How to check rtc_alarm wakeup in kmsg

Applicable platform:

MSM8953/MSM8952/MSM8996/MSM8998/SDM660/SDM630/SDM450/generic

Issue/problem description:

Usually, If below debug_mask is enabled, you will see "qpnp_rtc_alarm" print in kmsg when an alarm is triggered.

echo 1 > /sys/module/msm_show_resume_irq/parameters/debug_mask

<4>[350.215301] gic_show_resume_irq: 200 triggered qcom,smd-rpm

<4>[350.215301] gic_show_resume_irq: 203 triggered 681b8.qcom,mpm

<4>[350.215301] gic_show_resume_irq: 358 triggered 400f000.qcom,spmi

<4>[350.215301] __qpnpint_handle_irq: 186 triggered [0x0, 0x61,0x1] qpnp_rtc_alarm

However, In new qualcomm platform(8952/8976/8953...), you cannot find out the log in dmesg any more though it is casued by alarm triggered.

Issue Analysis:

Because Linux alarm framework doesn't use qpnp_RTC device to wakeup AP and not register the alarm to qpnp_RTC driver.

Instead of it, the alarm timer will be written to MPM directly before AP power collapse. Therefore, you can only see "smd-rpm" and "qcom,mpm" print logs as below, and no qpnp_rtc_alarm any more.

[177.890131] gic_show_resume_irq: 200 triggered qcom,smd-rpm

[177.890131] gic_show_resume_irq: 203 triggered 601d0.qcom,mpm

......

[201.387648] gic_show_resume_irq: 200 triggered qcom,smd-rpm

[201.387648] gic_show_resume_irq: 203 triggered 601d0.qcom,mpm

Here is the source code, if **"poweron_alarm" is TRUE**, then the **qpnp_rtc driver will not be used**. so you could disable power alarm feature and then can see RTC interrupts during suspend exiting.

echo 0 >/sys/module/qpnp_rtc/parameters/poweron_alarm

kernel\kernel\time\alarmtimer.c

```c
static int alarmtimer_suspend(struct device *dev)

{

.....

if (poweron_alarm) {

struct rtc_time tm_val;

unsigned long secs;

tm_val = rtc_ktime_to_tm(min);

rtc_tm_to_time(&tm_val, &secs);

lpm_suspend_wake_time(secs);

} else {

/* Set alarm, if in the past reject suspend briefly to handle */

ret = rtc_timer_start(rtc, &rtctimer, now, ktime_set(0, 0));

if (ret < 0)

__pm_wakeup_event(ws, MSEC_PER_SEC);

}

......

}
```