RK3308 System suspend configuration guide

ID: RK-KF-YF-004

Release Version: V1.0.0

Release Date: 2019-11-11

Security Level: ☐Top-Secret ☐Secret ☐Internal ■Public

DISCLAIMER

THIS DOCUMENT IS PROVIDED "AS IS". FUZHOU ROCKCHIP ELECTRONICS CO., LTD. ("ROCKCHIP")DOES NOT PROVIDE ANY WARRANTY OF ANY KIND, EXPRESSED, IMPLIED OR OTHERWISE, WITH RESPECT TO THE ACCURACY, RELIABILITY, COMPLETENESS,MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT OF ANY REPRESENTATION, INFORMATION AND CONTENT IN THIS DOCUMENT. THIS DOCUMENT IS FOR REFERENCE ONLY. THIS DOCUMENT MAY BE UPDATED OR CHANGED WITHOUT ANY NOTICE AT ANY TIME DUE TO THE UPGRADES OF THE PRODUCT OR ANY OTHER REASONS.

Trademark Statement

"Rockchip", "瑞芯微", "瑞芯" shall be Rockchip's registered trademarks and owned by Rockchip. All the other trademarks or registered trademarks mentioned in this document shall be owned by their respective owners.

All rights reserved. ©2019. Fuzhou Rockchip Electronics Co., Ltd.

Beyond the scope of fair use, neither any entity nor individual shall extract, copy, or distribute this document in any form in whole or in part without the written approval of Rockchip.

Fuzhou Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian,PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document is used to guide users how to configure RK3308 system standby mode according to product requirements.

Product Version

Chipset	Kernel Version
RK3308	4.4&4.19

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers
Software development engineers

Revision History

Version	Author	Date	Change Description
V1.0.0	Jianhong Chen	2019-05-01	Initial version
V1.1.0	Jianhong Chen	2019-11-11	Support list：add kernel 4.19

Content

RK3308 System suspend configuration guide

1. System Suspend

For all SoC platforms with trust, the work of system suspend is completed in trust. Each platform has different implementation for the system suspend in trust, **so the suspend configuration options between different platforms have no relevance and reference, and this document only applies to the RK3308 platform**.

The system suspend process generally has the following operations: turn off the power domain, module IP, clock, PLL, ddr enter self-refresh, switch the system bus to the low-speed clock (24M or 32K), power off the vdd_arm, configure the wake-up source, etc. In order to meet the needs of different products in suspend mode, the relevant configuration is currently passed to the trust through the DTS node during the kernel startup.

1.1 Driver File

```
./drivers/soc/rockchip/rockchip_pm_config.c
./drivers/firmware/rockchip_sip.c
./include/dt-bindings/suspend/rockchip-rk3308.h
```

1.2 DTS Node

```
rockchip_suspend: rockchip-suspend {
    compatible = "rockchip,pm-rk3308";
    status = "okay";
    // Configure general option
    rockchip,sleep-mode-config = <
        (0
        | RKPM_PMU_HW_PLLS_PD
        | RKPM_DBG_FSM_SOUT
        )
    >;
    // Configure wake-up source
    rockchip,wakeup-config = <
        (0
        | RKPM_GPIO0_WAKEUP_EN
        )
    >;
    // Configuration Power
    rockchip,pwm-regulator-config = <
        (0
        | RKPM_xxx
        )
    >;
    //  Configure reboot
    rockchip,apios-suspend = <
        (0
```

```
        | RKPM_xxx
        )
    >;
};
```

2. DTS Configuration

The currently supported configuration options are defined in:

```
./include/dt-bindings/suspend/rockchip-rk3308.h
```

2.1 Common Configuration

Configure Item:

```
rockchip,sleep-mode-config = <...>;
```

Configure option:

```
// Power off vdd_arm, which needs to be supported by the hardware circuit design
#define RKPM_ARMOFF              BIT(0)
// Turn off VAD module, if you do not need VAD wake-up
#define RKPM_VADOFF              BIT(1)
// Required by default
#define RKPM_PMU_HW_PLLS_PD      BIT(3)
// Turn off the 24M crystal oscillator, which can be enabled in the lowest power
consumption mode. It needs to cooperate with the xxx_32K clock source
configuration
#define RKPM_PMU_DIS_OSC         BIT(4)
// Use the 32K clock source inside the PMU as the system clock (compared to the
external 32K clock, this method is recommended)
#define RKPM_PMU_PMUALIVE_32K    BIT(5)
// Use external 32K crystal as system clock, not recommended
#define RKPM_PMU_EXT_32K         BIT(6)
// Not selected by default
#define RKPM_DDR_SREF_HARDWARE   BIT(7)
// Not selected by default
#define RKPM_DDR_EXIT_SRPD_IDLE  BIT(8)
// In case of RKPM_ARMOFF, turn off the clk clock of PDM
#define RKPM_PDM_CLK_OFF         BIT(9)
// When suspend, pwm-regulator sets the same voltage as the maskrom (otherwise, a
lower voltage will be used, which is only required for wide temperature chips
now)
#define RKPM_PWM_VOLTAGE_DEFAULT BIT(10)
```

The suspend mode currently supported by RK3308 can be divided into 2 categories:

- VAD products: It needs supporting VAD wake-up source during suspend mode, keeping clock of the related modules such as VAD / ACODEC / PDM and the 24M crystal and related PLL work normally. At present, trust will first detect whether the VAD related mode has been turned off during the kernel stage. If it is enabled, that means it is VAD products, so trust will switch to the low power mode that supports VAD wakeup.
- Non-VAD products: There is no module that needs to be maintained during suspend, and almost all modules and clocks can be turned off, it is an extremely low power mode. In this mode, the system clock can be switched to 32K or 24M.

2.2 Power Configuration

Configure Item:

```
rockchip,pwm-regulator-config = <...>;
```

Configure option:

```
// use pwm-regulator
#define RKPM_PWM_REGULATOR          BIT(2)
```

NOTE about Power:

- Please check if there is pwm-regulator on the board.

2.3 Wake-up Configuration

Configure Item:

```
rockchip,wakeup-config = <...>;
```

Configure option:

```
//  Supports all interrupt wakeups (without GIC management), not recommended
#define RKPM_ARM_PRE_WAKEUP_EN      BIT(11)
// Supports all interrupt wakeups (With GIC management, that allowed to be wakeup
source)
#define RKPM_ARM_GIC_WAKEUP_EN      BIT(12)
// SDMMC Wake-up
#define RKPM_SDMMC_WAKEUP_EN        BIT(13)
#define RKPM_SDMMC_GRF_IRQ_WAKEUP_EN    BIT(14)
// RK TIMER Wake-up
#define RKPM_TIMER_WAKEUP_EN        BIT(15)
// USB Wake-up
#define RKPM_USBDEV_WAKEUP_EN       BIT(16)
// PMU internal timer wake up (default 5s), generally used to test sleep wake up
#define RKPM_TIMEOUT_WAKEUP_EN      BIT(17)
// GPIO0 Wake-up
#define RKPM_GPIO0_WAKEUP_EN        BIT(18)
// VAD Wake-up
```

```
#define RKPM_VAD_WAKEUP_EN          BIT(19)
```

Points to note for wakeup sources:

- RKPM_GPIO0_WAKEUP_EN (preferred):

  GPIO0 ~ 3 only supports the pin group GPIO0 as a wake-up source. In this mode, the pin interrupt signal on GPIO0 is directly sent to the PMU state machine without going through the GIC. In terms of hardware design, it is recommended that users put the required wake-up sources as much as possible on the pin group GPIO0.

- RKPM_ARM_GIC_WAKEUP_EN (second choice):

  All wake-up interrupts registered to GIC with `enable_irq_wake ()` in the kernel are supported. The number of applicable wake-up interrupt sources is more than `RKPM_GPIO0_WAKEUP_EN`. However, this method is equivalent to pass the management of the wake-up to each module, and the system may be woken up by an unexpected interrupt during suspend.

- RKPM_TIMEOUT_WAKEUP_EN:

  The timer inside the PMU wakes up. Generating an timeout interrupt every 5s, which is generally only used for sleep wakeup test.

2.4 Debug Configuration

Configure Item:

```
rockchip,sleep-mode-config = <...>;
```

Configure option:

```
// Ignored
#define RKPM_DBG_INT_TIMER_TEST     BIT(22)
#define RKPM_DBG_WOARKAROUND        BIT(23)
#define RKPM_DBG_VAD_INT_OFF        BIT(24)
// Always keep all clks enabled during sleep
#define RKPM_DBG_CLK_UNGATE         BIT(25)
// Ignored
#define RKPM_DBG_CLKOUT             BIT(26)
// PMU state machine signal output
#define RKPM_DBG_FSM_SOUT           BIT(27)
// Ignored
#define RKPM_DBG_FSM_STATE          BIT(28)
// Dump some register: gpio/grf/sgrf...
#define RKPM_DBG_REG                BIT(29)
// Ignored
#define RKPM_DBG_VERBOSE            BIT(30)
#define RKPM_CONFIG_WAKEUP_END      BIT(31)
```

Debug Note：

- RKPM_DBG_CLK_UNGATE：If it is suspected that some clk was disabled during the suspend, and it causes the system wake up error, this configuration can be enabled.

- **RKPM_DBG_REG:** If it is suspected that some register value is changed by trust during suspend, this configuration can be enabled.
- **RKPM_DBG_FSM_SOUT:** After this configuration is enabled, the PMU state machine will always output a specific waveform signal through GPIO4_D5 during suspend to feedback the current internal state of the PMU state machine. This function is only helpful when the PMU state machine crashes during system suspend.

2.5 Reboot /Reset Configuration

Configure Item:

```
rockchip,apios-suspend = <...>;
```

Configure option:

```
#define GLB1RST_IGNORE_PWM0      BIT(23)
#define GLB1RST_IGNORE_PWM1      BIT(24)
#define GLB1RST_IGNORE_PWM2      BIT(25)
#define GLB1RST_IGNORE_GPIO0     BIT(26)
#define GLB1RST_IGNORE_GPIO1     BIT(27)
#define GLB1RST_IGNORE_GPIO2     BIT(28)
#define GLB1RST_IGNORE_GPIO3     BIT(29)
```

Reboot reset notes:

At present, RK3308 uses first global software reset by default, which all module will be reset when rebooting. If you need to keep certain module from being reset, please configure the above options. Currently supported: `pwm0 ~ 3 / gpio0 ~ 3` can be not reset.

Examples of requirements for GPIO not reset:

Some hardware circuit designs will provide "power hold" power control pins, which need to be pulled high / low by the software in the early stage of system power to ensure that the system power supply works normally. The "power hold" pin cannot be reset during the reboot. Otherwise, the system will power off.

3. Printed Log

The following briefly introduces the meaning of the trust print information during system suspend and wake-up. For convenience of annotation, some print contents are branched as follows. Different suspend modes will bring different prints.

RK3308 System suspend printing:

```
// The pin that is able to wakeup system
GPIO0_INTEN: 00000041
GPIO4_INTEN: 00001000
// Configuration from kernel:
```

```
v1.3(release):005c64b, cnt=1, config=0x8040009:armoff-hwplldown-ddrsw-gating-24M-
sout-
// Sleep process: each character represents a sleep step in trust
0123a4
// The enabled state of the related modules of VAD, 1: enabled, 0: disabled
Enabling: vad(1) acodec(1) pdm(0) i2s_2(1)
// PLL status occupied by each module, "Enabling" indicates the PLL that the
system is using (because not all PLLs will be on)
DDR: vpll0 | VOICE(sum): vpll0 | I2S: vpll0 | PWM: dpll | Enabling: apll dpll
vpll0 | CRU_MODE: 3955
// Disabling" means PLL that will be turned off during suspend
PMU Disabling: apll dpll vpll1
// PMU register value(not important), and "24Mhz" means the current system clock,
if the current is 32K, the printing will also change dynamically.
PMU: pd-000e wake-000c core-0bfb lo-180d hi-000e if-4001 24Mhz
// Printing sleep process step. Note: After printing "wfi", it means that the
system suspend is done!
5bRc678wfi
```

RK3308 System wakeup printing:

```
// print wake-up process
876ab543210
// wake-up resource
IRQ=89
PMU wakeup int: vad
VAD int=00000113
// The real suspend sleep time of this round
Wfi total: 2.419s(this: 2.419s)
```