

物联网节点 SoC 的功耗管理器设计

黄泽林^{1,2}, 乔树山^{1,2}, 袁 甲¹

(1 中国科学院 微电子研究所, 北京 100029; 2 中国科学院大学, 北京 100049)

摘 要: 物联网对传感节点 SoC 的功耗要求很高, 很多应用场景不需要芯片所有模块都一直处于工作状态, 甚至大部分时间内只有极个别模块需要工作. 基于这个特点, 本文提出了一种功耗管理器的设计, 通过时钟管理和电源管理实现了多工作模式, 以最精简的逻辑应对多种工作场景. 功耗管理器支持工作模式、睡眠模式和深度睡眠模式, 用 SMIC 130 nm 综合, 1.08 V 电压下工作频率为 32 kHz 时, 电流仅为 0.47 μ A.

关键词: 低功耗; 时钟门控; 电源门控; 电压域划分

中图分类号: TN4

文献标识码: A

文章编号: 1000—7180(2017)10—0001—04

Power Management Design for Low Power SoC

HUANG Ze-lin^{1, 2}, QIAO Shu-shan^{1, 2}, YUAN Jia¹

(1 Institute of microelectronics, Chinese Academy of Science, Beijing 100029, China;

2 University of Chinese Academy, Beijing 100049, China)

Abstract: The requirement of IoT sensor node SoC is strict. In most application scenario, there is no need that all the modules keep working all the time. Only a few modules need to work instead. We propose the idea of power management, realizing multi-work mode with clock management and power management, as well as using the least number of modules to work. Power management is able to support work mode, sleep mode and deep-sleep mode, the current is only 0.47 μ A at the voltage of 1.08 V and the frequency of 32 kHz, fabricated with SMIC 130 nm.

Key words: low power; clock gating; power gating; power domain division

1 引言

物联网的发展对 SoC 的功耗提出越来越高的要求, SoC 的低功耗设计是研究的热点^[1-3]. 传感节点芯片内包含基带处理器、信号处理和射频通信等高性能模块, 功耗很大. 在工作期间传感节点保持驱动若干个传感器、采集传感器数据、处理和分析数据、驱动无线数字基带、组网、驱动射频发送信息. 完成这些工作都需要消耗大量的电流, 而目前的电池技术电量有限, 难以使物联网传感节点保持长时间的工作. 物联网传感节点需要在不更换电池的前提下连续工作 1 年甚至若干年^[4], 因此物联网节点 SoC 的低功耗设计至关重要.

物联网传感节点并非所有模块都需要保持一直

工作. 在大部分时间内, 芯片内只有传感器及其驱动外设保持工作, 而 CPU 和一些高性能的通信模块处于空闲状态. 偶尔 CPU 会需要做一些搬运数据和信号处理的工作, 接着又进入空闲等待状态. 到了采集到异常数据、或者采集的数据累计到一定量时, 通信模块才从空闲状态进入工作状态, 将信息发送出去. 因此, 物联网传感节点中只有一小部分模块需要大部分时间内都保持工作^[5], 如果能将其他大部分处于空闲状态的模块关闭时钟或者关闭电源, 那么芯片的动态功耗和静态功耗就会显著地降低.

针对物联网 SoC 的这个特点, 本文提出一种功耗管理器的设计方法及其相适应的电源域划分方法, 根据芯片各模块的特点进行分类并归纳成不同的电源域, 利用功耗管理器控制各个电源域的时钟

门控和电源门控,使空闲的模块进入睡眠模式或者深度睡眠模式,从而降低芯片的动态功耗及静态功耗^[6],实现了灵活的、简单的、低功耗的功耗管理策略。

2 功耗管理器综述

功耗管理器是整个芯片的时钟和电源的控制中心,控制着芯片内各个模块的时钟门控和电源门控。在功耗管理器的控制下,一般的外设模块能进入 3 种模式:工作模式、睡眠模式和深度睡眠模式。CPU 除了这 3 种模式,另外能进入备份模式。

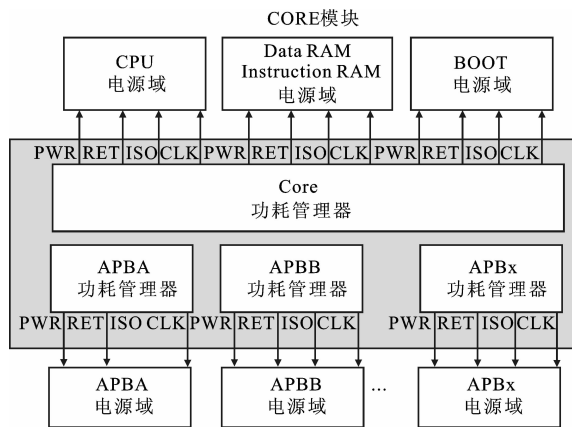


图 1 功耗管理器架构

如图 1 的阴影部分是功耗管理器的架构,里面包含了各个电源域的控制子模块,分别有 1 个 CORE 功耗管理器和若干个 APB 功耗管理器。APB 功耗管理器的数量等于芯片内 APB 电源域的数量。每个 APB 电源域都有一套时钟电源门控控制信号,包括 POWER、RETENTION、ISOLATION 和时钟门控 CLK^[7]。每个 APB 功耗管理器负责自己的电源域,而 CORE 功耗管理器负责 3 个电源域,分别是 CPU 电源域、SRAM 电源域和 BOOT 电源域,每个电源域分别有自己的时钟门控控制信号和电源门控控制信号。通过控制这些时钟门控和电源门控,可以让各个电源域进入不同的模式。

功耗管理器的设计存在两个关键问题。一,多电源域的划分颗粒度存在着折中,当划分颗粒度很细时,芯片内的电源域数量很多,芯片的时钟和电源控制就可以很灵活,但是对芯片后端的流程形成压力,芯片电源环数量太多也会占据一部分面积。相反,如果电源域颗粒度很大,则功耗管理的缺少了灵活性。二,设计控制时序,控制 CORE 功耗管理器的 3 套时钟电源控制信号和 APB 管理器的每一套时钟电源控制信号,让各个模块进入不同的模式。

3 电源域的划分

3.1 电源域划分方法

为了兼顾功耗管理的灵活度和芯片实现的复杂度,本文根据物联网传感节点 SoC 各外设的特点,提出如图 2 所示的电源域划分方法。

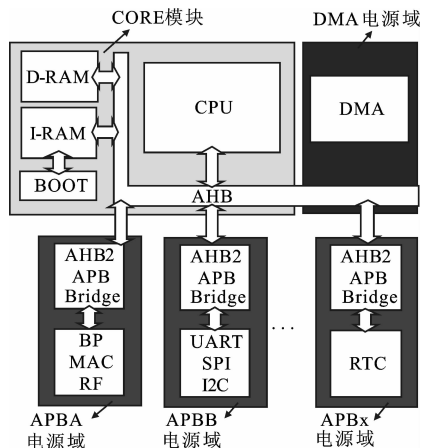


图 2 物联网传感节点 SoC 的电源域划分方法

CORE 模块:包含 CPU 电源域、SRAM 电源域和 BOOT 电源域。这 3 个电源域功能密切,需要统一地受同一个功耗管理器控制。

DMA 电源域:DMA 的功能相对比较独立,可以单独作为一个域。

APBA 电源域:这个电源域包含的都是传感节点中高性能高功耗的模块,如基带处理器 BP、以太网接口 MAC 和射频模块。这些模块面积大、MOS 管数量很多、工作频率相对较高,但是工作时间很短,往往是空闲了一段很长的时间之后,这些通信模块才会工作一段很短的时间。这是物联网 SoC 跟其他通信 SoC 的重要区别。这些高性能模块在同一个电源域里,占用芯片相当一部分面积。由于大部分时间内都处于空闲状态,若保持电源的关闭,则可以节省芯片内相当比例的静态功耗。

APBB 电源域:这个电源域包含的是传统普通 SoC 都有的驱动接口,如 I2C、SPI 和 UART 等,用来驱动外部传感器和各种芯片外部的模块。这些模块面积不大,工作频率不需要特别高,但是工作时间相对较长,在芯片工作的大部分时间内需要负责驱动外部传感器采集数据。

其他更低功耗的电源域:如图中的 APBx 电源域,包含了 RTC 这种工作频率特别低而工作时间特别长的模块。

常开域:芯片内的功耗管理器是唯一的不会被

关断电源和时钟的电源域,也称为常开域。

3.2 划分方法的优点

按模块的属性划分电源域,可以实现芯片功耗管理的“数量级”控制. APBA 都是高性能高频率模块,这些模块占了芯片相当部分的动态功耗及静态功耗,关闭之后芯片的功耗就下降了很大一部分. APBB 的模块相对较低,但在 APBA 关闭之后, APBB 模块就是芯片功耗的主要贡献者,可能比其他 APB 的总和还要多. 芯片在关闭 APBB 时就进入了更低的功耗状态. 如此类推,这样划分电源域,在高功耗高性能场景的功耗控制范围很大,在低功耗低速场景的功耗控制范围很精细.

另外,电源域间的功能相对独立,相互之间的跨时钟域传输由 AHB 转 APB 桥实现,每个 APB 电源域可以采用不同的工作频率,每个电源域的时钟和电源独立地开关而不影响其他电源域的工作。

4 电源域时钟电源的管理

4.1 APB 功耗管理器设计

物联网传感节点的各个电源域并不一直在工作,大部分时候大部分模块都处于空闲状态.当电源域所有模块都处于空闲状态时,可以断开电源域时钟或者断开电源,以降低整个电源域的功耗,延长芯片的电池寿命.

电源域的时钟和电源控制可以通过软件写入配置寄存器来实现,功耗管理器的状态机根据配置寄存器的数值跳转,跳转经过的不同状态分别控制着电源信号 ISOLATION、RETENTION 和 POWER,以实现时钟或者电源的门控.

图3中的状态机状态比较多,但最终稳定的状态是 S_0 的Deep Sleep、 S_5 的Sleeping和 S_6 的Working,状态机最终驻留在哪个状态取决于配置寄存器Sleep Mode的数值.

通过图 3 的状态机, 传感节点 SoC 可以通过软件控制芯片内各个电源域的时钟和电源开关情况, 根据当前应用情景通过软件来降低芯片的动态功耗和静态功耗.

4.2 CORE 功耗管理器设计

物联网传感节点的 CPU 在大部分时间都处于空闲状态,只有在少数时间需要采集数据、信号处理和信息发送。CPU 是传感节点 SoC 内一个非常大的模块,在 CPU 空闲不工作的时候关闭其时钟或者电源,可以显著节省芯片的功耗^[8]。

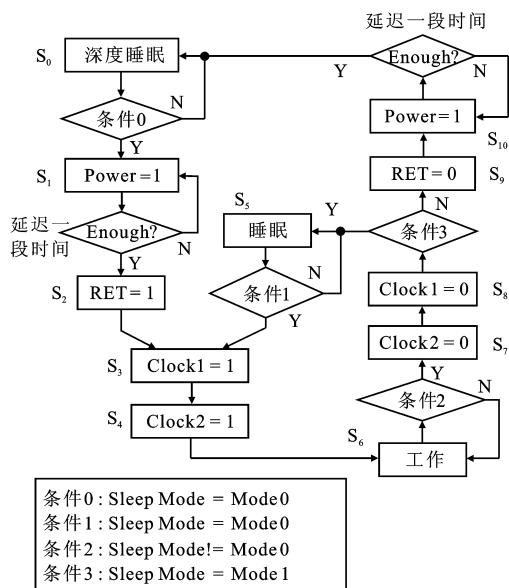


图 3 APB 功耗管理器的状态机

CPU 的空闲和 SRAM、BOOT 密切相关. 当 CPU 空闲的时候, SRAM 也不被访问, SRAM 也可以进入睡眠. 当 CPU 和 SRAM 都从掉电状态进入工作状态, 需要 BOOT 模块重新加载代码到 I-RAM, 加载数据到 D-RAM. 因此可以设计 4 种 CORE 电压域的工作模式:

Mode 0 工作模式: CPU 和 SRAM 的时钟和电源都被打开, 正常工作.

Mode 1 睡眠模式: CPU 和 SRAM 的时钟被关闭, 电源保持供电.

Mode 2 深度睡眠模式: CPU 的时钟和电源被关闭, SRAM 的时钟被关闭, 电源保持供电.

Mode 3 备份模式: CPU 和 SRAM 的时钟都被关闭. 由于 SRAM 的数据断电丢失, 在睡眠之前, 需要将 SRAM 的数据保存到 FLASH 中, 等待到被唤醒的时候将数据从 FLASH 重新放到 SRAM 中.

这 4 种模式的相互转换可以用如图 4 所示的状态机实现. 这个状态机有 8 个状态, 每个状态都对应着一系列电源信号 ISOLATION、RETENTION 和 POWER 的操作. 不同模式下的状态转换最终的稳定状态都一样, 分别是 S0 睡眠和 S4 工作, 但经过的状态不一样. 以下是各模式在睡眠和唤醒时所经过的状态:

Mode 1: $S_0 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_0$

$$\text{Mode 2: } S_0 \rightarrow S_1 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_7 \rightarrow S_0$$
$$\text{Mode 3: } S_0 \rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_6 \rightarrow S_7 \rightarrow S_0$$

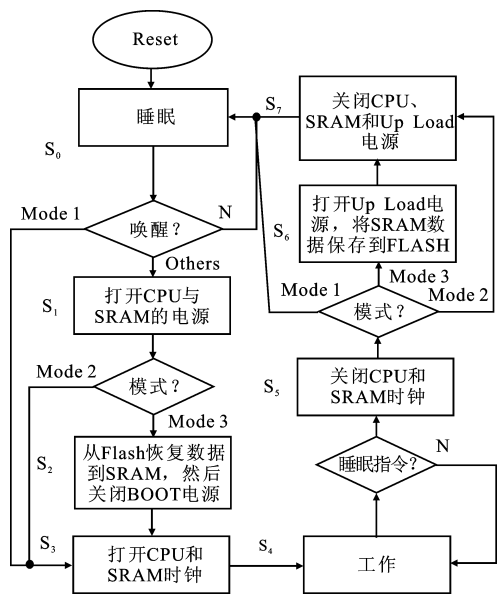


图 4 CORE 功耗管理的状态机

越深的睡眠,走过的状态越多,睡眠和唤醒的时间越长,但睡眠的静态功耗越低. 物联网传感节点 SoC 可以根据应用场景,在唤醒时间和睡眠功耗之间折中,选择最适合的睡眠模式,尽可能降低 SoC 的功耗.

5 实现的功能及功耗

5.1 最终功能

如图 5 所示,功耗管理器一共控制着 3 个 APB 电源域和一个 CORE 电源域,各电源域模块都处于空闲状态时,软件可以通过配置功耗管理器的配置寄存器将空闲电源域电源断开,从而降低了功耗.除了关闭空闲模块 CPU 和 APBA 通信模块电源,可以尽量使用低功耗的外设 APBC 去采集传感器的数据,让芯片电池的续航能力更长.

5.2 功耗代价

物联网传感节点 SoC 引入功耗管理器,能根据芯片内各个模块的空闲状态,在唤醒的时间和节省的功耗之间作出折中选择最适合的睡眠深度让各个电源域进入睡眠,从而显著节省了传感节点的功耗,但需要付出的代价是功耗管理器本身在整个芯片工作期间都要消耗一定的动态功耗和静态功耗.因此功耗管理器本身的功耗在物联网 SoC 的低功耗设计中至关重要.

本文的功耗管理器一共有 1 个 CORE 管理器、4 个 APB 管理器,整个功耗管理器在 SMIC 130 nm 工艺下综合,在电压 1.08 V、频率 32 kHz 时的功耗如表 1 所示.

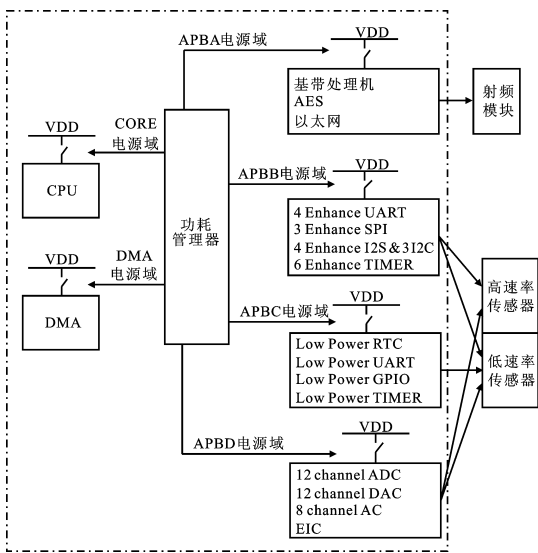


图 5 功耗管理器的一种应用

表 1 频率为 32 kHz 各子模块功耗 μW

子模块	CORE 功耗管理器	单个 APB 功耗管理器	总功耗
动态功耗	0.064	0.008	0.096
静态功耗	0.677	0.321	1.961
总功耗	0.742	0.329	2.058

整个功耗管理器(1 个 CORE 功耗管理器和 4 个 APB 管理器)的总功耗为 $2\mu\text{W}$ 左右,电流小于 $2\mu\text{A}$. 综上分析,该功耗管理器能满足低功耗的要求.

6 结束语

本文提出的功耗管理器,支持 4 种不同的睡眠模式,使 SoC 在不同场景中用最精简的逻辑工作,大大降低 SoC 的功耗.功耗管理器本身的工作电流仅为 $0.47\mu\text{A}$,仅引入极小的代价.

参考文献:

[1] 王祚栋,魏少军. SOC 时代低功耗设计的研究与进展 [J]. 微电子学, 2005, 35(2):174-179.

[2] Conti F, Palossi D, Marongiu A, et al. Enabling the heterogeneous accelerator model on ultra-low power microcontroller platforms [C] // 2016 Design, Automation & Test in Europe Conference and Exhibition. Dresden, Germany, 2016:1201-1206.

[3] Piguet C, Renaudin M, Omnes J T, et al. Special session on low-power systems on chips (SoCs) [C] // Design, Automation and Test in Europe. Conference and Exhibition. Munich, Germany, 2001: 488-494.

(下转第 10 页)

局搜索和更快的收敛速度.

5 结束语

本文针对 IaaS 层虚拟机放置问题,给出了基于 SLA 的 QoS 相应指标,并在此基础上提出了基于 QoS 多维度改进的差分进化算法为资源放置算法.算法以差分进化算法为原型,结合 NSGA 的非支配排序思想,通过对非支配排序的改进,优化收敛速度和全局搜索,实现多维约束资源的放置.实验通过将 QoS 指标作为算法的相应约束,验证了本文算法在资源放置过程中可以提高虚拟机的放置效率与放置效果.

参考文献:

[1] 吴海双,张亮,李杰辉,等. IaaS 环境中一种保证 SLA 的资源调度策略[J]. 计算机工程,2013,38(7):51-54.
[2] Buyya R, Beloglazov A, Abawaiy J H. Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges[J]. Eprint Arxiv, 2010, 12(4):6-17.
[3] Verma A, Ahuja P, Neogi A. pMapper: Power and migration cost aware application placement in virtual-

ized systems[C]//International Conference on Middleware. New York, NY, USA,2008.
[4] 王新杰,雷印杰,严华,等. 云计算中面向能耗降低的虚拟机多资源放置算法[J]. 计算机测量与控制,2015,22(12): 4133-4137. DOI:10.1109/ICACCS.2013.6938749.
[5] 李强,郝沁汾,肖利民,等. 云计算中虚拟机放置的自适应管理与多目标优化[J]. 计算机学报, 2011, 34 (12):2253-2256.
[6] 董健康,王洪波,李阳阳,等. IaaS 环境下改进能源效率和网络性能的虚拟机放置方法[J]. 通信学报,2014,34 (1):72-80.
[7] Rodrigo N Calheiros, Rajiv Ranjan. Cloudsim:a novel framework for modeling and simulation of cloud computing infrastructures and services [R]. Australia, The University of Melbourne, 2010.

作者简介:

吴运涛 男,(1991-),硕士. 研究方向为移动云计算. E-mail:sr71tao@sina.com.
魏恒义 男,(1958-),高级工程师,硕士生导师. 研究方向为网络及分布式计算.
骆登欣 男,(1994-),硕士. 研究方向为移动云计算.
朱正东 男,(1963-),高级工程师. 研究方向为高性能计算.

(上接第 4 页)

[4] Moorandi D, Sicari S, Pellegrini F D, et al. Internet of things: vision, applications and research challenges [J]. Ad Hoc Networks, 2012, 10(9): 1497-1516.
[5] Gubbi I, Buyya R, Marusic S, et al. Internet of Things (IoT): a vision, architectural elements, and future directions [J]. Future Generation Computer Systems, 2013, 29(9):1645-1660.
[6] Piguet C, Renaudin M, Omnes J T, et al. Special session on low-power systems on chips (SoCs) [C] // Design, Automation and Test in Europe. Conference and Exhibition. Germany, Munich, 2001: 488-494.
[7] Keating M, Flynn D, Aitken R, et al. Low power methodology manual for system-on-chip design [M]. New York, USA: Springer, 2007: 33-40.
[8] Singhal V K, Menezes V, Chakravarthy S, et al. A

10.5 uA/MHz at 16 MHz single-cycle non-volatile memory access microcontroller with full state retention at 108nm in a 90nm process [C] // International of Solid-State Circuit Conference. San Francisco, CA, USA, 2015:1-3.

作者简介:

黄泽林 男,(1991-),硕士研究生. 研究方向为低功耗 SoC 设计.
乔树山(通讯作者) 男,(1981-),博士,副研究员. 研究方向为低功耗数字集成电路、电力线通信等. E-mail:qiaoshushan@ime.ac.cn.
袁 甲 男,(1985-),博士. 研究方向为极低功耗数字集成电路的设计方法.