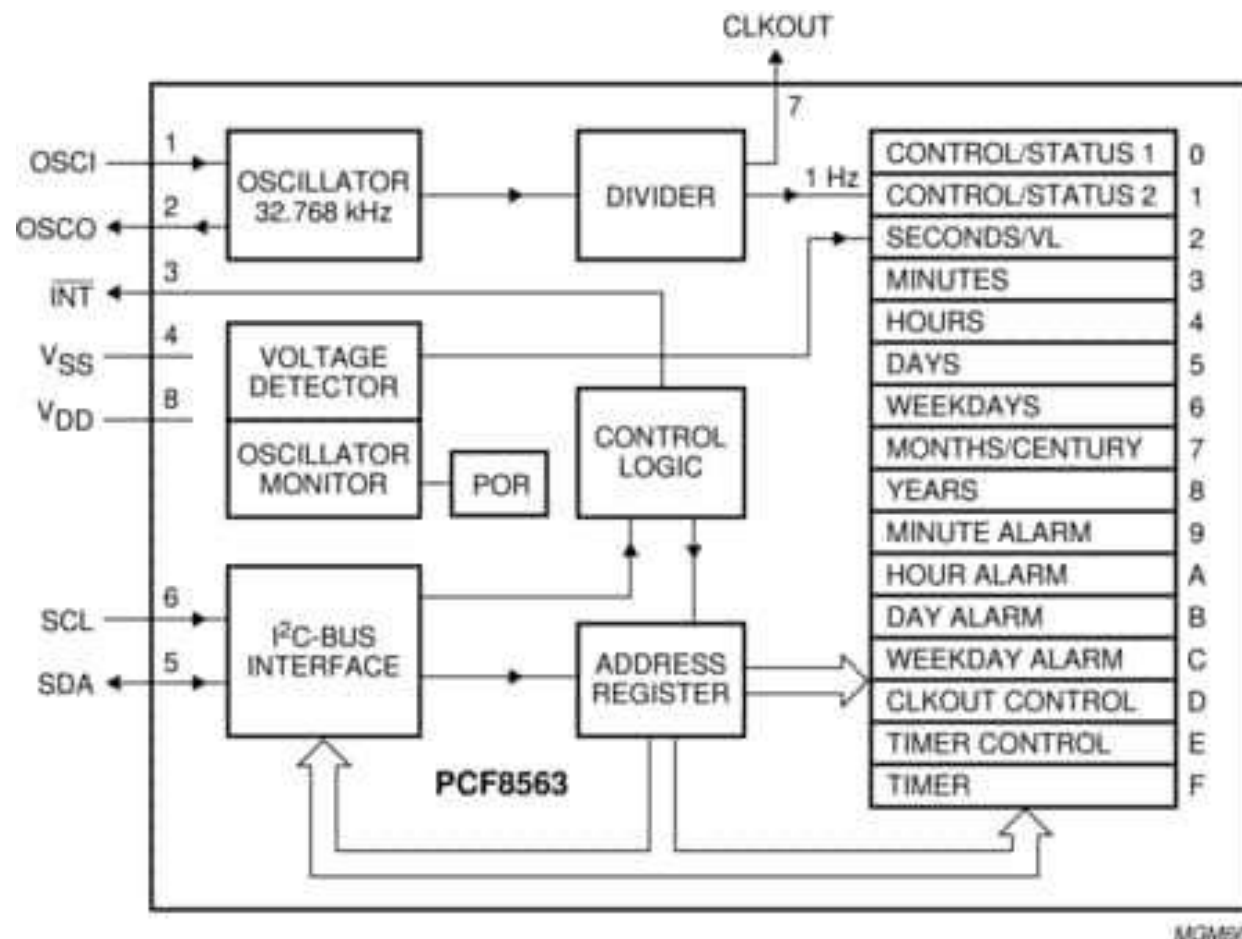
测量时间间隔；超时或周期到达时可触发中断

07 RTC的硬件原理



北京大学
PEKING UNIVERSITY

- **时钟产生与分频模块**：外接晶振产生稳定的时钟信号，经分频电路转换为 1 Hz 计时脉冲。
- **时间计数与存储模块**：通过级联计数器实现年月日持续计数
- **闹钟与定时模块**：通过预设闹钟寄存器及定时器，实现时间比较中断触发。
- **控制与通信模块**：通过 I²C 总线实现时间数据读取及寄存器配置。
- **电源与振荡监测模块**：通过电压检测、上电复位及振荡监测电路，保障芯片初始化和时钟运行状态的可靠性。



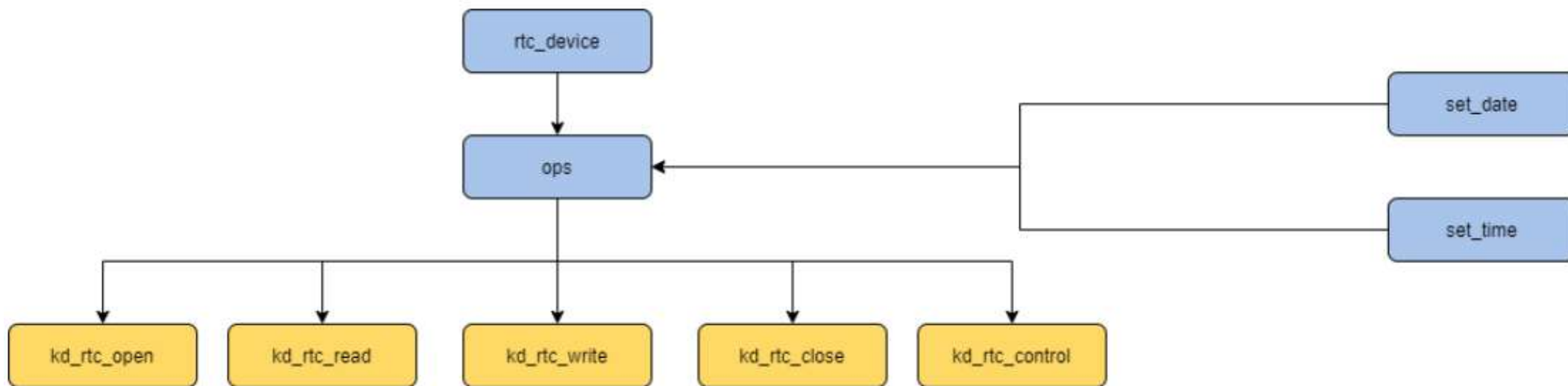
某RTC芯片结构图

08 K230上的RTC

- **设备抽象**：rtc_device 表示注册的 RTC 设备。
- **操作分发**：ops 将上层请求分发给对应的 RTC 驱动函数。

- **读写与控制**：kd_rtc_read/write 读取或写入时间数据；kd_rtc_control 执行 RTC 控制操作；open/close 管理设备使用。
- **时间设置**：set_date 和 set_time 分别设置日期与时分秒，再由底层驱动完成配置

RTC 驱动结构如下图所示：



09 K230与K210 RTC API对比



北京大学
PEKING UNIVERSITY

K230 API	参数字母	含义与示例
rtc = machine.RTC()	无	创建 RTC 对象，后续通过该对象读写时间。
rtc.datetime()	无	读取当前时间，返回 (年, 月, 日, 星期, 时, 分, 秒, 微秒)。
rtc.datetime(Date)	Date	设置时间。Date是上述格式的 8 项元组。例如： rtc.datetime((2024, 1, 1, 0, 0, 0, 0, 0))。按顺序分别表示 (年, 月, 日, 星期, 时, 分, 秒, 微妙) 星期字段 使用 0-6 表示星期一至星期日。
rtc.ntp_sync()	无	从网络时间服务器校准 RTC，需要设备已联网；返回 True / False 表示同步是否成功。
rtc.timezone(timezone)	timezone	读取或设置时区，时区范围是 -12 到 12

10 K230与K210 RTC API对比

K210 API	参数字母	含义与示例
rtc_init()	无	初始化 K210 RTC 外设。
rtc_timer_set(Y, M, D, h, m, s)	Y, M, D, h, m, s	设置年月日时分秒。例如：rtc_timer_set(2018, 9, 12, 23, 30, 50)。这里的 timer 指 RTC 的日历计时功能，不是通用间隔 Timer。
rtc_timer_get(&Y, &M, &D, &h, &m, &s)	6 个输出指针	读取 RTC 当前年月日、时分秒，函数将结果写入指针指向的变量。
rtc_alarm_set(Y, M, D, h, m, s)	Y, M, D, h, m, s	设置 RTC 的闹钟比较时间
rtc_tick_irq_register(B, Type, Callback, CTX, PRI)	B	是否单次触发；true：触发一次后关闭 Tick 中断；false：持续周期触发。
	Type	Tick 的时间单位/触发频率RTC_INT_SECOND：每秒；RTC_INT_MINUTE：每分钟；RTC_INT_HOUR：每小时；RTC_INT_DAY：每天。
	Callback	中断回调函数;定时事件到达时执行
	CTX	传给回调函数的上下文指针;用于向回调传递用户数据；没有数据时可传 NULL
	PRI	RTC 中断的 PLIC 优先级
rtc_alarm_irq_register(B, Mask, Callback, CTX, PRI)	Mask	闹钟匹配字段选择；它描述的是“比较哪些字段”结构体包含年月日时分秒。字段为 1 表示比较该字段，为 0 表示不比较；

实验演示环节

注：课程网站下载的镜像可能会遇到Timer例程跑不通的情况，到此网站下载并烧录01Studio K230最新镜像即可
https://download.kendryte.com/developer/releases/canmv_k230_micropython/daily_build/?lang=zh