

摘 要

随着 IC 设计的规模更大，速度更快，以及便携式设备的广泛需求，设计中功耗的问题越来越凸现出来，所以在整个设计流程中就需要对功耗进行分析和低功耗设计，这些技术可以保证芯片的每一部分都能高效、可靠、正确地工作。

选择合适的低功耗手段，必须以细致的功耗预估为前提，并且也要掌握工具的适用范围和能达到的低功耗底限。在流程中尽可能早的分析出功耗需求，可以避免和功耗相关的设计失败。通过早期的分析，可以使用高层次的技巧来降低大量的功耗，更容易达到功耗的要求。

本论文围绕数字 CMOS 电路的功耗问题进行展开，主要分成两大部分。

首先针对超大规模集成电路中的功耗分析进行探讨，介绍了在 RTL 级、门级不同层次上对功耗进行分析的方法和对实际设计的指导意义，并对一个 450 万门的超大规模芯片在各层次上进行功耗分析，并和流片后测试得到的结果有着很好的吻合。

然后是对低功耗进行了一些结构上的设计。动态电压缩放（DVS）技术是一种通过将不同电路模块的工作电压调低到恰好满足系统最低要求来实时降低系统中不同电路模块功耗的方法，有着良好的应用前景。本论文实现了一款动态电压缩放（DVS）电路，可应用于突发吞吐量工作模式的处理器，通过和一个电路实例的整体仿真，验证了该 DVS 电路的低功耗效果。

关键字：低功耗；功耗分析；动态电压缩放

Abstract

Liu Hainan (Microelectronics and Solid-State Electronics)

Directed by Professor Zhou Yumei

As the design of IC go into larger and faster, the issue about power consumption is more critical. It is necessary to analysis the power accurately and manage low power techniques in every step of the design flow, so as to assure the efficient, reliable and correct function.

Choosing the appropriate low power solutions depends on careful power analysis as well as understanding the capabilities of available tools. Analyzing power requirements as early as possible in the design flow helps avoid power related disasters. Early analysis also makes power goals easier to attain because higher-level techniques save the greatest amount of power.

The thesis is made up of two main parts based on the discussion of the digital CMOS power consumption.

First of all, this thesis introduces and demonstrates a top-down VLSI design methodology for power analysis, discuss the method to estimate the power on RTL and gate level, which could serve as a guide to the floorplan and place & route. And estimate the power consumption about a 4.5 million VLSI on several level, draw some conclusion from comparing the test result of the fabricated chip.

In the second, completed a low power technique on the structure level. Dynamic Voltage Scaling is a technique using the lowest level voltage in real time on different block dramatically reducing energy consumption, while maintaining the desired level of performance, which has a nice prospect to realize low power. The thesis has developed a DVS circuit, which could get the corresponding lowest voltage according to the system frequency. Take a 16X16 multiplier as a test circuit to simulate together, proving the low power action of DVS.

Keyword: low power, power analysis, Dynamic Voltage Scaling

目 录

摘 要	I
目 录	III
第一章 绪论	1
1.1 前言	1
1.2 低功耗设计研究的现状	2
1.3 论文的内容与安排	3
第二章 低功耗设计方法	5
2.1 功耗模型	5
2.2 低功耗设计方法	6
2.3 各个层次上的功耗预估	13
2.4 450 万门超大规模芯片的功耗预估	20
第三章 动态电压缩放电路	24
3.1 DVS 概述	24
3.2 DVS 的适用范围	28
3.3 DVS 的应用	31
3.4 DVS 的性能指标	32
3.5 动态 DC-DC 转换器的设计考虑	34
第四章 动态电压缩放控制电路的实现	41
4.1 DVS 原理框图	41
4.2 电路的实现	43
4.3 电路的仿真与低功耗验证	53
第五章 总结	57
参考文献	58
发表文章目录	60
致 谢	61

第一章 绪论

一、前言

随着微电子技术的迅猛发展，最突出的表现是特征尺寸的不断缩小，集成度遵从摩尔定律不断提高。1964 年，Intel 公司创始人之一 Golden Moore 便预测说集成在单个硅芯片上的晶体管数量每 18 个月将会翻一番，同时芯片成本也将相应下降，这就是著名的“摩尔定律”。

集成电路芯片的功能变得更加丰富，工作速度越来越快，器件尺寸越来越小，芯片的成本越来越低。当前系统芯片（SOC）成为微电子发展的必然趋势。基于 SOC 的芯片设计是将更大，更复杂的系统集成于单个芯片。单个芯片内可能集成通用微处理器核(MCU Core)、数字信号处理器核(DSP Core)、存储器核(Memory Core)、专用电路(ASIC)等^[1]。系统功能的多样性和复杂性一方面增加了芯片功耗，对低功耗设计和精确预估功耗提出了挑战；另一方面也提供了更多降低功耗的机会。与此同时，器件越做越小，工作频率越来越高，使芯片单位面积的计算负荷迅速上升，导致高性能芯片的功耗越来越大。尤其是在进入 90nm 后，芯片频率可能高达 3GHz，其巨大的发热现象极大地影响了芯片功能的实现，已经被业界称为发热壁垒。

功耗的迅速增加将会引起一系列的问题，比如芯片的散热措施也需要不断更新，从改变封装形式到添加散热装置，明显地增加了芯片的成本。最新的苹果 Mac5 处理器，已经开始采用水冷散热的措施。

在半导体工艺不断进步的同时，以电池为供电形式的手持设备和笔记本电脑等便携式电子产品迅速普及开来，系统的功耗已经成为这些系统设计首要考虑的因素。尽管电池技术一直在提高，但与半导体和通信产业的飞速发展相比，它的供电能力和重量一直是便携设备的瓶颈。在过去的 30 年中，电池的容量仅增加了 2—4 倍^[2]，不可能跟上集成电路发展的要求。低功耗设计对于确保合理的电池工作时间是具有决定意义的，所以说，各种便携式通信及其它消费电子产品的快速发展，是要求低功耗设计的一个直接推动因素。

另一个与功耗有关的问题是可靠性。功耗增加会引起芯片温度的升高，将引发一系列的故障机制，如硅片连线故障、封装故障、电学参数漂移、电迁移等等，都会导致器件可靠性下降。温度每升高 10°C ，器件的故障率就要提高 2 倍^[3]。

高速数字电路中的地线反弹（Ground Bounce）现象也要求对功耗进行细致的分析，尤其是对瞬态的功耗分析有更为迫切的要求。随着数字电路工作频率的不断提高，其输出跳变速率也不断上升，再加上 I/O 数目的增加，在接电源、地引脚上的电流变化率也会大大上升，而使芯片内部电源、地线的电压有一个较大的变化，会损害信号的完整性，这就需要功耗完整性（Power Integrity）方面的考虑。PI 是不能够避免的，但可以从各个方面着手降低这方面的影响，比如封装形式、容性负载，电路板级优化等。如果能够降低芯片的功耗，就会使电流的绝对数值减小，对降低地线反弹会有直接的改善。

基于以上原因低功耗设计越来越引起人们的关注，已经继速度、面积之后，和可测性一起成为现在 VLSI 设计中的又一个热门领域。

二、低功耗设计研究的现状

低功耗电路设计就是在电路设计过程中采用各种设计手段，降低电路的功耗，当然所采用手段不能明显降低电路的时序性能，电路应该是一个合理的速度、面积、功耗和可测性的折衷。

根据低功耗措施介入电路设计阶段的不同，低功耗设计方法可以分为多个层次，按照抽象层次的不同，可以分为：系统级、算法级、结构级、逻辑门级、电路级、版图级和工艺级。每个级别可以达到的低功耗设计效果也完全不同。抽象层次越高表明在数字系统的设计中进行低功耗考虑得越早，所采用的低功耗设计策略的效果就会越明显。

系统级低功耗设计主要考虑软硬件分工、动态电压管理等方面，它研究的重点是数字系统如何操作和控制各个芯片的工作，达到降低功耗的目的。

算法级的低功耗设计研究主要在算法的复杂性、规整性和所需的数据精度几个方面。

结构级可以使用包括并行化处理、流水线处理和分布处理在内的并发处理，以及门控时钟、操作数隔离和电源管理的方法降低功耗。

逻辑门级低功耗设计的主要手段有：工艺映射、改变晶体管尺寸、缓冲器插入、调整相位、管脚置换和因式化简等。

电路级主要利用动态逻辑、传输门逻辑、异步逻辑等电路结构来降低功耗。

在版图级阶段，设计者可以考虑互连线电容的降低和以功耗驱动的自动布局布线来降低功耗。

工艺级是可以进行低功耗考虑的最低层次，也就是在具体电路实现采用某些措施来降低电路的功耗，在这个层次主要可以考虑：逻辑类型的选择、优化工艺降低电容、电压缩小（**voltage scaling**）等方面。这里的电压缩小是指在具体电路实现时通过降低电路的供电电压来达到降低功耗的目的，是系统级的动态电压管理的具体实现。

抽象层次的合并与细化会产生其他不同的分类方法，但这些技术的基本思想都是为了降低电源电压、电压摆幅、电容、开关活动率中的一项或几项。

国外很多大学和公司都开展了数字电路的低功耗研究，在各个层次上都进行了很深入的研究，并取得了很显著的成果。我国在集成电路的低功耗研究的各个层次上也都有所开展，主要集中在各高校和研究所。

三、论文的内容与安排

本论文围绕数字 CMOS 电路的功耗问题进行展开，主要分成两大部分。首先针对超大规模集成电路中的功耗分析进行探讨，介绍了在 RTL 级、门级不同层次上对功耗进行分析的方法和对实际设计的指导意义，并对一个 450 万门的超大规模芯片在各层次上进行功耗分析，根据流片后测试得到的结果，得到一个相互对照的结论。然后是对低功耗进行了一些结构上的设计，实现了一款动态电压缩放（DVS）电路，适用于突发模式的处理器，并和一个电路实例一起进行仿真，验证了该 DVS 电路的低功耗效果。

研究重点及章节安排如下：

第二章首先对数字 CMOS 电路的功耗建模进行介绍，然后介绍了在不同层次上的低功耗技术，最后是对低功耗 ASIC 流程中的功耗预估进行着重讨论。第三章是针对动态电压缩放电路而展开的，介绍了动态电压缩放的概念、指标、适用范围和一些设计上的考虑，以及现阶段动态电压缩放的应用和开发状况。第四章是动态电压缩放控制电路的实现，包括各部分电路的实现、仿真结果低功耗验证与分析。第五章是对全文的总结。

第二章 低功耗设计方法

本章对数字 CMOS 电路进行了功耗建模，系统地介绍了在各个层次上的低功耗技术，并对低功耗 ASIC 流程中的功耗预估进行重点讨论。

第一节 功耗模型

CMOS 电路的功耗可以分为两部分：广义地被分成静态和动态两大类。

一、静态功耗

静态功耗是由于 MOS 晶体管的开关特性并不是理想的，存在漏电流、亚阈值电流、衬底注入电流等非理想的因素，在门不翻转时，即不活动或静态时所消耗的功耗。绝大部分静态功耗是由漏源亚阈值漏电产生，主要由于减小阈值电压而不能使晶体管完全截止。有源区和衬底间的漏电也能产生漏电流。因此，静态功耗也常称为漏功耗。

二、动态功耗

动态功耗是指消耗在电路动作时的功耗，动态功耗是由于 CMOS 电路的开关动作所引起的。当外加激励加在电路上，使节点上电压变化时，电路便处于活动状态。因为在输入端的电平变化可能不能导致在输出端的逻辑变化，所以在输出端即使不发生逻辑变化时也可以发生动态的功率消耗。动态功耗由两方面组成：开关转换功耗和内部功耗。

1. 开关转换功耗

开关转换功耗：对于驱动单元的开关转换功耗是指消耗在对单元的输出负载电容进行充放电的消耗，这些电容是由连线、器件的寄生电容、CMOS 的输入电容等构成，驱动单元的总负载电容是指驱动的输出端所有的节点和门电容的总和。因为充电和放电是单元输出端的逻辑转换的结果，所以当逻辑转换增加时，功耗也相应增加。因此，单元的开关转换功耗是总负载电容和逻辑转换率的函数。开关转换功耗构成了 CMOS 电路功耗的大部分组成。

$$P_{dyn} = CV_{dd}^2 \alpha f$$

其中 C 为负载电容, V_{dd} 是工作电压, f 为电路工作频率, α 是开关活动率。在所有的功耗中, 充放电功耗是最重要的部分, 至少占了整个功耗的 90% 以上^[3]。而相当多的低功耗设计技术正是围绕减少充放电功耗展开的。

2. 内部功耗

内部功耗: 内部功耗是指在开关转换过程中消耗在单元内部的功耗。在开关过程中, 通过对单元内部的电容充放电而消耗的功耗。内部功耗还包括对于 p 管和 n 管在瞬间短路时所消耗的功耗, 也称为短路功耗。

$$P_{sc} = \kappa W \tau f$$

其中 κ 是由工艺和电压决定的, W 是晶体管的宽度, τ 是输入信号上升/下降的时间, f 是开关频率。

对于转换时间较短的电路, 短路功耗很小, 但对于转换较慢的电路, 短路功耗对于一个门的消耗的 50% 还要多^[4]。短路功耗还受到晶体管的尺寸和负载电容的影响。

第二节 低功耗设计方法

人们一直以来主要关注的问题是如何提高芯片的工作速度、节省硅片面积与成本以及提高系统工作的可靠性, 而对电路功耗的考虑通常处于相对次要的位置。但是由于 VLSI 本身的发展及市场需求使情况发生了一些根本的变化, 在很多领域, 降低功耗已成为数字系统设计的一个最为重要的问题。

低功耗设计贯穿于从系统级、算法(行为)级、结构级、逻辑电路级直到器件/工艺级的整个数字系统设计流程。在低功耗设计中, 首先要明确一个系统中的功耗分布, 在此基础上针对功耗消耗大的模块单元、关键路径和非关键路径进行功耗优化。由系统级到逻辑电路层次其功耗优化由系统设计者决定, 器件/工艺层次由半导体工艺决定。

从系统和算法级提出的低功耗设计方法的效率要比门级和寄存器传输级的高得多，门级和寄存器传输级得低功耗设计方法基本都是在电路的基本结构确定的前提下，添加少量的电路和微调电路的结构。算法级和系统级低功耗设计方法主要是从电路的体系结构和编码等方面入手，对具体电路实现没有特殊要求，这些低功耗措施对综合工具和布局布线工具完全是透明的，因此最后得到的电路质量较高。而结构级的低功耗设计方法是在确定电路实现方案时就要考虑电路的功耗问题，在电路的面积、速度、功耗和可测性方面做出权衡，因此设计出的电路在面积、速度和可测性方面会有所牺牲，但电路的功耗往往会有一个数量级的下降。

一、系统层次功耗优化

系统级功耗控制的基本方法是针对系统的不同工作模式和状态，设计出相应的低功耗工作模式。一方面系统应能按照性能设计要求完成系统功能；另一方面要求系统消耗尽可能低的功耗，即尽可能地工作于低功耗工作模式。由于系统所执行任务的多样性，对系统，特别是对处理器核的工作性能有不同的需要。以工作频率为性能标志，系统功耗设计应根据不同的任务需要让处理器工作于功耗尽可能低的状态，即较低的工作频率，将任务调度和电压缩放结合起来降低功耗

1. 软硬件划分

软硬件划分是从系统功能的抽象描述（如 C 语言）着手，把系统功能分解为硬件和软件来实现，对于一个系统的功能，可通过在处理器上运行软件来实现和通过专用电路实现，比较两者的功耗得出一个低功耗的实现方案，软硬件划分的技术处于设计的起始阶段，给降低功耗带来更大的可能。在根据系统功能说明进行软硬件协同设计、确定指令体系结构时，不同的设计出发点所导致的设计功耗结果差别会很大。因此系统硬件的各个子模块划分以及软件上设置不同的工作状态对功耗的优化非常重要。

2. 存储器优化

存储器是集成系统的重要组成部分，存储器的功耗包括对存储器本身功耗和处理器与存储器之间通信的功耗。进行低功耗设计时应合理设置存储器层次，对片上存储器和片外存储器进行合理规划，同时考虑到存储元件的类型 (SRAMs, DRAMs, cache)，及对它们的大小、组织结构进行合适的优化。

3. 动态功耗管理

动态功耗管理主要有两种技术，一种是电压缩放，即根据任务所需要达到的性能来选择处理器的电压或频率；另一种是进入低功耗的休眠状态，如果没有任务执行则关闭处理器来降低功耗。该技术的核心就是如何根据系统的状态信息决定系统何时对电压进行缩放，或何时进入休眠状态^{[5][6]}。

二、算法/行为层次功耗优化

对所需算法的复杂性、并发性进行分析，尽可能利用算法的规整性和可重用性达到降低功耗的目的。由于实现一个具体问题的方法往往具有多样性，而且它与具体实现的依赖性很小，所以算法的选择相对来说自由得多，而且它对最终硬件实现的功耗的影响也会很显著。

1. 算法的复杂性

每条指令的执行都需要一定的能量，对于确定的处理器，其每条指令的功耗是一定的，所以最简单的衡量一个算法是否在功耗方面为最优的方法，就是根据运行此种算法所需的指令数来判断。考虑到各条指令所需的功耗并不相同，更为精确的方法应当是以各条指令所消耗的能量为权值的加权和。

2. 算法的规整性

一个规整的算法本身就很适合 ASIC 来实现，对于降低功耗来说也同样是有利的：在 ASIC 中，规整的算法使得用来描述状态的有限状态机变得简单，使其耗能减少；算法的规整性会使所需要的判断分支语句减少，判断分支语句会引起流水线中空泡增多，导致无效的能量消耗。

3. 算法的数据精度

实现相同的功能要求，不同的算法所需的数据精度是不同的。如果数据精度高，数据的宽度就会很大，必然使运算部件的规模增加，功耗也会增加。

4. 控制-数据流图低功耗变换

在高层次综合技术中，算法通常表达成控制-数据流图的形式，进行优化后映射到特定的硬件上。这类技术在过去只考虑系统性能和系统费用的优化，在此基础上，将一个高层的功耗模型提供功耗的因子加入到优化算法中，从而获得低功耗的实现方案。

三、结构层次功耗优化

从前面的讨论可知，降低电路的工作电压可以有效地降低电路的功耗，但同时会增加电路的延时，因此设计者在降低电路工作电压的同时，必须采取措施提高电路的工作速度。采用适当的结构，可以在维持电路速度不变的情况下，降低电源电压。

1、采用并行结构

并行是将一条数据通路的工作分解到两条通路上完成，这样每条数据通路的工作频率都为原来的一半，数据通路允许的延时增加了一倍，此时电路可以采用较低的工作电压。虽然增加的电路和由它引起的连线资源会导致电容的增加，并且输出端口增加的多路选通器也会导致部分功耗，但通常情况下这两部分增加的功耗不会导致电路实际功耗的增加。

2、采用流水线结构

采用流水线结构是降低功耗的又一途径。电路流水就是采用插入寄存器的办法降低组合路径的长度，提高电路速度，在此基础上降低电路的工作电压，可以极大地降低电路地功耗。

将电路流水化和并行化达到节省功耗的前提是可以选择电路的工作电压，如果电路工作电压固定，上面两种方法只能提高电路的工作速度，并相应地增加了电路的功耗。

3、总线的低功耗设计

数字系统中总线的基本特点就是负载较大、走线较长、数据传输密度高，通常都具有较大的电容，形成大量的功耗。

1) 降低电压幅度

减小总线上信号的电压变化幅度（通常小于 1V）对降低具有特大电容总线的功耗非常有效。它的额外代价是总线和功能模块之间的信号电平的变换电路。

该方法对于芯片工作电压较高（3.3V 和 5.0V）的总线效果显著，随着工艺的进步，工作电压也随之降低，该技术在未来的应用范围将很有限。

2) 总线分割

对总线进行分段控制，根据总线和功能模块连接的物理结构，在信号传输时，隔断总线的无关部分，从而减小总线的实际电容，以达到降低功耗的作用。该技术作用于版图布线的设计阶段，受到系统总线结构和版图布局等因素的制约。从系统级设计的角度，对 SOC 系统的总线结构进行分割设计日益成为趋势。

3) 总线编码

降低总线上的翻转率同样是节省总线功耗的一种方法，而在完成同样功能的前提下要降低总线上的有效翻转频率只有改变总线上传输数据的编码。常见的总线编码形式有二进制原码、one-hot 编码、格雷码、总线反转码（Bus-invert Coding）和二进制补码。该技术的额外代价是附加的编码和译码电路，但随着工艺的发展，布线的电容和功耗将大大超过门单元，因此该技术在未来将具有很好的应用前景。

4、门控时钟

在时序电路中，全部触发器在时钟信号的作用下同步地转换状态。在开关过程中，如某一触发器地次态与现态相同，该触发器就处于维持状态，这时时钟对触发器的触发就会造成无效功耗。另外，一个系统中的不同模块在某些时

段内会处于空闲状态，他们此时的开关动作和时钟触发对系统都是无意义的。门控时钟的设计思想就是在上述的维持和空闲状态时，关闭其时钟信号，从而有效减小时钟驱动的功耗。门控时钟可以作用于局部电路或一个模块，也可以是整个电路，当然作用的范围越大，减少的功耗也越就显著。在电路中加入门控时钟非常容易，设计者可以自己在语言中描述，也可以通过综合工具自动来加入。

5. 操作数隔离

操作数隔离就是通过降低组合模块输入的翻转次数，来降低组合模块的有效翻转率达到降低组合模块功耗的目的。

组合电路的内部工作状态完全是由其输入信号决定，即使输入信号的翻转会引起组合电路输出信号的改变时，组合电路内部会有很多器件参与工作，将会导致电路消耗功耗，因此降低组合电路模块输入信号的无效翻转可以降低者部分电路的功耗。

一些算术模块，如算术逻辑单元(ALU)，加法器和乘法器是经常执行多余计算的单元。这些单元的输入端是比较理想的进行操作数隔离的对象。

6. 异步电路结构

异步逻辑是完全不同于同步设计的一种设计方法，与同步电路相比较，异步电路设计在低功耗设计中有着很大的潜力。异步电路本质上是数据驱动的，靠握手信号连接，只在需要时才工作，减少了模块的无效翻转，降低了电路的翻转活动率。异步逻辑不采用全局时钟，不存在时钟偏斜问题，节省了时钟翻转消耗的功耗。它的速度是根据输入情况变化的，是各种输入的“平均情况”；而同步电路中全局时钟的周期必须大于最坏情况，故同步电路的速度是“最坏情况”。异步电路的速度“平均情况”要比同步电路的“最坏情况”快，故异步电路可在低电压下达到所要求的速度，降低了电路所需要的电压，大幅度降低了电路的功耗，异步电路虽然在功耗方面有它特殊的优势，但因为实现方面的困难，要出现商用化的异步处理器可能还需要相当长一段时间。另外一个原

因是大多数异步电路的指令集都是自定义的，用户使用起来不方便，不能普及，因此大多数的异步微处理器还不能代替市场上现有的微处理器。

7. 动态电压缩放 (Dynamic Voltage Scaling)

这是系统级动态功耗管理策略在结构级的实现。因为实际情况下并不要求处理器总是工作在高性能状态，当工作量不大时，处理器工作在一个降低了的频率下就可以达到性能上的要求，此时就可以降低电路的工作电压，导致电路功耗的下降。要实现动态电压缩放，需要电路可以快速根据要求来切换电路的工作电压。本论文所实现的低功耗设计就是应用的这种结构，将在三、四两章中详细地展开。

UC Berkeley 设计的一种基于 ARM8 的体系结构的处理器就具有 DVS 功能，它采用 0.6 μ m CMOS 工艺实现。该处理器具有两种工作方式，当它工作在 1.1V/8MHz 时，功耗为 1.8mW；工作在 3.3V/100MHz 时，功耗为 220mW。

四、逻辑电路层次功耗优化

1. 公因子提取

在逻辑综合中，公因子提取和共享是简化逻辑网络、减少实现电路成本的常用方法。对提取的不同公因子计算其功耗因子，选择功耗最小的来实现电路。

2. 工艺映射

工艺映射实现把逻辑表达式或布尔网络映射到目标库中的门单元的过程。在映射过程中，尽量将活动因子大的节点隐藏于门单元的内部，从而减少其电容负载，降低功耗。

3. 门尺寸优化

门尺寸优化的基本思想是对非关键路径的门缩小尺寸以减小面积和功耗。由于使用驱动能力小的单元，电平转换将变慢，引起短路电流增加，这是门尺寸优化的制约因素。

对关键路径的门尺寸优化则根据路径上不同结点的跳变频率，对相关的门单元分别增大和缩小，在保证时序约束的条件下，减少功耗。在通过改变单元尺寸大小来节省功耗时，应该同时考虑负载驱动和电路延时的因素，在满足时序的前提下，调整电路的尺寸，尽量减小每个门输入信号的偏斜和输出负载。

4. 管脚置换

一般情况下，对于库单元功能相同的引脚，在综合时是等价的。但实际上，不同引脚的电容、信号延迟等参数是不同的。管脚置换的基本思想就是根据输入信号实际翻转率的不同，将翻转率较高的输入信号连接到电容较小的管脚上，达到降低功耗的目的。这种方法需要知道信号实际翻转率和准确的单元库模型。

五、器件/工艺层次功耗优化

电路按比例缩小，不仅提高了系统的集成度，也带来了功耗上的好处。系统集成使得芯片间的通讯量降低，可使功耗减小。

工艺的进步使多层金属布线成为可能，增加可布置的金属层数，使用上层金属进行全局互连，可以减小互连电容，从而减小延迟和功耗。

采用多阈值、阈值可变器件，以满足不同工作状态的需要。

新型的低功耗器件和计算方案也在研究中，发展了各种“可逆”计算的结果。如，在电路速度要求不高的应用中，可用绝热计算(Adiabatic Computing)这样的新型低功耗器件结构，以时间换取功耗的减少。

还可以通过加工技术的提高减小芯片和封装的电容，也能改善功耗性能。如采用 SOI 技术。SOI 技术能减少寄生电容和体效应，从而减小功耗。

第三节 各个层次上的功耗预估

由于 SOC 的趋势，在面积和速度上的增长导致功耗问题已经越来越受人们的关注。已经不仅仅是为了延长电池的供电时间，散热和可靠性方面的问题比如电迁移和电压降都变得非常重要。

在主流深亚微米的设计中，功耗可能会限制功能的实现和性能的高效，大大的影响可制造性和产品成品率。大功耗会增加结温，温度的升高会增大晶体管的延迟，也会增加互连线上的电阻。如果不考虑功耗，一些仅仅针对提高性能的设计手段是往往是达不到预想效果的。产品性能的不达标会降低产品的收益。另外，如果功耗过高，则需要在系统级对温度进行监测和管理。总之，这些功耗的问题极大地提高了 SOC 系统的成本。在 SOC 的设计流程中，需要对功耗进行适当的考虑，以减小成本。

低功耗 ASIC 流程和传统 ASIC 流程的不同之处在于：传统的设计流程对功耗的处理是到最后才作考虑的，直到设计流程到门级或晶体管结构级才做整体的分析。如果在这时为得到更高的能量效率，希望做进一步的修改，就已经太迟了。因此若想将设计的焦点集中在实现最大限度低功耗的话，就要在设计的最开始，即最高层次描述时，像对性能进行分析那样，对功耗也要进行分析；每个设计层次的功耗分析，要保证该层次的功耗设计能满足设计的要求；来自高层的功耗信息也将配合和约束下一层次的设计。低功耗设计流程的初衷就是尽可能早地、尽可能准确地预估出功耗，并采取一系列措施以便得到尽可能多的能量节省。

一、功耗计算

整个芯片的功耗是动态功耗和静态功耗的总和。动态功耗是指消耗在开关逻辑时，在单元内部（内部功耗）和驱动连线负载（开关功耗）所消耗的功耗。

$$Dynamicpower = CV^2f$$

其中 C 为负载电容， V 是工作电压， f 逻辑转换数。

随着半导体结构的不断缩小，器件和线间电容也不断减小，这一点意味着更快的速度和更小的功耗。但考虑到设计规模和开关转换率的激增，芯片功耗的表征却是越来越大。

静态功耗（漏功耗）是指在晶体管没有开关活动时的功耗。

$$Static \cdot power = VI_{STAT}$$

虽然晶体管会有漏衬间的反向漏流，但漏功耗的绝大部分是由晶体管关断时亚阈值电流产生的。

当经常处于闲置状态时，对漏功耗进行精确建模便相当重要了。随着阈值电压的降低，亚阈值电流变得越来越显著起来。如图 2.1，Intel 公司的资料表明，当工艺水平在 $0.13\mu\text{m}$ ，以及以上时，漏功耗要占到芯片总功耗的 50%^[7]。漏功耗的增加会指数倍地增加芯片的失效概率。

降低功耗的主要手段是降低电源的供电电压 V_{dd} 。因为动态功耗和供电电压的平方关系，电压的降低可以很好地改善芯片的动态功耗。但是，又因为电路的开关延迟与负载电容和 V_{th}/V_{dd} 成正比，所以降低电源电压也会降低电路的速度。因此，为了维持电路快速驱动并翻转的能力， V_{th} 需要和 V_{dd} 同时降低。这样就会使漏电流增加，增大了漏功耗。所以，需要一个低功耗的流程来对性能要求和功耗进行折衷。

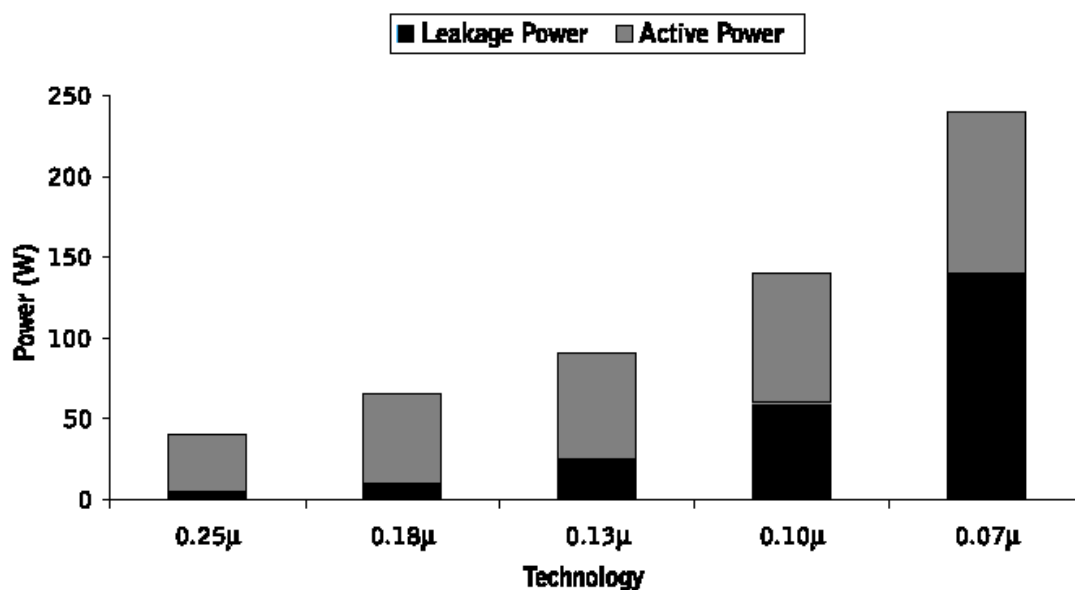


图 2.1 不同工艺下漏功耗比重示意图

二、功耗预估和分析

前面也提到了，在整个设计过程中，对功耗问题考虑地越早，对功耗最终效果的影响也就越大。例如，在系统和算法级，应用并行结构取代串行可以降低时钟频率，可以显著地降低功耗，不过，用并行手段是以大面积为代价的。当时钟频率降低后供电电压也可以适当降低，因为功耗与电压是平方律关系，与频率和翻转是线形关系，所以并行结构可以节省大量的功耗。

图 2.1 列出了可以应用于 SOC 流程中的一些功耗优化和功耗分析的技术。

System Design	Power Optimization Architecture optimization (e.g. parallel vs. serial) Supply voltage scaling Clock frequency scaling	Power Analysis Power estimates based on - Estimated gate counts - Estimated activity
RTL Design	Module clock gating	RTL power analysis based on defined clocks and registers - Estimated gate counts - Realistic activity
Floorplanning	Voltage islands	
Synthesis	Threshold voltage scaling Power optimization in synthesis RTL clock gating	Gate level power analysis based on - Actual gate counts - Realistic activity - Wireload models - Final libraries
Place and Route		Gate level power analysis based on - Actual gate counts - Realistic activity - Accurate routing - Final libraries

图 2.2 SOC 设计流程中功耗优化和分析技术

在流程的所有设计阶段，如表 2.1 的四个阶段进行功耗预估是十分必要的。随着获得库的信息，以及设计的逐步实现，功耗预估的结果也愈逼近最后的真实值。

表 2.1 功耗预估的四个阶段

进行预估的阶段	门数的计算方法	负载的计算方法	使用的预估工具
1.设计和库文件选择阶段	粗略估计	不确定	Spreadsheet
2.早期综合	粗略估计	DC 中线负载模型	Design Compiler Power Compiler
3.后期综合	精确的（摆放后）	线负载模型/SPEF	Power Compiler Physical Compiler PrimePower
4.版图实现后	准确的	提取的 SPEF	PrimePower

1. RTL 级功耗分析

在设计的最初阶段，功耗分析的目的只是需要提供一个相对粗略的预测值，为后续设计的可行性进行方向性的指导。这时，库可能还没有最后确定，只能通过线程表分析法，找出最优功耗相关的库和体系结构。在得到库信息后，或者以库中的相关信息代替线程表中的值，或者用 Design Compiler 和 Power compiler 代替线程表分析方法进行功耗分析。

1) 线程表分析法

功耗分析线程表包括门数的大约值，每个模块粗略的活动因子，一组制造商提供的 $\mu\text{W}/\text{MHz}$ 数据和相应的功耗预估值。通过在这个层面上的分析，可以明确那些因为功耗过大不可能实现的设计，避免了再用几个周去实现该电路的时间浪费。

要使用线程表分析法，需要首先估计每个模块的逻辑门数（每一种类型库单元）和活动率。还要有每种类型的单元在开关时的功耗，可以根据库提供商的手册对相关速度的功耗（ $\mu\text{W}/\text{Hz}$ ）赋一合适大小的数值。一个单元块的内部功耗由下面的关系给出：

$$\text{Power consumption} = \text{Gate Count} \cdot \mu\text{W}/\text{MHz} \cdot \text{Activity} \cdot \text{Frequency}$$

在一个模块中累加所有不同单元的内部功耗值，就可以得到模块的整体内部动态功耗预估。在综合前，门数的预估是基于体系结构的选择和对整个设计

的理解来进行经验性的推测。比如，可以根据总线大小，字长，控制的层次和存储器深度来对门数进行粗略地预估。得到库文件后，在早期综合后，使用 Design Compiler 中的 *report_refeference* 命令对模块的门数进行预估，可以得到设计中每种类型实例的数目。

2) 开关活动率

对功耗计算的关键方面就是活动率的确定。是否通过仿真来记录翻转情况，会导致所预估的门翻转率在精准程度上有很大不同。得到库信息后，推荐通过功能仿真来得到开关活动性。

开关活动性通过翻转率 (Toggle Rate) 来衡量。翻转率是指设计对象 (内部连线或端口) 上，单位时间内逻辑 0-1 和逻辑 1-0 的转换次数。例如，一个内部连线在 100ns 的时间间隔内有 50 次逻辑 1-0 转换和 50 次逻辑 0-1 转换，那么它的翻转率 (TR) 为 1；如果连线在 10ns 内有 5 次逻辑 1-0 转换和 5 次逻辑 0-1 转换，那么它的翻转率也为 1。若以纳秒为时间单位，翻转率 1 表示每一纳秒，有一次活动翻转。功耗和翻转率的线形关系可以直观地解释为，对于每一次翻转，在状态变换的时间内，必须从电源供应一定的能量来完成这一电路内部状态的变化。

在这里，必须明确一点：只有当开关活动性真正代表芯片内部的实际工作状况时，在各个抽象层次上的功耗预估才是有意义的。在实际工作中经常遇到的一个失误是，仅仅仿真完系统启动阶段的时序，就根据仿真情况确定了开关活动率的数值。这个阶段的数值，往往不能真正的代表实际的工作情况，这样会导致功耗预估的不准确。可用 RTL 仿真器自动产生开关活动性交换格式

(Switching Activity Interchange Format) SAIF 文件，只有当测试向量和实际状况相符时，才能保证得到的活动率为准确值。又因为当前的工具不能够自动产生测试激励，所以设计者应该对电路的功能有很好的理解和把握，需要自己编写出合乎正常工作状态的测试激励。

下面的几条命令是在 VCS 中通过仿真产生 SAIF 文件，用到的程序语言接口 (Programming Language Interface) PLI 系统任务：

```
$set_gate_level_monitoring("rtl_on");  
  
$set_toggle_region;  
  
$toggle_start;
```

```
$toggle_stop;
```

Power Compiler 使用 *power_estimate* 命令调用 SAIF 文件来确定库和约束，并反标到设计中，得到功耗预估。Power Compiler 对于没有反标的端口，在每一个上升沿，缺省的开关活动率是 0.25；并且在模块内部，该端口后面各节点是以这个缺省值为基准，经过传输后并通过相应计算得到其数值。

在没有准确的负载和开关数据的情况下，在得到了内部功耗后，开关功耗一般在经验上被预估为内部功耗的 30%，这个数据是比较粗略的预估结果。这种方法往往是用来评估使用不同设计策略后的功耗结果对比，而不是用来准确的确定芯片的实际功耗数值。就像前面提到的，在 RTL 级粗略的预估可以为因为设计存在不可接受的大功耗而在早期提出不可实施的预警。

在早期的分析中，动态功耗是最重要的指标，不过漏功耗也可以通过基于各种单元漏电流的数值进行预估。因为漏电流在高或低状态时的大小是不同的，漏电流的分析必须基于信号在某一确定逻辑状态下的状态概率来进行。状态概率是在 0 和 1 之间的一个数值，可以根据信号各自的功能进行预测。例如，一个低有效的复位信号处于 1 状态的概率（SP1）为 1 或者接近 1。对于数据总线信号，除非一些特殊的体系结构，SP1 一般被设置为 0.5。在得到库信息后，通过仿真，将信号处于一个逻辑状态的时间与总仿真时间相除，就可以得到状态概率。

2. 门级功耗分析

经过综合后，就可以通过 Power Compiler 基于实际的门数和仿真得到的开关活动率，计算出相对准确的功耗预估值。开关活动率和布局布线前所使用的线负载模型是导致在这个阶段上和实际情况有出入的主要因素。开关活动率可以通过在门级用 VCS 进行仿真得到 SAIF 文件而提高精确度。除了第一个命令应该修改外，和 RTL 级产生 SAIF 文件的命令类似，第一个命令如下：

```
$set_gate_level_monitoring("on");
```

再次强调一下，只有当测试激励代表实际的应用情况时，所得到的开关活动率才是较为准确的。

经过物理优化后,可以通过 Physical Compiler 用 *write_parasitics -distributed* 命令,产生 SPEF 文件,来反标 Steiner 布线和寄生 RC 参数的预估值,以增加负载的精确性,从而提高了功耗预估的准确性。

在版图实现后,通过门级仿真可以得到值变存储文件(Value Change Dump) VCD 文件,利用 PrimePower 进行更为准确的分析。在仿真过程中,VCD 文件记录了信号的变化和内部节点的动作、各层次的数据连接,路径的延迟,时序和事件信息等。

如果芯片的 I/O 数目过多,在进行高速开关转换以及驱动很长的连线时,I/O 也将是影响功耗准确度的重要因素。使用 I/O 的集中负载模型将会得到非常悲观的结果,如果设计的目标是要得到准确的功耗值,而不是最坏情况的功耗预估,就需要对 I/O 进行更为准确的分析。可以在 Hspice 中用准确的分布式阻抗模型,对关键的 I/O 单元类型进行仿真。可以计算出在每个上升/下降沿上所消耗的能量值,利用 Hspice 输出的电流和时间,以及梯形积分法(Matlab)可以得到每一个瞬间的 I/O 单元的内部能量消耗。再依据 PrimePower 分析过程中得到的 I/O 翻转率,就可以得到相对精确的功耗值,再加上芯片核心部分的功耗,就可以对整体功耗进行相当准确的预估。

选择合适的低功耗手段,必须以细致的功耗预估为前提,并且也要掌握工具的适用范围和所能达到的低功耗底限。在流程中应该尽可能早的分析出功耗需求,以避免和功耗相关的设计失败。通过早期的分析,可以使用高层次的技巧来降低大量的功耗,更容易达到功耗的要求。

第四节 450 万门超大规模芯片的功耗预估

下面所要讨论的内容首先是功耗预估和分析对布局布线的指导意义,然后根据前面所述的功耗分析和预估理论,对一款 450 万门的超大规模芯片的功耗在各个层次上进行预估。

一、功耗预估和分析对布局布线的指导意义

综合后，设计进入物理设计阶段，主要包括布局、布线、DRC、LVS 等步骤。

版图规划（Floorplan）就是根据综合后报告的设计规模，估计整个芯片的大小，根据电路的逻辑功能和模块之间的连接关系大致确定宏模块（Macro）的位置，同时要考虑 I/O Pad 和 Power/Ground Pad 在 core 周围的排列顺序，以使 core 内部的逻辑电路与 I/O Pad 的连线最短，减小连线延迟，然后对电源线和地线网络进行设计。规划的结果将直接影响到芯片的工作时序和电路性能。

在电源线地线布线之前要对功耗进行预估，根据功耗的预估值，及由库提供商给出的 I/O 最大承受电流值和所允许的邻近电源地同时翻转的输出数目限制，确定需要多少对 Power/Ground PAD，其中前一个方面是满足芯片供电的需要，后一个方面则是为满足电源完整性的要求。然后再考虑到 Metal 的最大电流密度，就可以确定 P/G ring 的宽度和 power straps 的数量、宽度及间距。因为 P/G ring 对越靠近芯片中间的 cell 供电能力越弱，靠近芯片中心区域的 IR Drop 也是最大的，所以靠近芯片中间的位置 power straps 的间距要小一些，以提供足够的电流。

电源线和地线是两条几乎涉及整个芯片各个位置的全局线网，它的电性能保障和布线结果优化会对芯片产生直接的影响。电流流过电源线（或地线）会带来电压降及金属电迁移问题。在电源线和地线的布线过程中，若不考虑这两个问题，则当有过大的电压降就会导致逻辑错误或降低开关速度，而金属电迁移，则可导致电源线地线发生断线而过早失效。为了减小电源线地线上的电压降及电流密度，通常，电源线地线的宽度远大于一般信号线的宽度^[8]，但是会增加芯片面积，浪费更多的布线资源。因此需要尽可能准确地预估出功耗值，以尽量减小布电源线地线时所留有的余量宽度，使它们占用的芯片面积最小。

以上的这几个方面都要求能够在布局布线前，对整个设计的功耗有一个较为准确的预估，来大致确定 Power/Ground ring 的宽度、power straps 的数量、宽度及间距和需要的 Power/Ground Pad 的数量，而保证所进行的电源地 I/O 数目的选择和电源线地线的布线能够满足设计的需要。

二、450 万门的超大规模芯片的功耗预估

本设计的功耗预估主要在以下几个阶段：

阶段一：应用最坏情况活动率和较为准确的线负载模型进行预测。

阶段二：应用真实的活动率和准确的线负载模型进行预测。

阶段三：利用提出的线负载寄生参数和基于门级仿真的真实的开关活动率进行预测。

阶段四：流片后的实测结果。

表 2.2 各种阶段不同工作模式下的功耗（单位 W）

	功能一	功能二	功能三
阶段一	4.032		
阶段二	3.525	2.095	3.601
阶段三	2.957	2.336	2.980
阶段四	2.066	2.736*	3.075*

*整个测试板的功耗，包括芯片及其他部分逻辑电路

表 2.2 列出了在四个阶段，对于芯片在不同工作模式下功耗分析和测试的结果。除功能二外，其他部分完全符合第三节中所介绍的功耗预估理论，并且功能二所出现的偏差也是在能够接受的范围内，可以视为误差。

整个过程都很好地验证了，功耗预估的准确性和设计过程中所能提供信息的准确程度是紧密相关的功耗预估理论。

图 2.3 是所采用的功耗流程图。

总结

本章中我们首先介绍了数字 CMOS 电路的功耗模型，然后介绍了各个层次上的低功耗策略，各个层次上的功耗预估方法和需要的信息。表明在 RTL 层次上，对所进行设计的功耗进行分析，能够较快地得到功耗结果。虽然比较粗略、不够精确，但能够对整个设计进行一定的指导，如果和预期的功耗相差较大，则可以很快返回 RTL 代码，在结构上进行功耗考虑和修改而得到合适的功耗；对于确定布局布线时的电源和地的 Pad 数和布线金属宽度也有一定指导意义。

第三章 动态电压缩放电路

动态电压调节技术是一种通过将不同电路模块的工作电压调低到恰好满足系统最低要求来实时降低系统中不同电路模块功耗的方法。该技术基于这样一种观察结果：即电路模块中的最大时钟频率和电压是紧密相关的。如果一个电路能够估算出它必须做多少工作才能完成当前的任务，那么从理论上讲就可以将时钟频率调低到刚好能适时完成该任务的水平。另外，降低时钟频率还意味着可同时降低供电电压。所以，电路就从以下几个方面降低了功耗：更低的时钟频率、更低的电压、更低的漏电流(因所有的漏电流都与电压成正比关系)。

第一节 DVS 概述

一、DVS 的概念

在确定了数据通路的体系结构，速度和面积后，可以通过选择供电电压，阈值电压和器件尺寸来对功耗进行折衷优化。这就会引出不同的减小功耗措施，如表 3.1 所示。

表 3.1 供电电压，阈值电压和器件尺寸对功耗的优化方法

	固定因素	休眠模式	工作期间
动态功耗	低 Vdd, 多 Vdd, 晶体管逻辑优化	门控时钟	动态电压缩放
漏功耗	多阈值	关断管, 可变阈值	可变阈值

供电电压的减小，可以降低每一个操作任务所消耗的能量，延长电池的供电时间，前提是以牺牲系统的性能为代价。对于应用在快速响应的场合，这种性能的下降是不可接受的。在实际应用中，并不需要处理器总是工作在峰值性能上，这样降低电压来节省功耗就变得可行起来。以一个用在便携设备中的通用处理器为例。处理器的计算任务可以分为三个主要类别：高计算强度任务、

低速计算和空闲模式操作。高强度计算和短响应任务需要处理器全速大吞吐量地运算，尽可能达到实时的目的，MPEG 和音频解码属于这方面的例子。低吞吐量和长响应任务，比如文本处理，数据导入和存储文件备份可以在相对很宽松的时间期限下进行处理，只需要有相对较小的吞吐量就可以完成。若仍在全速下计算，就会很快地完成计算任务，其后是长时间的等待，这时的能量消耗是不必要的。另外便携处理器会有较长时间的空闲状态，等待用户的外部唤醒。总起来说，这类处理器所要求的计算吞吐量和处理时间是动态时变的。

即使是高强度的计算操作，比如MPEG解码，在处理数据流时也表现出随计算需求的动态变化。每一帧图象所要计算的离散余弦变换（IDCT）数目依赖于画面的运动情况会有很大的变化。如图 3.1，描述的是典型的图象处理中每帧的IDCT数目分布。可以看出处理这个转换时，处理器的计算负荷会有很大的变化^[9]。

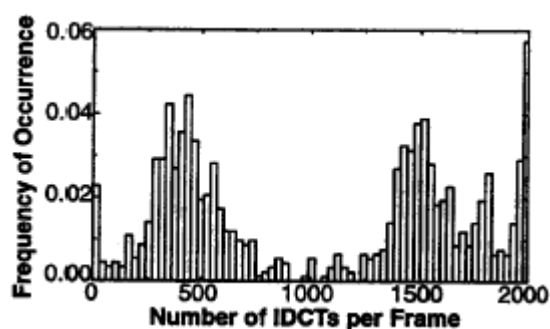


图 3.1 每帧中 IDCT 的数目分布

在处理低负荷时，可以减小处理频率，随着频率的降低，电压也可随之降低，这样就能够极大地减小能量消耗。为了保证在高负荷时维持最大吞吐量，在低负荷时尽可能节省能量，根据应用需要对电压和频率进行动态的调节，这种技术就是动态电压缩放技术(Dynamic Voltage Scaling)。如图 3.2，图示了每个独立操作所要消耗的能量与吞吐量的关系。最上面线是固定 3.3V 电源电压下的情况，当电压固定时，即使吞吐量降低，由于要完成同一操作，其处理的周期数是固定的，也就是说处理的相应时间要延长，导致每个操作的能量消耗仍然是恒定的；下面一条实线是动态电压缩放后的能耗曲线。可以看出，在处理速

度降为原来的 10% 时, 电压可以从 3.3V 降低到 1.05V, 处理相同的操作时, 能量消耗可降到原来 $1/10^{[10]}$ 。

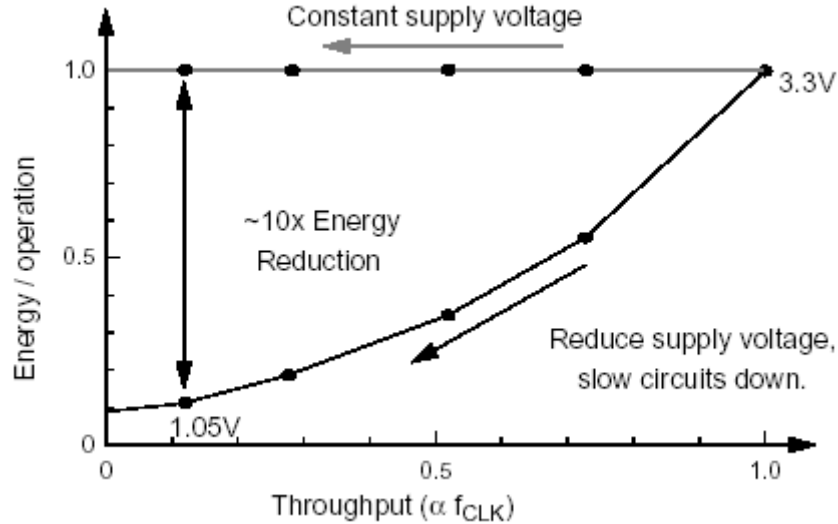


图 3.2 每个操作的能量消耗和吞吐量的关系图

在实际调降电压之前, 系统必须得到最小的时钟频率是多少。为了精确地控制DVS, 需要一个负荷调度部件来实时地改变电路的工作频率, 负荷调度部件分析当前和过去状态下系统工作情况的不同来预测电路工作负荷的变化, 根据已处理的和当前所处理的数据量, 预测下一步的工作负荷, 给出对应的工作频率, 可靠的预测是实现低功耗的保证。很多文献上都提到了在硬件和软件上实现这一预测的方法^{[11][12]}, 当然, 这些算法是和具体的应用紧密相关的。

国外的许多院校和研究机构都进行了电压缩放算法方面的研究, 提出了经典的最早预期 (Earliest Due Date EDD)、最早期限优先 (Earliest Deadline First EDF) 和比率单调调用 (Rate Monotonic Schedule RM) 任务算法^[11]及一些改进的算法, 以给出系统需要的最小时钟频率。概括说来, 这些都是基于降低功耗和保持实时性能两方面进行折衷而实现的。

二、DVS 的可行性

下面的公式给出了延迟与供电电压 V_{dd} 的关系:

$$f_{clk} < \frac{1}{delay_{max}} = \frac{(V_{dd} - V_k)^\alpha}{V_{dd}}$$

其中 α 和 V_k （阈值电压）是和CMOS的工艺条件相关的。

所以数字电路的传输延迟依赖于工艺条件和供电电压。

数字系统的最高工作频率是由其关键路径上的最大延迟决定。

回到功耗模型，我们可以看出功耗和电压的平方成正比，所以降低供电电压可以极大地降低功耗，但降低电压会导致延迟的增大，使关键路径的时间变长，而导致最高工作频率的下降，使系统性能降低。

但是根据图 3.3^[13]，各种门或逻辑电路块从简单到复杂，他们的延迟随电压的变化趋势是相同的，其中包括了反相器、寄存器、晶振环和SRAM。这样就保证了电压降低后，虽然工作频率会有所降低，仍能保证其逻辑功能不变的效果，为DVS的实现提供了可能。

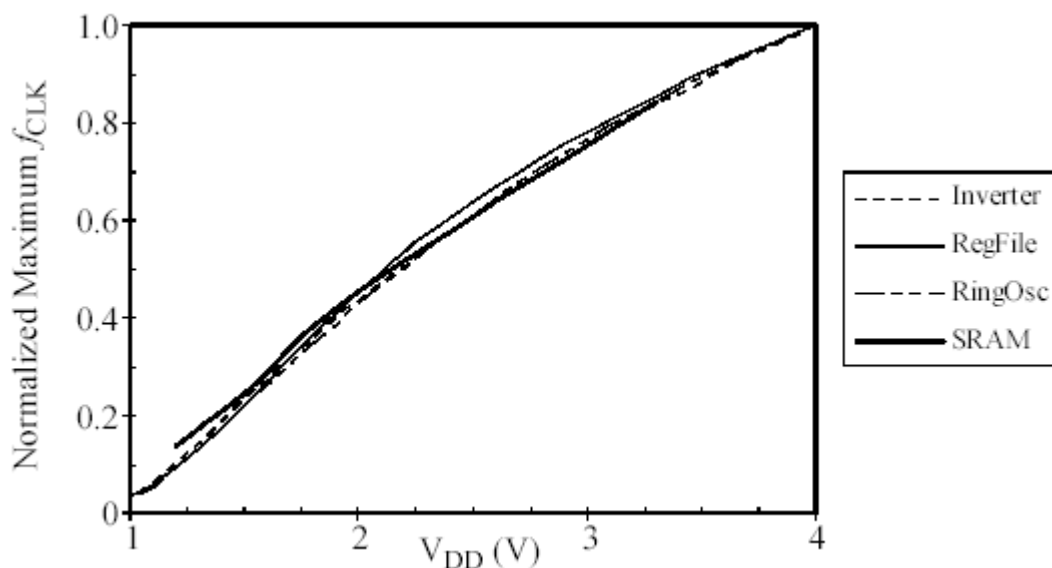


图 3.3 不同电路的延时和电源电压的关系

实际情况的工作负荷是时变的，并不要求处理器总是工作在高强度下。在保证规定时间内完成任务的情况下，可以适当降低处理器的工作频率，增加关键路径的延迟，也就是说电路可以工作在低压下，达到降低功耗的目的。

图 3.4 是一个 DVS 系统的原理框图。简单的说，DVS 就是芯片的电压根据其工作负荷而动态变化。当计算负荷要求低时，提供低的供电电压；计算负荷高时则提供高压值，从而达到既能节省功耗又不牺牲电路性能的目的。动态电压缩放是在处理器工作时对处理器性能和功耗的一种折衷。

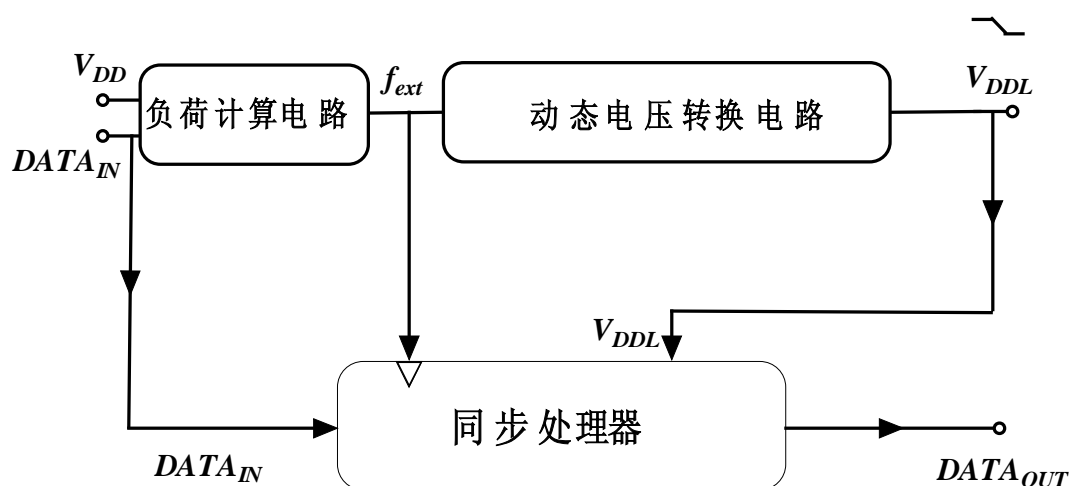


图 3.4 动态电压缩放系统图

第二节 DVS 的适用范围

一、处理器的分类

首先根据处理器的工作方式不同，可以分为三大类：固定吞吐量模式(Fixed Throughput)、最大吞吐量模式(Maximum Throughput)和突发吞吐量模式(Burst Throughput)。吞吐量指的就是固定时间内完成的操作。

象 DSP 这种类型的处理器，它们实时输入的语音、图象信号速率都是基本固定的，处理器的吞吐量也是固定的，属于固定吞吐量模式。

对于网络服务器或给多个用户提供服务的主机系统，它们处理器总是处于工作状态，运算速度越快越好，属于最大吞吐量的工作模式。

PC 机、便携机和 PDA 中的处理器在相当一部分情况下都工作在空闲状态，在响应用户的请求后才进行计算工作，属于突发吞吐量工作模式。

这里提出的 DVS 系统的应用范围主要针对的就是最后一种工作模式的处理器，利用 DVS 后，它们的功耗会有显著的减小。

二、突发吞吐量工作模式

突发吞吐量工作模式的处理器实际的工作负荷是时变的，会在不同时段内处理高运算强度任务，或低强度任务，如图 3.5。

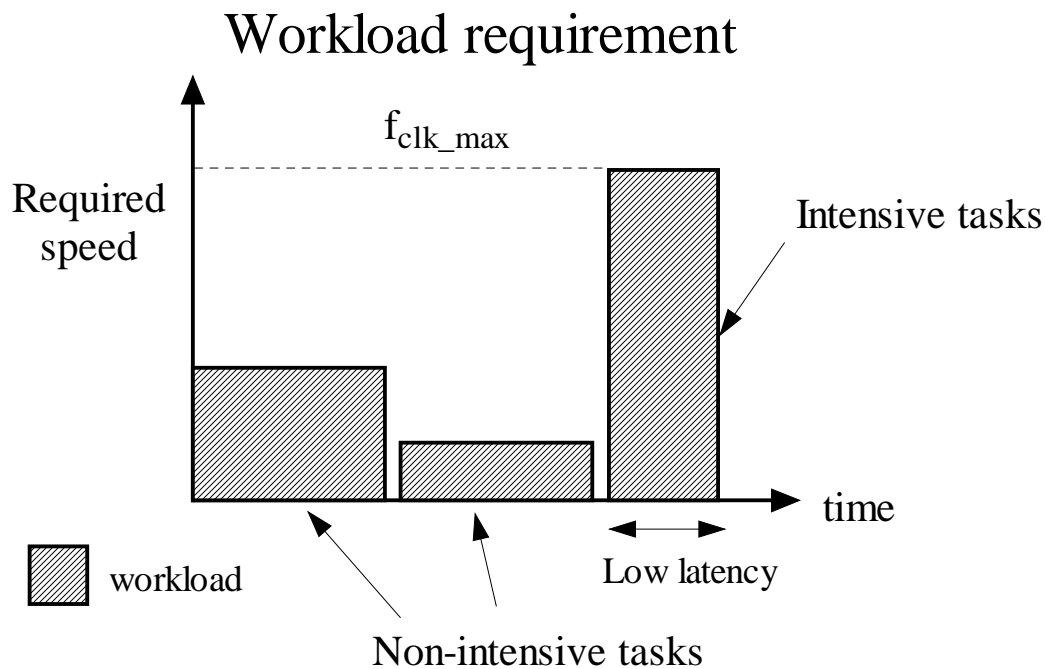


图 3.5 突发吞吐量工作模式的工作负荷

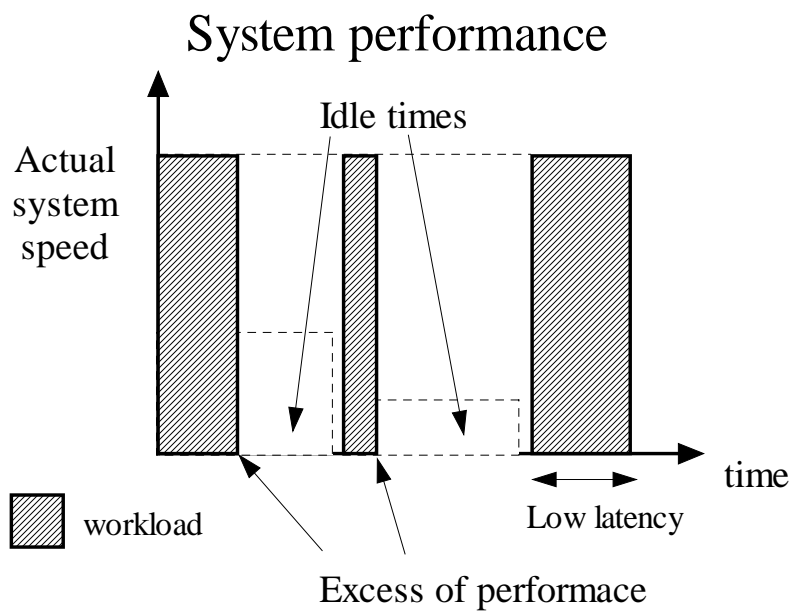


图 3.6 突发吞吐量工作模式的处理器性能图

在通常的数字系统中，频率和供电电压是固定的，为保证系统性能，频率需要满足处理最高运算强度的最大速度。对于非最高运算强度的任务，处理器一样也会用最高频率来处理，就会在任务处理完毕后，出现一段系统的空闲时间，如图 3.6。

这时就可以考虑利用这一段空闲时间，在保证规定时间内完成任务的前提下，降低处理器的工作频率，也就是说可增加关键路径的延迟，因此电路可以工作在低压下，达到降低功耗的目的，实现的转换如图 3.7 所示。

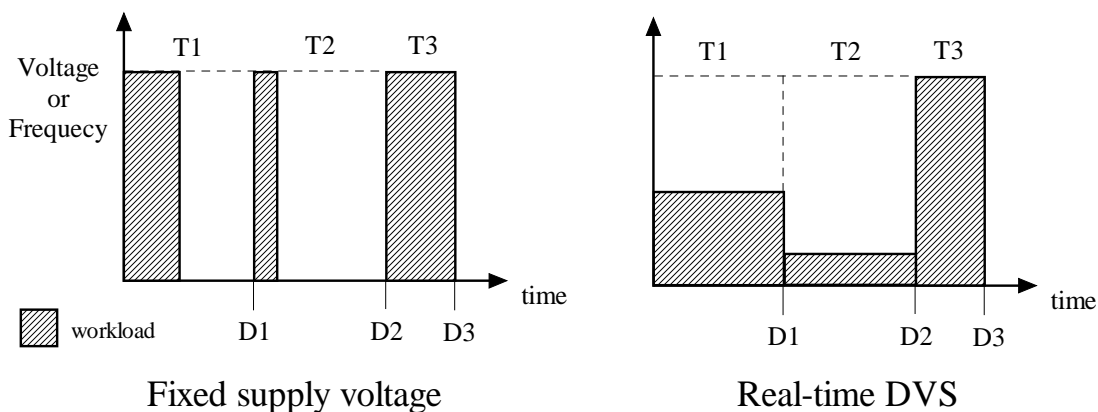


图 3.7 固定电压和 DVS 系统性能对比

DVS 除了对处理器的工作模式有一定要求外，由于供电电压需要在一定范围内变化，电路的实现形式应该注意两点：

1. 尽量采用静态逻辑

这是因为动态逻辑，比如 NMOS 传输门，因为其传输逻辑“1”时会会有一个阈值损失，在用接近阈值电压的电源供电时会导致逻辑错误，除非能够保证供电电压一直在 2 倍 V_{th} 以上。

2. SRAM 中灵敏放大器 SA 不能采用电荷转移型等更复杂类型的 SA

因为为了提高响应时间和降低功耗所设计的复杂类型的 SA，只能由固定电压供电，用可变电压供电则会失效。

也就是说，电路中的待控部分应该是在一定范围变化的电源电压下，能够保证正确功能的电路。

第三节 DVS 的应用

在很多移动设备中，DVS 得到了广泛的应用，笔记本，手持设备。90%的笔记本中都采用了这一技术。满足了在保证性能基本不变情况下，功耗的极大节省。

几大微处理器生产厂商在处理器中都已应用了这一功耗节省技术，虽然命名不同，但核心都是 DVS 技术，如表 3.2。

表 3.2 几大微处理器应用的 DVS 技术

Technology	Voltage Range(V)	Frequency Range(MHz)	Energy Savings	Year
LongRun(Transmeta)	1.2-1.6	500-700	44%	2000
Powernow!(AMD)	1.4-1.8	200-500	41%	2000
Xscale(Intel)	0.75-1.75	150-800	84%	2001

可以看出，工作电压根据需要可在一定的范围内变化，功耗有显著的降低。例如，应用了DVS技术的Crusoe处理器可以工作在最大频率 700Mhz，功耗为 5.5W；在处理相同的数据类型和数据量时，Pentium4 处理器工作于 2Ghz，功耗为 90W^[14]。

不过，这样一个相对来说显而易见的策略实现起来却需要对几乎所有的设计层面进行修改：单元设计、模块设计、模块互连、芯片规划、系统架构、以及操作系统和应用设计。

通常，在 SOC 系统中，有些部分的电压是要保持不变的，比如外部的存储器就需要工作恒定的电压下，所以在设计分析分块和做功耗预算时，应该整个系统来考虑功耗的节省。

Synopsys、ARM 和 NSC 公司正在开发一种使电压/频率隔离孤岛、电平转换器和控制电路更易实现的设计流程。

使用多电压岛或多电压区域，可以满足功耗和性能两方面的要求。划分区域的依据是根据功能的不同将部分逻辑放在一个单独的区域。在需要高速工作

的区域使用高电压，对时序要求低的区域供给低电压。电压的缩放一般是根据频率的缩放进行。

为了实现多电压供电，必须提供不同的电源引脚和集成在芯片中的模拟电压转换器。电压转换器的效率必须包括在整体功耗的计算范围中。如果仅仅是一小部分需要用到低电压，则在电压转换器上消耗的功率可能会大大超过通过利用低电压实现的功耗节省，这是要进行折衷衡量的。在进行电压岛设计时，需要电平转换器单元，来保证信号在不同的电压岛之间能够正确传输。

另外，为了更进一步降低电压值，可以根据系统的实时需要动态地来降低岛的供电电源电压值。应用这种方法，需要提供在所有电压下的库单元信息。Synopsys Scalable Polynomial Models (SPMs) 提供了需要的时序和功耗信息，Non Linear look up table Models (NLMs) 也可以被用在电压岛的设计中。

Synopsys、ARM, National Semiconductor 和 Artisan Components 共同开发了一款芯片，并验证了可以极大地降低功耗。芯片应用了专门的硬件和软件在不同的区域来控制电压和频率，应用高层次的电压频率缩放技术。

控制器件包括ARM的Intelligent Energy Manager软件，用来平衡处理器的工作负荷和能量消耗。NS的PowerWise硬件监测处理器的性能，并和电压转换器进行交互，在每一个工作频率上，降低供电电压到最低值。这个系统还能补偿因为制造工艺和工作时因为温度的波动而造成的硅片的性能偏差^[15]，达到了很好的低功耗效果。

第四节 DVS 的性能指标

由 DVS 的具体应用决定了两个基本的目标，

1. 根据给定的处理器吞吐量命令，应尽可能地减小整个 DVS 系统的能量消耗。
2. 根据要求，应尽可能快的转换电压值。

DVS 系统可以用多种方式来实现，但所有的策略都必须有数字系统和电压转换器之间的交互。最简单的方法是事先了解一个给定任务的延时要求和实时

底线，并将该信息提供给电压转换器。不过，通常数字系统的应用者并不能不享有该信息，或至少对很难分析出该信息。因此解决方法就是通过硬件或软件，对任务进行分析和预测，比如 ARM 采用了在操作系统代码中嵌入性能监视器的方法来了解每个应用运行对系统的任务要求。在假定每一任务都需要 CPU 全速运行之后，监视器就可给出实际性能估计，随后这一信息将提供给硬件，由硬件来实现电压的动态缩放。

一、跟踪时间

根据预测的结果，给出相应的工作频率时钟。当数字系统希望改变时钟频率，电压转换器的控制端接收命令后，进行电压调整，同时继续监测电压的变化对系统的影响，直到满足要求为止。因为转换器输出电压的连续性，转换器不可能瞬时地实现电压值的改变，在两个稳定电压值之间，需要有一个跟踪时间。

改变电压的跟踪时间如果过大，就会使待控芯片工作在高于最优的电压下，而使节省的能量也没有达到最优。如果系统在单位时间内所要求的电压变化非常大，即每秒需要在很多个稳定的电压值之间进行切换时，因为两次电压变化必须在要求的时间内完成，跟踪时间就处于一个比较显著的位置，不同的应用对电压切换的要求也不同。例如，Transmeta 中的 LongRun 技术，支持在每秒钟内有 200 次电压变化，这意味着每个稳定状态的时间为 5ms。在现在的电压转换器中，可以方便地实现这样的要求。但如果为节省更多的能量，在需要更高的变化率时，使跟踪时间尽可能地缩短就显得非常重要了。

解决跟踪时间过长的另外一个途径是关闭时钟，等待电压稳定后再打开时钟，这样导致的后果是系统的性能无法得到保证。

二、跟踪能量

跟踪能量是指在输出电压变化过程中，消耗在转换器上的能量。在转换率很快的情况下，每次转换电压的能量消耗也要考虑在内，这是动态电压转换的另一个指标。

在高性能的动态 DVS 系统中，跟踪时间和跟踪能量将是衡量性能的重要指标，并会对系统的能量节省产生重要影响。因此他们应该在整个系统中进行考虑，以保证最大限度地降低功耗。

下面由各个指标的数学表达式可以直观地得到进行折衷的方法。

纹波电压表达式：

$$\Delta V_{PWM} = \frac{V_o \cdot (1-D)}{8 \cdot L \cdot C \cdot f_s^2}$$

跟踪时间表达式：

$$\tau_{LC} = \sqrt{L \cdot C}$$

影响这些指标的主要因素是滤波电容的大小。因为 DVS 系统中的电压转换器需要快速地改变电压值，要求跟踪能量尽可能地小，所以需要使输出电容尽可能地小。另一方面，低的输出电容可能会导致：

1. 滤波开关噪声（静态调整时）
2. 在负载电流改变时使输出电压发生动态变化

这些问题都可以通过提高转换器的开关频率来解决，开关频率越高，就需要越小的电容和电感，会得到越好的动态效果。开关频率增加了，在开关管中的驱动损耗和开关损耗就会增加，在很高的频率下(达到 MHz)以上时，寄生效应在开关过程中会逐渐显著起来，会增加设计的难度。提高开关频率的唯一途径就是把电压转换器和待控芯片集成在一个芯片上，因为这样电源到地的路径变小了，寄生电感和电容在高频下的寄生影响就会相应地减小。

第五节 动态 DC-DC 转换器的设计考虑

电压的缩放最终是由动态电压转换器来实现，因此动态电压转换器是 DVS 系统的关键部件。DVS 的设计初衷是减小系统的功耗，延长电池的使用时间，这就需要转换器具有较高的能量转换效率，这也是选择开关电压转换器，而非线形转换器的原因。开关电压转换器中，可分为 Buck（降压）型，Boost（升压）型和 Buck-Boost（混合）型。考虑到 Buck 型的高效率，在 DVS 电路中一般选用 Buck 型电压转换器^[4]。

现在DC-DC开关转换器因为其极高的效率 90%-92%和灵活性,作为低电压转换器应用在各种移动设备中,比如PDA,笔记本,手机和充电器中^[16]。

应用在 DVS 系统中的动态电压转换器相比传统的来说必须具有更小及更高的性能。

对于应用在 DVS 系统中的电压变换器有下面的几个要求:

1. 小尺寸
2. 在整个电压工作范围内保持高的转换效率,以及低的跟踪能耗
3. 根据负载的需要快速地改变输出电压
4. 稳定的静态调整功能

尺寸和效率是互为矛盾的两个方面,在实际设计中,必须根据具体应用进行相应的折衷。小的尺寸意味着高的开关频率,这样就可以减小电压转换器中具有最大尺寸的两部分——电感和电容的大小。

另外, DVS 必须在很宽的输出电压和电流范围内保持转换器的高效率,以使整个系统得到能量的最大节省。在过去的电压转换器设计中,总是在一个固定的输出电压值上保持高的效率,不能根据负载的变小来调整转换器而使损耗也变小,当输出低电压时在控制级和驱动级上的损耗将会占主导地位。电压转换器的静态和动态的合理折衷也是一个主要问题。设计中希望提供的电压尽可能是可以随意变化的,或者是以很小的代价就可以设置电源电压到任意低的值。

一、Buck 型转换器

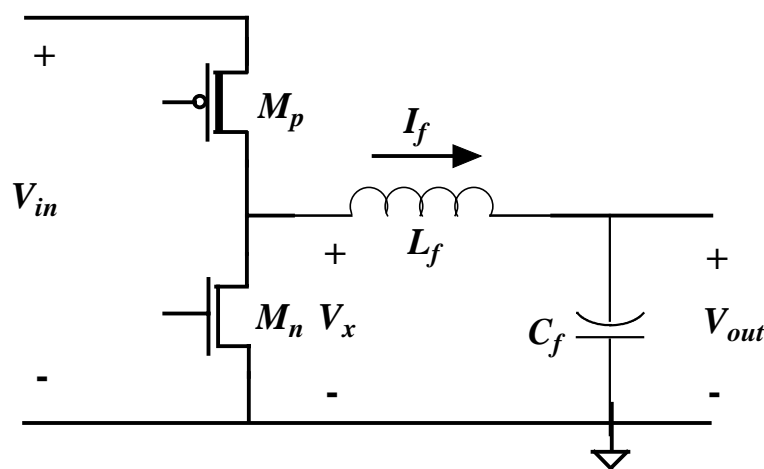


图 3.8 Buck 型转换器原理图

低输出电压 Buck 型电压转换器，如图 3.8，可以输出 $0 \leq V_o \leq V_{in}$ 范围内的任意电压。基本的 PWM 工作原理如下：通过晶体管导通与截止在反相器的输出 V_x 处得到占空比为 D ，频率为 $T_s = f_s^{-1}$ 的方波。在一个周期内， V_x 处的电压波形如图 3.9。

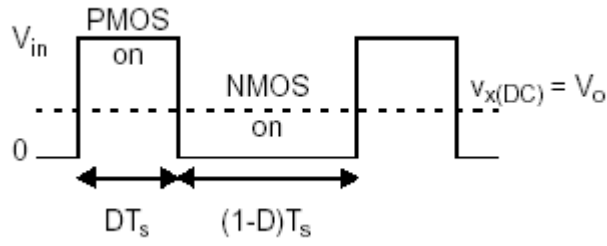


图 3.9 V_x 处的电压波形

通过二阶低通滤波器（ L_f 和 C_f ），就可以滤出方波中的直流平均量，减小交流量到可以接受的数值。理想情况下，直流输出电压为输入电压及占空比的乘积： $V_{out} = V_{in} D$

二、PWM 方式的工作原理^[16]

以 PMOS 器件 M_p 打开为起始。在一个开关周期 T_s 的时间间隔 D 内，反相器的输出节点 V_x ，和 V_{in} 短接。在电感两端会有有一个恒定的正电势差 $V_{in} - V_o$ ，电流 I_{L_f} 会线性地从最小值增加到最大值。电池中一部分能量就储存在电感中，另一部分传给滤波电容和负载。

PMOS 关断，NMOS 管 M_n 打开后， M_n 的作用是为电感续流，将 V_x 短接到地。在周期中的 $(1-D)$ 部分，电感两端会有有一个恒定负电势，电流 I_{L_f} 会线性地

从最大值减小到最小值。电感中存储的能量这时就转换到滤波电容和负载上去了。上面这个过程在以后的周期中不断重复。

定量分析如下：

为了使讨论简化，下面把各元件都视为理想器件。

当 PMOS 导通时， V_x 节点电压等于输入电源电压 V_{in} ，此时有：

$$V_L = V_{in} - V_{out} = L \frac{di_L}{dt}$$

当 NMOS 导通时， V_x 节点电压近似为零，得到：

$$V_L = -V_{out} = L \frac{di_L}{dt}$$

电感储能 $\frac{1}{2}LI^2$ ，系统要工作在稳定状态下，需要使电感在一个开关周期内储存与泄放的能量相等，也即是在一个开关周期内电感电流的变化量总和为零，于是可以得到：

$$di_L = \int_0^{ton} (V_{in} - V_{out})dt + \int_{ton}^{toff} (-V_{out})dt = 0$$

t_{on} 与 t_{off} 分别表示 PMOS 和 NMOS 的导通时间，用 T_s 表示开关信号的周期，

由 $T_s = ton + toff$ ，将上式化简解出下式：

$$V_{out} = V_{in} \times \frac{ton}{T_s} = V_{in} D$$

其中 $D = ton/T_s$ 为开关信号的占空比。可以看出，占空比 D 代表了输入输出电压关系，在 Buck 开关变化器中输入输出是线性的关系，改变 D 的值就可以通过二阶低通滤波器获得所需要的直流电压。

三、 输出滤波器的设计

二阶低通滤波器（ L_f 和 C_f ）的作用就滤出方波中的直流平均量，并减小交流纹波量到可以接受的数值。滤波输出端供给负载 R_L 直流电流 I_o 。

在实际供电电路中，为了达到滤波的目的，须使 $L_f \cdot C_f \gg \omega_s^{-2}$ ，其中 $\omega_s = 2\pi f_s$ ， f_s 是转换器的开关频率。满足这个条件后，就可以只在时域内考虑电感电容值的大小，而不必再在频域内分析。

忽略输出电压纹波（ V_x 峰峰值 $\gg V_o$ 峰峰值），假设方波周期为 T_s ，电感的电流波形是一个周期为 T_s 的三角波，峰峰电流波动为 ΔI ，并相对于输出电流 I_o 是对称的。一个周期内， ΔI 可由在时间段 D 上对电压 $V_x(t)$ 积分得到

$$\Delta I = \frac{V_{in} \cdot D \cdot (1-D)}{L_f \cdot f_s} = \frac{V_o \cdot (1-D)}{L_f \cdot f_s}$$

考虑没有负载情况下，输出电压的纹波由下式决定：

$$\Delta V = \frac{\Delta I}{8 \cdot C_f \cdot f_s} = \frac{V_o \cdot (1-D)}{8 \cdot L_f \cdot C_f \cdot f_s^2}$$

纹波电压以输出电压 V_o 为对称，与周期 T_s 呈分段平方律的关系。

上面两个公式是最小化电感电容值时的两个基本关系。可以看出所需的电感电容值随着 f_s^{-1} 下降，开关频率越高，转换器就可以设计得越小。对于输出电压纹波，与 L_f 和 C_f 的乘积有关，而不仅仅是与一个单独器件有关系。这些都助于指导转换器的设计。

四、动态电压转换器的控制

前面介绍的“静态”电压 DC-DC 转换器，在设计初始，就确定了一个固定的输出电压值，也就确定了对应的转换效率。在这种 DC-DC 转换器中，负载和转换器之间没有交互，转换器将输出的回馈，和一个固定的电压参考源做对比，然后经过脉宽或脉频调制得到稳定的输出。

动态电压缩放(DVS)系统需要在处理器工作的情况下，通过改变转换器的输出电压，对待控处理器的性能和功耗进行动态折衷。这种动态 DC-DC 转换器和它的负载必须进行交互，得到和工作负荷相关的正确电压值。

因为其应用范围不同，动态 DC-DC 转换器和静态 DC-DC 转换器在有些要求方面存在着差异。相同之处在于两种转换器都要求在调整过程中，高效率地输出 DC 电压。动态 DC-DC 转换器还必须在很宽的电压和电流范围中达到这个要求。另外动态转换器须在所要求的转换时间内完成输出电压的改变，达到伏每毫秒的变化率。

为一个模块的给定频率找到尽可能最低的电压，并引导电源提供该电压，闭环控制系统技术是适合解决这个问题的一种合适方案^[17]。

比如 NSC 公司设计了一个实时地为一个模块设定一个正确的最小电压的方法性能监视器的可综合结构，将其放在靠近关键路径或模块中散热最厉害的地方。它们用作关键电路的性能行为模型。通过用性能监视器来测量延时，对电路能正常运行的频率进行实时测量。

在一个数字反馈控制环路中使用该信息，控制器能根据需要对供电电压发布升高或降低的调整命令，以期在希望的时钟频率下，调到最小的工作电压值。这一基于监视器的反馈方法能自动地根据温度变化甚至芯片内工艺参数的改变进行修正。

图 3.10 示出一个本论文实现的利用电压和频率跟踪闭环来实现的动态电压缩放系统。处理器负荷调度部件根据负荷大小给出处理器所需的最低实时工作频率，提供给处理器和动态电压转换器。动态电压转换器由速度检测器，计数控制器和 BUCK 型 DC-DC 转换器组成。速度检测器中对处理器的关键路径进行了复制，根据实际电压和所需电压产生一个数字的误差信号，送入计数控制

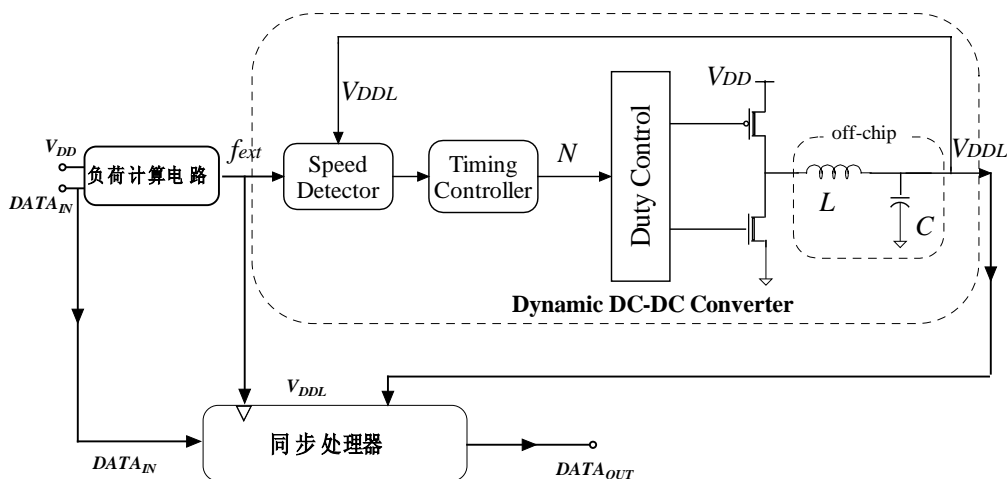


图 3.10 动态电压缩放系统

器，得到相应的负载数 N 。DC-DC 转换器则根据负载数 N 供给处理器满足需要的电压 V_{DDL} ，并对电池电压和负载电流 I_{DDL} 的变化进行调整，得到相对稳定的 V_{DDL} 。因此，处理器工作在最低的供电电压下，并且对数据的处理要求依然能够满足。

第四章 动态电压缩放控制电路的实现

这一章里，介绍了一款动态电压缩放电路的具体实现，是利用电压和频率跟踪闭环技术来实现的动态电压缩放，其功能是根据负荷调度部件所给出的处理器最低实时工作频率，输出相应的最低供电电压。

第一节 DVS 原理框图

本章所要实现的动态电压缩放原理框图如图 4.1：

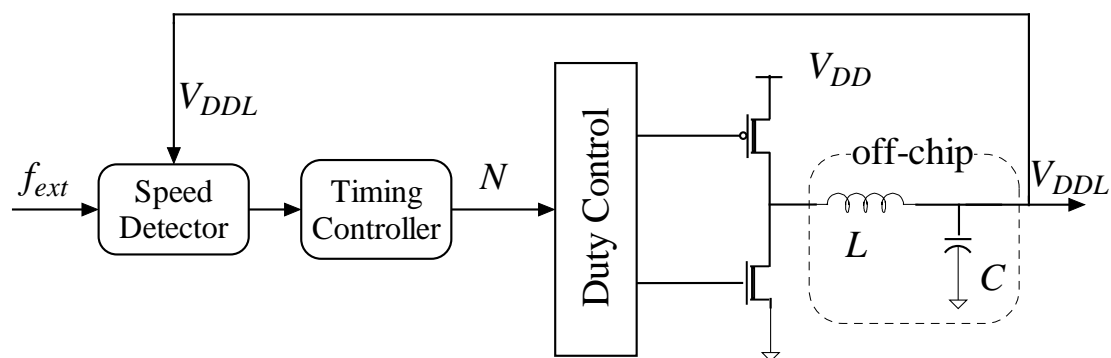


图 4.1 动态电压缩放原理框图

由三个主要部分组成：

1. BUCK 型降压转换器，
2. 计数控制器，
3. 速度检测器。

BUCK 型电压转换器产生 $(N/64) \cdot V_{DD}$ 的供电电压 V_{DDL} ，为待动态电压缩放的电路部分供电，其中 N 是由计数控制器生成的从 0 到 63 的整数。因此对于 $V_{DD} = 3.3V$ 情况下， V_{DDL} 的精度大约是 50mV。占空比控制器产生占空比为 $N/64$ 的方波，通过外接的电感 L 、电容 C 所组成的二阶低通滤波器，得到电压的平均值 $(N/64) \cdot V_{DD}$ 。为了保证供给芯片的电压在工作电压范围以内，可以

通过在计数控制器中增加一些逻辑，来控制 N 的取值来达到，其中最小 N 值对应最小电压值，最大 N 值对应最大电压值。

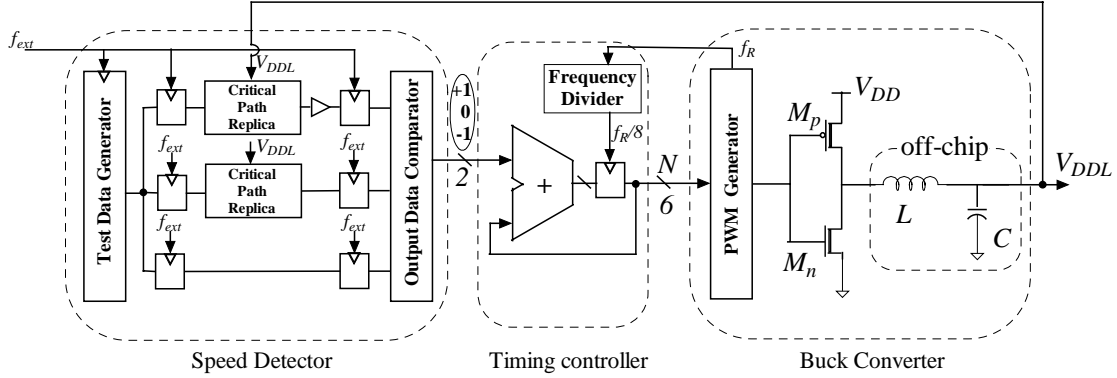


图 4.2 动态电压缩放结构图

图 4.2 是所要实现的动态电压转换器和其反馈控制回路的详细描述。计数控制器通过累加速度检测器的输出值来得到 N 的值，若速度检测器输出+1 则提高 V_{DDL} ，-1 则降低 V_{DDL} ，0 则维持 V_{DDL} 的电压大小。计数器的时钟是取自 BUCK 型电压转换器中 PWM 中振荡环所产生的方波，经过 8 分频后得到。

速度检测器是通过监测所复制的待控芯片关键路径上的延迟来工作的。当 V_{DDL} 相对于电路频率 f_{ext} 来说太低，不能满足路径上的延时要求时，速度检测器输出+1 来提高 V_{DDL} ；反之，当 V_{DDL} 太高，速度检测器输出-1 来降低 V_{DDL} 。通过这个反馈控制，DVS 就可以产生既能满足待控芯片频率要求，又可以尽可能地得到最低的供电电压 V_{DDL} 。为了留有一个安全余量，应用在速度检测器中的关键路径应该比所复制的关键路径的延迟略大一些。

速度检测器的周期是基于 f_{ext} 的，要比低通滤波器的时间常数短得多，所以这个反馈回路可能会出现振荡。8 分频后得到的计数器的频率可以保证产生快速稳定的反馈控制，具体的分析请见第二节有关计数控制器部分的讨论。

第二节 电路的实现

下面是对实现的动态电压缩放电路，分三大组成部分进行详细的介绍。

一、BUCK 型电压转换器

设计的 BUCK 型电压转换器的结构框图如图 4.3。

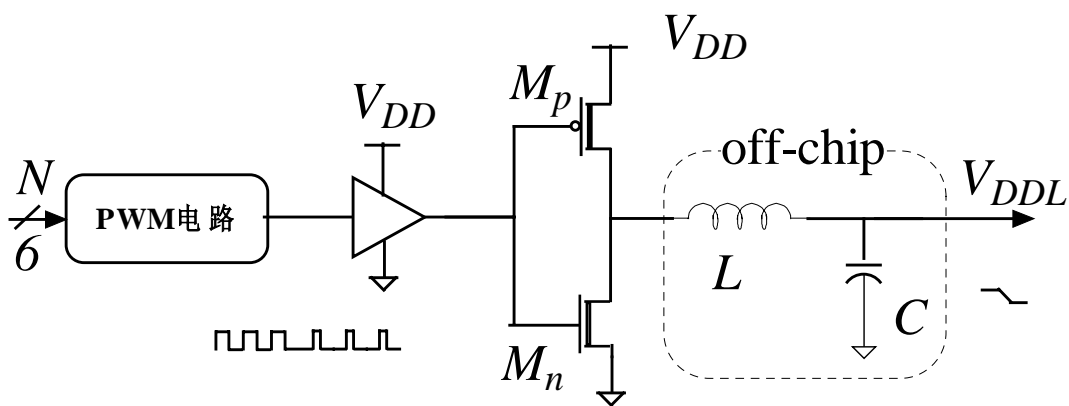


图 4.3 Buck 型电压转换器结构框图

应用在 DVS 电路中的 Buck 型电压转换器的功能是：根据上一级计算出的负荷量 N ，给出与之相适应的 V_{DDL} ，作为待控电路的供电电压，以期实现低功耗。其中 V_{DDL} 与 V_{DD} 的关系如下：

$$V_{DDL} = \frac{N}{64} \cdot V_{DD}$$

1. 低功耗小面积 PWM 发生器的设计

这里要介绍的 PWM 发生器结构图如图 4.4，这是一个可控占空比的发生电路。需要的主要硬件：由 16 个反相器和 1 个与非门构成的振荡环，16 选 1 MUX 和 4 选 1 MUX，一个异或门 X 。

$S0 \sim S5$ 代表 N 的值： $S0$ 低位 (LSB)， $S5$ 高位 (MSB)。

$$N = S0 + S1 \cdot 2 + S2 \cdot 2^2 + S3 \cdot 2^3 + S4 \cdot 2^4 + S5 \cdot 2^5 \quad (1)$$

16 选 1 MUX 从振荡环路的 16 个节点中选出一个信号。在点 S' ，信号 B 改变。

$$S' = S1 + S2 \cdot 2 + S3 \cdot 2^2 + S4 \cdot 2^3 \quad (2)$$

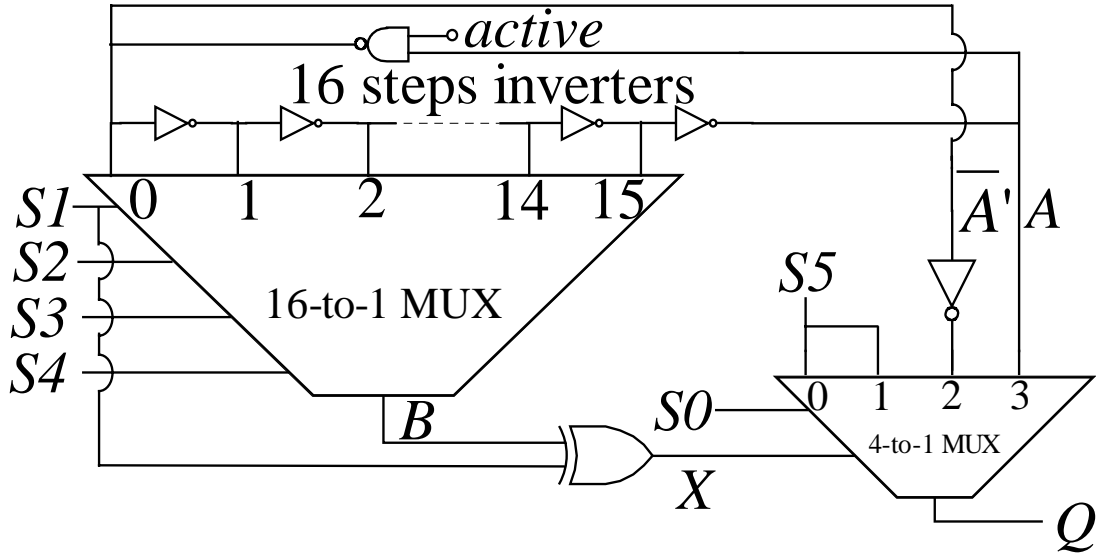


图 4.4 PWM 发生器

振荡环中反相器的延时应该尽量设计成与非门延时的两倍。不过即使不够准确，也只是会使输出电压的线形度稍稍变差，不会影响到整个功能。又因为其他单元的传输延时（产生 \overline{A} 的反相器，16 选 1 MUX，4 选 1 MUX 和异或门）要比振荡环总延时的 5% 还要小，所以在以后的讨论中就被忽略了。振荡环产生的信号占空比是 0.5，周期是 T_R 。在一个周期内，信号要经过 16 个反相器和与非门两次，振荡环的周期表示如下：

$$T_R = (16 \cdot 2d + d) \cdot 2 = 66d \quad (3)$$

这里 $2d$ 代表反相器的延时。根据 $S5$ 和 $S0$ ，在图 4.5 中给出了 Q 的详细时序描述，表 4.1 是 4 选 1 MUX 的真值表。 T_H 是 Q 的脉宽。

$$S5 = 0$$

$$S0=0 \quad T_H = S' \cdot 2d$$

$$S0=1 \quad T_H = d + S' \cdot 2d$$

$$S5=1$$

$$S0=0 \quad T_H = S' \cdot 2d + \frac{T_R}{2}$$

$$S0=1 \quad T_H = d + S' \cdot 2d + \frac{T_R}{2}$$

所以 T_H 可由下式给出：

$$T_H = S0 \cdot d + S' \cdot 2d + S5 \cdot \frac{T_R}{2}$$

再由（1）-（3）得出方程：

$$\begin{aligned} T_H &= (S0 + S1 \cdot 2 + S2 \cdot 4 + S3 \cdot 8 + S4 \cdot 16 + S5 \cdot 32 + S5) \cdot d \\ &= \begin{cases} N \cdot d & \text{for } N \leq 31 \\ (N+1) \cdot d & \text{for } N \geq 32 \end{cases} \end{aligned} \quad (4)$$

由方程(3)(4)，占空比 D 表示为：

$$\begin{aligned} D &= \frac{T_H}{T_R} \\ &= \begin{cases} \frac{N}{66} & \text{for } N \leq 31 \\ \frac{N+1}{66} & \text{for } N \geq 32 \end{cases} \end{aligned} \quad (5)$$

X	$S0$	Q
0	0	$S5$
0	1	$S5$
1	0	\overline{A}
1	1	A

表 4.1 4 选 1 MUX 真值表

PWM 得到的输出有 64 个量化单位的范围,是振荡环中反相器数目的 4 倍,因为控制器的功耗主要消耗在振荡环路的翻转上,占大部分面积的主要是多选一 MUX 和振荡环所以实现了较小的面积和功耗。

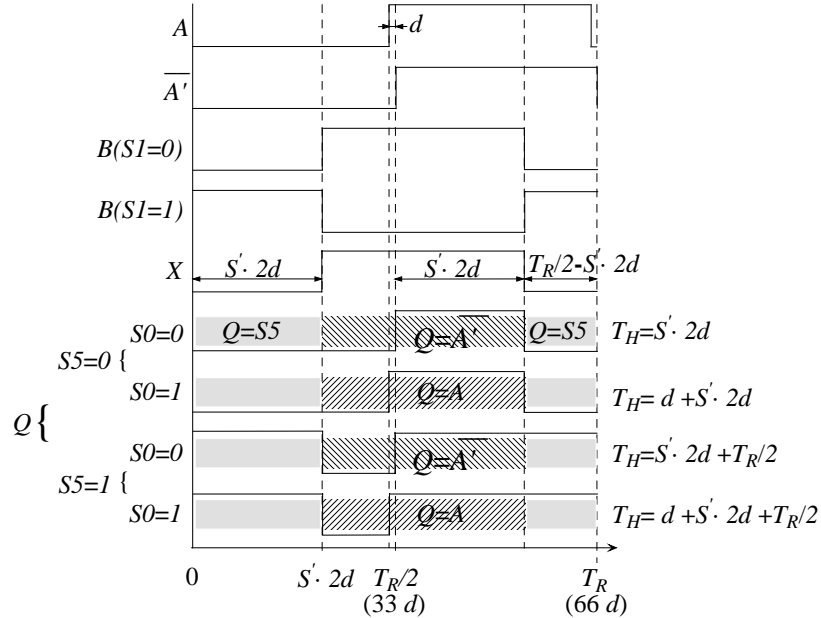


图 4.5 时序图

对图 4.4 结构除反相器环外用 SMIC.18 库标准单元实现,并在 Nanosim 中仿真,其中振荡环中反相器的延迟应该根据 PWM 所需要的振荡频率进行相应

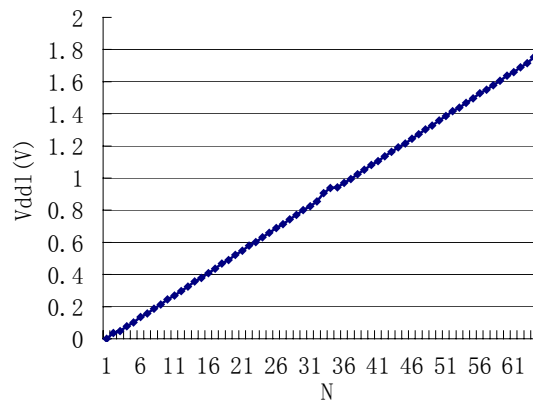


图 4.6 电压转换器输出电压和 N 的关系图

的调整。本电路最终设计的目标是电压转换器的振荡频率为 1MHz，所以通过调整振荡环中反相器的宽长而达到要求。另外选用外接器件 $L=4.7\mu\text{H}$ ， $C=5.5\mu\text{F}$ 。若在给定负载为 5Ω 时，任意选择 N 的值，得到相对应 V_{DDL} 的结果如图 4.6。其中 V_{DDL} 的最大纹波电压为 0.01V，过渡时间最长为 90us，根据文献^[18]的最小时间控制定律，该转换器可以保证在所需两稳定电压之间的转换时间大于 100us 的要求下是稳定的，其纹波电压也符合要求。转换器的输出电流最大可达 360mA，是文献^[19]所报道最大输出电流的 3 倍，有较大幅度的提高。另外该转换器在不同 N 条件下的平均功耗为 30mW，适于集成在待控芯片的功率消耗比较大的 DVS 系统中。

2. 电平转换器

虽然结果的线性度较好，但由于开关管导通时源漏等效电阻的影响，会在电源电压为 1.8V 时，电压转换器输出的最高电压不能完全达到 1.8V，而是要比 1.8V 略低，比如上面仿真的结果就为 1.775V。为了使最高电压达到 1.8V，在实际电路的实现中，选择了 3.3V 的电源供电。当然在仿真时，开关管选择的模型参数也需要是相对应 3.3V 的。由于 PWM 发生器输出的方波的峰值电压为

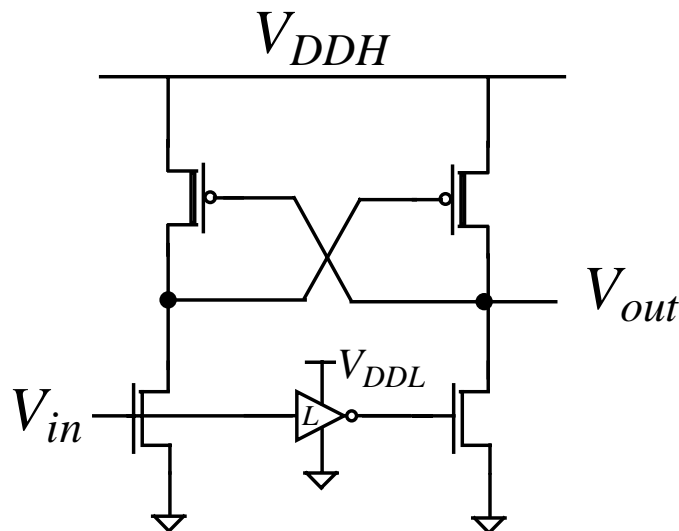


图 4.7 电位转换电路

1.8V，所以不能直接用来控制开关管，而是要经过一个由低到高的电平转换，该电路如图 4.7。该电平转换器能够较好地完成电压的转换，同时功耗也在容忍的范围内。 V_{in} 是需要转换的低电平， V_{out} 是转换后的电压，电源 V_{DDH} 是高电压，其中反相器 L 是由低电压电源 V_{DDL} 供电。

二、速度检测器

1. 速度检测器的电路图

速度检测器的工作原理就是：检测信号经过三条路径后的时序关系，来给出判断结果，如图 4.8。

三条路径分别为：

- 1) 待控芯片关键路径的复制“CPR”。
- 2) 同上的复制“CPR”，再加上一个 10% 左右的延时余量，称为“CPR+”。
- 3) 直接连接两个寄存器的路径，称为参考路径“REF”。

因为即使在很低的 V_{DDL} 下，直接相连的路径总能在周期 f_{ext} 内正确地传输测试数据，所以可以把他作为参照路径。如果其他路径的延时比所给定周期 f_{ext} 时间长时，即不能满足时序上的要求。则路径末端的触发器，不能够在规定的时刻采到所需要的数据。各个路径的输出通过与直接相连路径 REF 的输出进行对比，就可以推断出工作在电压 V_{DDL} 下的芯片是否能正确地工作在周期 f_{ext} 下。当 V_{DDL} 过低时，“CPR”和“CPR+”两个路径的输出同时都是错误的，速度检测器应该输出+1 来提高 V_{DDL} 。当 V_{DDL} 较高时，“CPR+”路径上的延迟要短于给定频率 f_{ext} ，也就是说通过两条路径后的数据输出都是正确的，速度检测器输出-1 来降低 V_{DDL} 。当 V_{DDL} 的大小刚好能够达到下面的结果时：“CPR”路径的输出是正确的，“CPR+”的输出是错误的，也就是说电压能够使“CPR”路径上的延迟满

足频率 f_{ext} 的要求,还保证了电压仅高出所需最小电压值在一个很小的余量范围内。这时速度检测器就输出 0 值,用来维持 V_{DDL} 的大小。对于得到稳定的 V_{DDL} 电压来说,存在这么一个不能检测出的电压范围是必要的,但同时也会产生一个偏差。偏差必须尽量缩小,只不过有个前提是需要比 V_{DDL} 的最小精度要大。这是因为如果偏差值比 V_{DDL} 的精度小的话,在这个电压范围内就不会存在一个 V_{DDL} 的电平值,会导致一个为 V_{DDL} 精度大小的纹波。6%的余量对应着 80mV 的 V_{DDL} ,比精度 50mV 要大一些,加到一起 V_{DDL} 的最大误差值为 130mV。

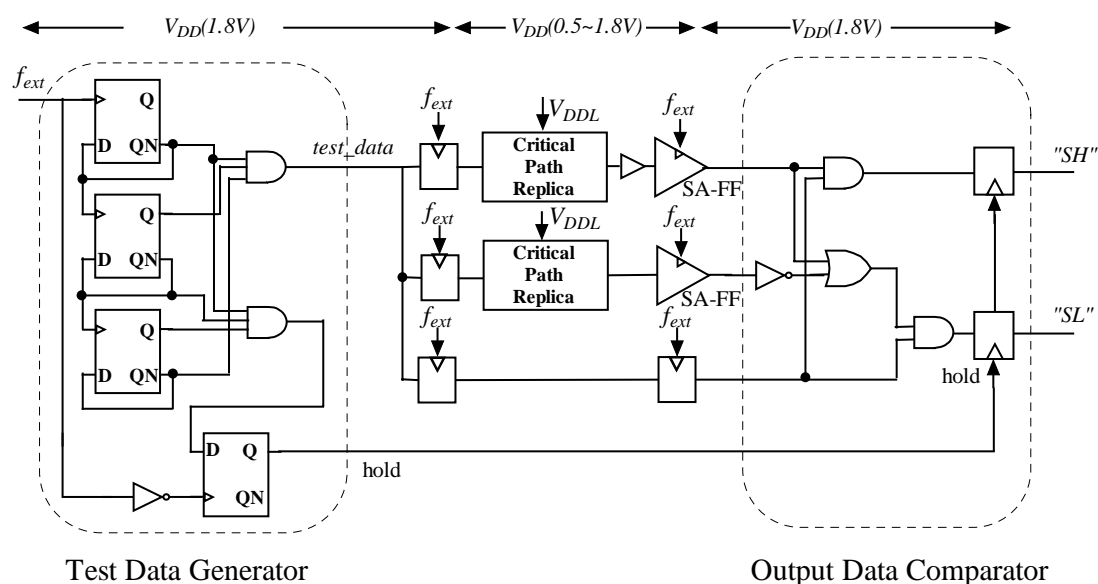


图 4.8 速度检测器电路图

2. 速度检测器的时序图

时序图如图 4.9。由图 4.8 中测试数据发生器产生测试数据,根据外部所给频率 f_{ext} ,每八个周期在 f_{ext} 上升沿 1 处产生一个由低到高的测试信号 $test_data$,在沿 3 后的下降沿产生另外的由低到高的触发信号 $hold$ 。在 f_{ext} 上升沿 2 使三个路径前端的寄存器置位,并同时在三个路径上传输。在 f_{ext} 上升沿

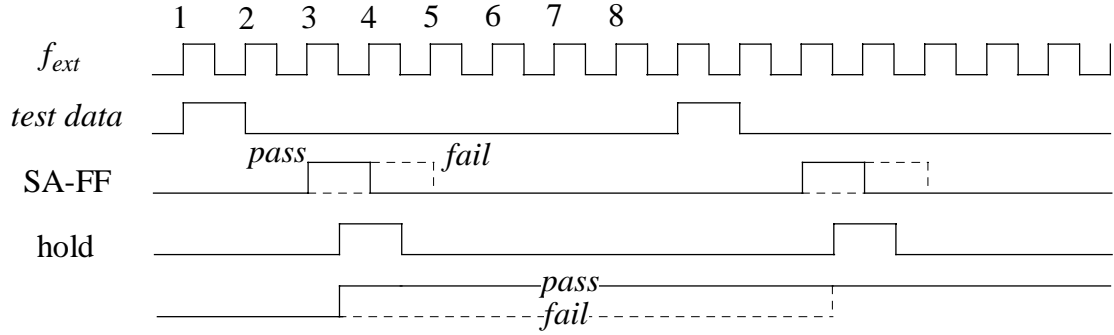


图 4.9 速度检测器时序图

3 处，由灵敏放大器 SAFF 判定该路径的电压 V_{DDL} 能否满足在一个 f_{ext} 周期内完成 $test_data$ 的传输，检测信号 $test_data$ 是否已经到达，如果已到达，SAFF 的输出为 1，否则为 0。三个路径的检测结果送入后面的输出数据比较器。比较后的结果被存在寄存器中，通过一个 $hold$ 信号来触发。输出信号只有三种 00、01 和 11，分别代表 0，+1 和 -1。

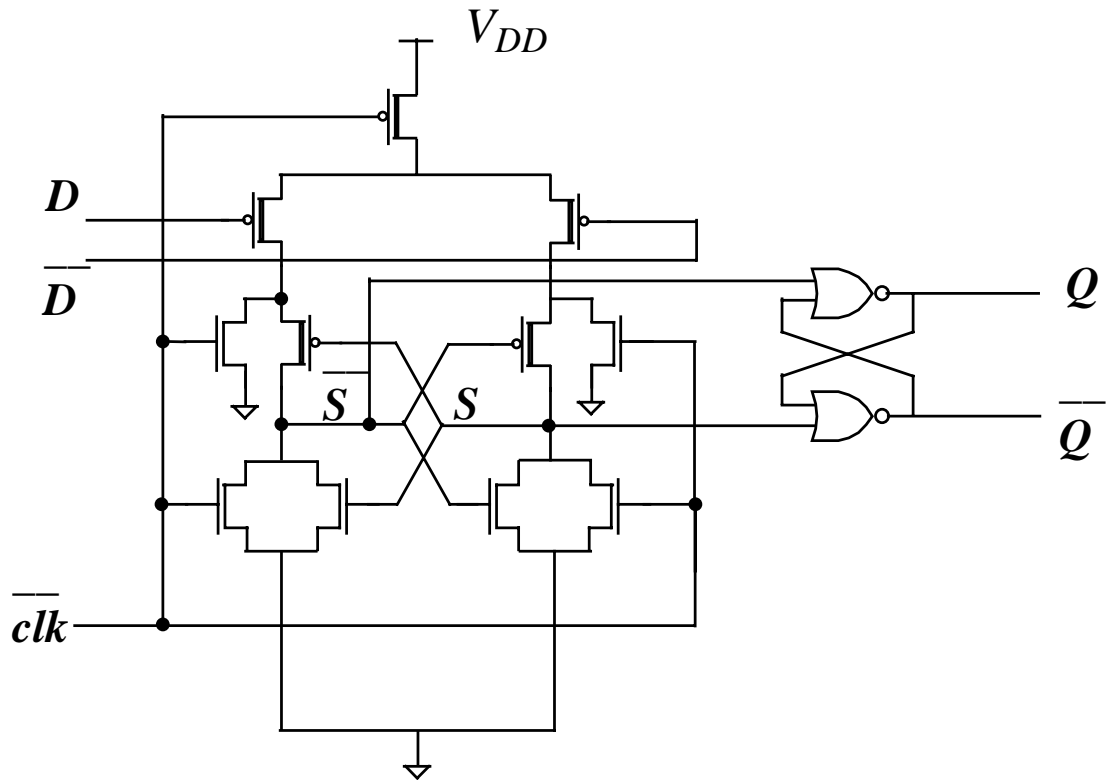


图 4.10 灵敏放大器 (SAFF) 结构图

关键路径的延时链是用标准单元库中的延时单元来实现。为了尽可能地减小功耗，延时链上单元的数目应尽可能地小，因为数目越多，相同条件下的翻转就越多，功耗就越大，所以选择的延时单元为延时最大的单元。又关键路径的复制路径是工作在动态电压 V_{DDL} 下，其要驱动的逻辑是工作在 $1.8V$ 的，故需要将信号进行电平转换到 $1.8V$ ，使用前置灵敏放大器的寄存器 SAFF 完成电平的转换和寄存，电路图如图 4.10。需要先得到通过关键路径延迟后的信号的反相信号，和延迟后的信号同时输入 SAFF，输出端信号 Q 的峰值即为高电平 V_{DD} 。

本电路可以通过改变所复制的关键路径的长度，方便的应用于有不同频率要求的待控处理器。

三、计数控制器

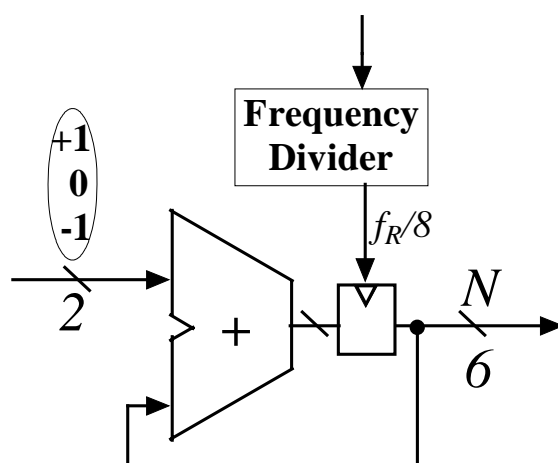


图 4.11 计数控制器

计数控制器是用来得到控制脉冲占空比信号 N 的，实现了快速并且稳定的反馈控制，如图 4.11。计数控制器的时钟频率为 f_N ，计数控制器选用 6 位全加器，速度检测器的 2 位输出信号作为 6 位输入的最低两位，其余 4 位用两位中的高位进行扩展，另一个输入端直接由输出寄存器回馈给出。

f_N 越高，响应就越快，但稳定性也越差。然而，传统的稳定性分析和补偿手段也很难应用于这一系统，主要是因为以下几个原因：在速度检测器里，电

路的速度是 V_{DDL} 的非线性函数；其输出+1 或-1 与 V_{DDL} 的误差幅度无关；除了低通滤波部分用的是模拟电路，这个控制的大部分都是用数字电路来实现的。基于上面的分析， f_N 的选择是根据多次实验的结果得到的一个经验数值，在不同的计数频率下，所输出的动态电压的波形如图 4.12 所示。在这里选用的是 1M 的 8 分频，即 0.125MHz。

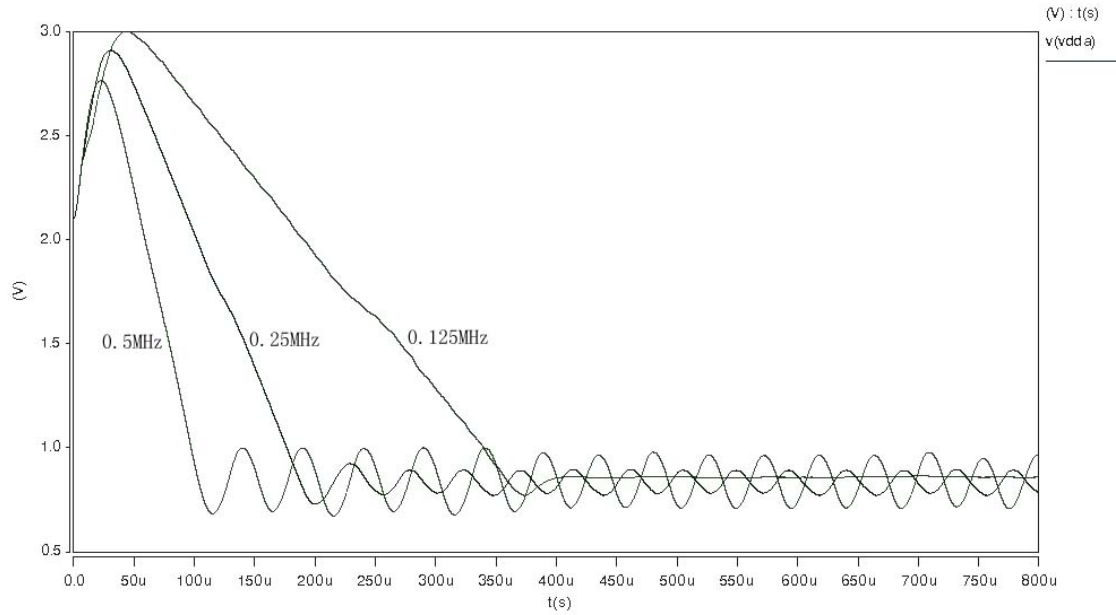


图 4.12 动态电压波形图

四、附加控制电路

因为开关管是用 3.3V 电源供电，最后的转换器要应用于 1.8V 的待控芯片，在 0.5V 以下，由于数字电路无法工作，所以还要附加一些控制逻辑来控制电压转换器的输出最高电压为 1.8V 和最低电压 0.5V。这时相对应的计数器的值应该为 36 和 10。假设速度检测器的输出高位为 SH，低位为 SL，在 N=36 时，A=1；N=10 时，B=1，根据真值表 4.2 和 4.3，可以得到：

$$H = SH \cdot SL \cdot \overline{B}$$

$$L = SH \cdot SL \cdot \overline{B} + \overline{SH} \cdot SL \cdot \overline{A}$$

根据以上逻辑关系，将电路实现后的 H 和 L 作为全加器的输入，就可以实现对输出电压的限制。

表 4.2 加法器输入高位 H

SH SL $\begin{matrix} =0.5V & =1.8V \end{matrix}$	00	01	11	10
00	0	0	X	0
01	0	0	X	0
11	1	1	X	0
10	X	X	X	X

表 4.2 加法器输入最低位 L

SH SL $\begin{matrix} =0.5V & =1.8V \end{matrix}$	00	01	11	10
00	0	0	X	0
01	1	0	X	1
11	1	1	X	0
10	X	X	X	X

第三节 电路的仿真与低功耗验证

本电路用 SMIC.18um 工艺实现,为了对本电路所实现的效果进行直观地分析,采用了一个简单的乘法器作为待控单元组成一个 DVS 系统,如图 4.13。

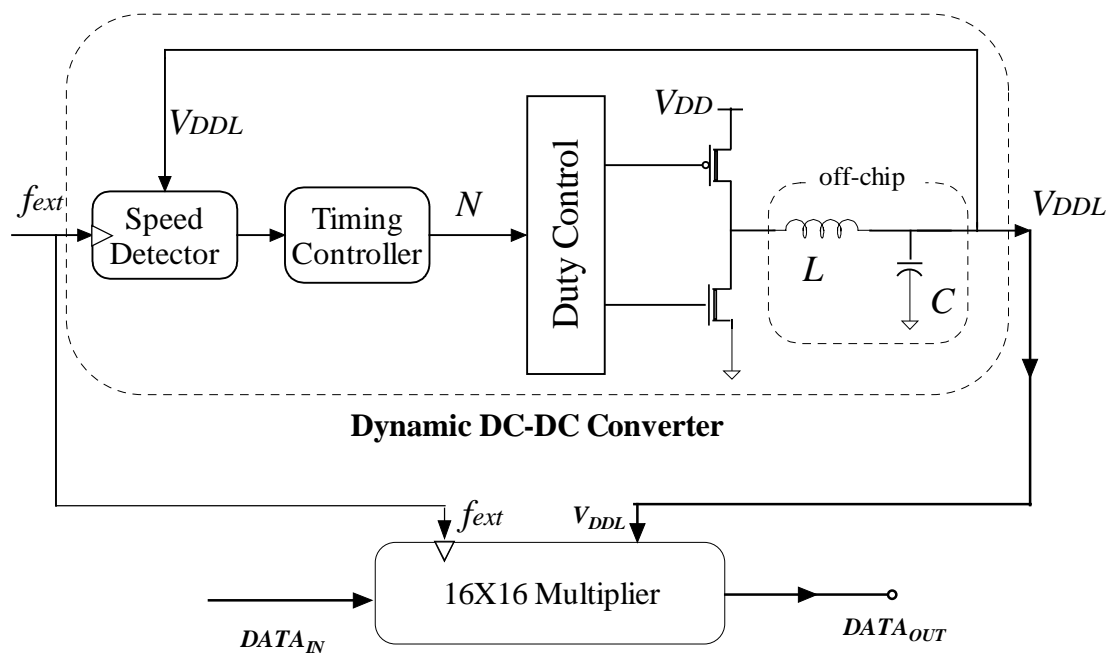


图 4.13 DVS 系统测试图

该乘法器用 Module Compiler 实现，为 16X16 的无符号乘法，其中加法器选用逐位进位加法器（ripple adder），输入输出都使用寄存器，最长路径的延迟为 5.5ns。

使用相同的测试激励，分别在不同周期 6ns、9ns、12ns、18ns 和 24ns，对应频率 167MHz、111MHz、83.3MHz、55.6MHz、41.7MHz 下用 Nanosim 对其进行仿真，得到的结果如下：

表 4.4 各频率下的结果

Frequency	V_{DDL}	Dynamic dc-dc converter power	Multiplier	
			$V_{DDL}(DVS)$	V_{DD}
167MHz	1.829V	34.296mW	13.164mW	12.677mW
111MHz	1.292V	23.893mW	4.086mW	8.453mW
83.3MHz	1.079V	23.64mW	2.541mW	6.34mW
55.6MHz	0.967V	22.782mW	1.125mW	4.225mW
41.7MHz	0.858V	21.446mW	0.725mW	3.169mW

不同频率情况下，乘法器仍能保证正常工作，并且对比没有对乘法器进行动态电压缩放的数据，可以看出动态电压缩放后功耗得到了很大的改善。不过这都仅仅是考虑乘法器本身的功耗大小，没有将转换器考虑在内。因为本论文

表 4.5 频率切换过程中的跟踪时间

Frequency	Track time
167MHz to 111MHz	130us
167MHz to 83.3MHz	150us
167MHz to 55.6MHz	290us
167MHz to 41.7MHz	300us
41.7MHz to 167MHz	140us

的出发点是利用乘法器作为一个验证电路,证明了 DVS 的低功耗可实现性。前面的章节中也已经提到,DVS 的低功耗优势在于突发吞吐量工作模式的处理器。

在进行两个频率之间的切换过程中,两个稳定电压之间的跟踪时间如表 4.5

图 4.14、4.15、4.16 示出频率由 167MHz 切换到 111MHz, 167MHz 切换到 83.3MHz,和 41.7MHz 切换到 167MHz 时动态电压转换器输出端电压的波形图。

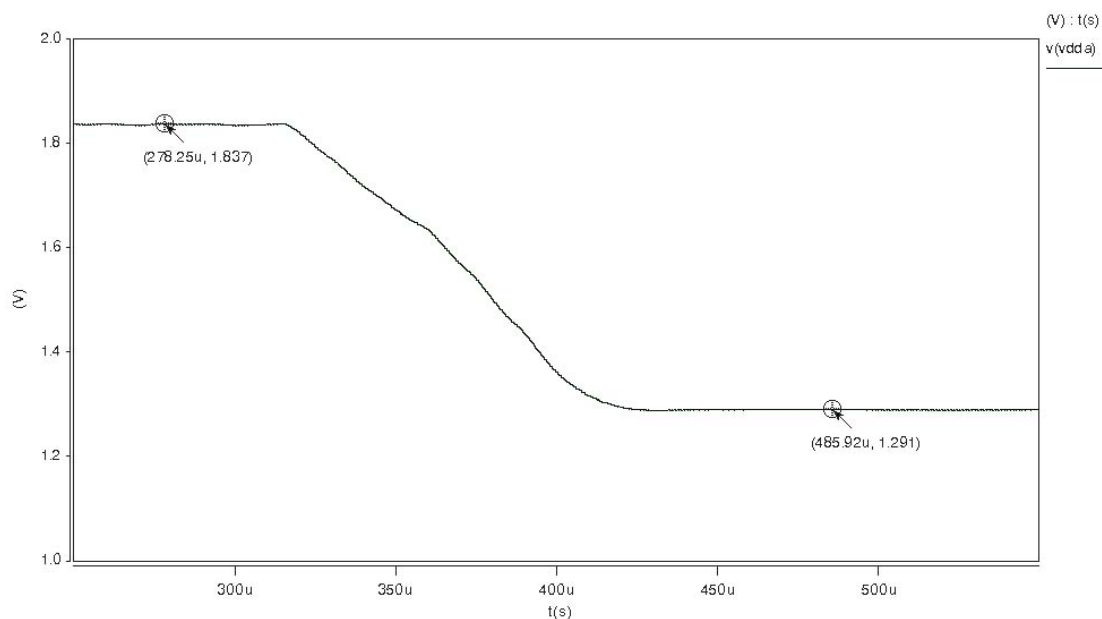


图 4.14 频率由 167MHz 切换到 111MHz 的电压波形图

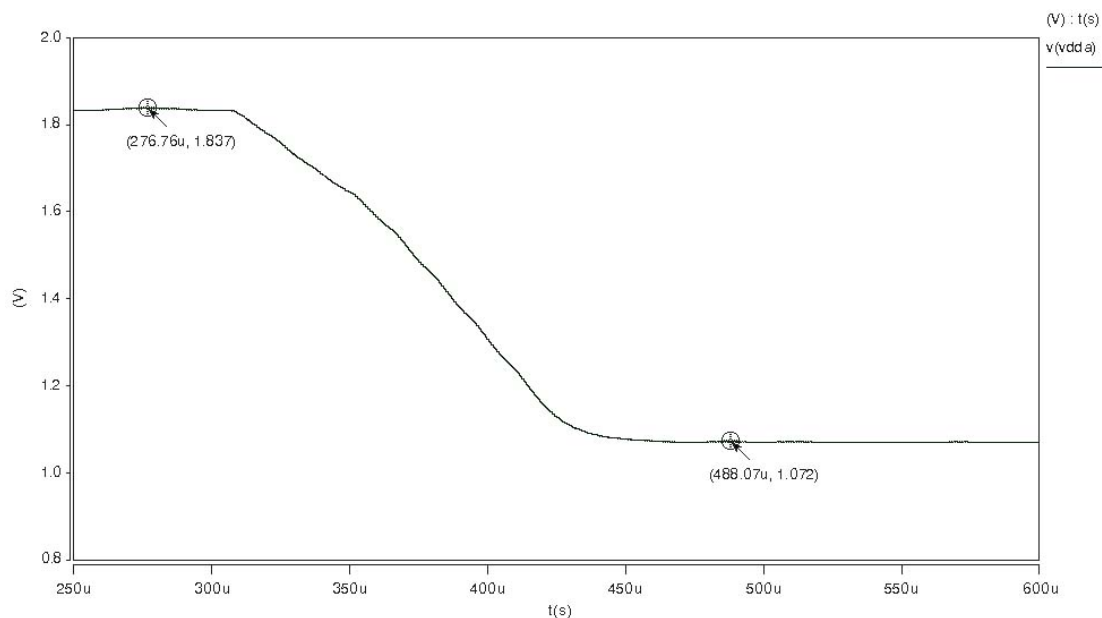


图 4.15 频率由 167MHz 切换到 83.3MHz 的电压波形图

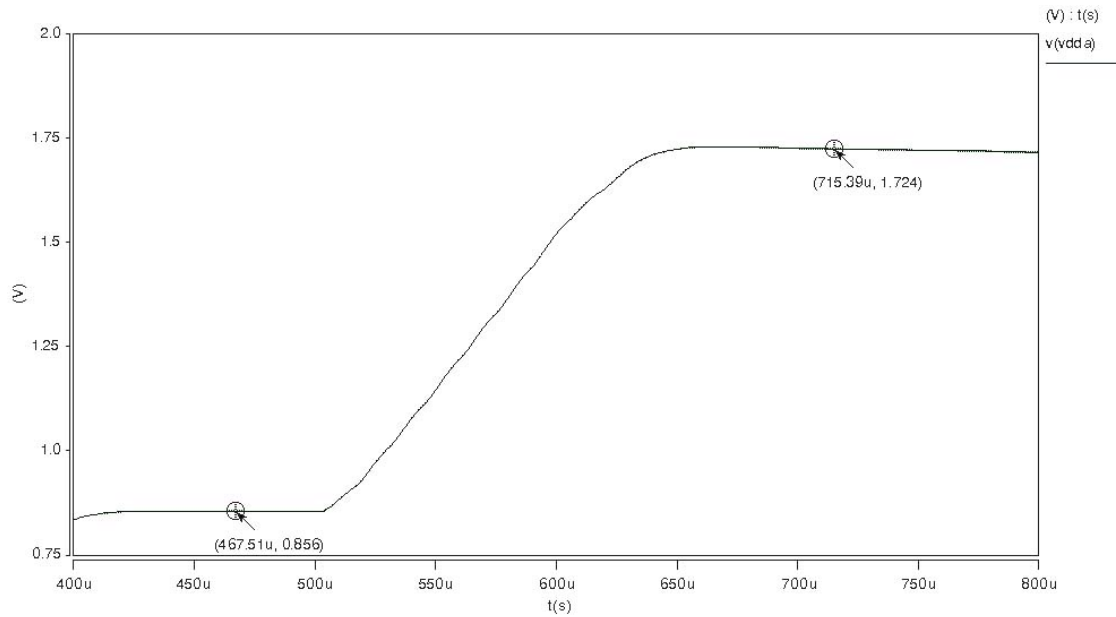


图 4.16 频率由 41.7MHz 切换到 167MHz 的电压波形图

由于本DVS电路的跟踪时间达到百us左右,相对来说切换速度是比较慢的,所以跟踪能量是可以忽略的,这和实际的仿真的 1mW-2mW情况相符^[20]。

由结果可以得到,本DVS电路能够较好地跟踪不同频率切换下的电压输出转换,并且达到了低功耗的效果。根据最小跟踪时间法则^[18],能够保证在切换时间大于 0.2ms的系统的稳定性。

第五章 总结

本文系统地介绍了各个层次上的低功耗设计方法，深入地研究了超大规模集成电路设计中的功耗预估和一种结构层次上的低功耗设计方法——动态电压缩放技术。

本论文取得了以下主要成果：

1. 对一个 450 万门的超大规模芯片在 RTL 级和门级上进行了功耗分析，作为设计初期可行性分析和后端布局布线的参考，保证了一次投片的成功，并和流片后测试得到的结果有着很好的吻合。
2. 确定了超大规模集成电路设计中各个层次上的功耗预估方法，并总结了一整套低功耗设计的设计流程。
3. 系统、细致地研究了数字 CMOS 电路设计中各层次上的低功耗实现方法，并着眼于一种结构级的低功耗技术——动态电压缩放技术，详细研究了该技术的工作原理、适用范围和性能指标。
4. 实现了一款具体的动态电压缩放电路，通过仿真表明该电路可以实现动态电压缩放的功能：根据给定的不同频率，输出相应的满足要求的最小电压值，并能够跟踪频率的变化。
5. 实现了一个可控占空比的数字 PWM 控制电路，该电路功耗低，所占面积小，适于用做动态电压转换器的集成。
6. 将一个 16X16 的乘法器和实现的动态电压缩放电路一起组成一个动态电压缩放系统，进行仿真，对比使用动态电压缩放电路前后乘法器的功耗大小，验证了使用该动态电压缩放电路的低功耗效果。

国外在低功耗技术上处于领先地位的几家公司，正在致力于开发基于动态电压缩放的一整套设计流程。因此，本论文所研究的动态电压缩放有着良好的应用前景。

参考文献

- 1 Mary Jane Irwin, Low Power Design for System on a chip design, In: ASIC/SOC Conference, 1999. Proceedings. Twelfth Annual IEEE International, 1999, 422-422.
- 2 L.Benini, M.Favalli, and B.Ricco, Analysis of Hazard Contributions to Power Dissipation in CMOS IC's, International Workshop on Low-Power Design, 27-32, 1994.
- 3 Small C. Shrinking devices put the squeeze on system packaging. EDN, 39(4):41-46,1994.
- 4 A. Stratakos. "High-Efficiency, Low-Voltage dc-dc Conversion for Portable Applications", Ph.D thesis, Univ.California, Berkeley, CA, 1999
- 5 I. Hong, M. Potkonjak, M. Srivastava, On-Line Scheduling of Hard Real-Time Tasks on Variable Voltage Processors. International Conference on Computer-Aided Design, pp. 653-656, Nov. 1998.
- 6 T. Okuma, T. Ishihara, H. Yasuura, Real-Time Task Scheduling for a Variable Voltage Processor. DAC, pp. 176-181, June 1998.
- 7 Intel. Pulished in IC Insights Inc.2003 Technology Trends.
- 8 乔长阁, 孔天明, 夏阳等. 最小面积电源和地线网络的设计. 电子学报, 1998, 26 (8): 126-128
- 9 A. Chandrakasan, V. Gutnik, T. Xanthopoulos, Data Driven Signal Processing: An Approach for Energy Efficient Computing. IPED, IEEE, 1996.
- 10 T.D. Burd, "Energy-Efficient Processor System Design", Ph'D thesis, University of California.
- 11 F. Gruian, Hard Real-Time Scheduling for Low-Energy Using Stochastic Data and DVS Processors, ISLPED'01, pp. 46-51, August 6-7, 2001.
- 12 W. Kim, D. Shin, Performance Comparison of Dynamic Voltage Scaling Algorithms for Hard Real-Time Systems, RTAS'02, IEEE.

- 13 A. Burstein, Speech Recognition for Portable Multimedia Terminals, Ph.D. Thesis, University of California, Berkeley, Document No. UCB/ERL M97/14, 1997.
- 14 A. Soto, P. Alou, J.A. Cobos, J. Uceda, The Future DC-DC Converter as an Enable of Low Energy Consumption Systems with Dynamic voltage scaling, in IEEE Industrial Electronics Conference, IECON'02.
- 15 www.synopsys.com/sps
- 16 张占松, “高频开关稳压电源”, 广东科技出版社, 1990.
- 17 www.arm.com
- 18 A. Soto, A. Castro, P. Alou, Analysis of the Buck Converter for Scaling the Supply Voltage of Digital Circuits, IEEE Journal of Solid-State Circuits, pp. 711-717,
- 19 T. D. Burd, T. A. Pering, A. J. Stratakos, R.W. Brodersen, “A Dynamic Voltage Scaled Microprocessor System,” IEEE J. Solid-State Circuits, vol.35 (11), Nov. 2000, pp. 1571-1579.
- 20 W. Namgoong, M. Yu, and T. Meng, “A High-Efficiency Variable-Voltage CMOS Dynamic dc-dc Switching Regulator”, IEEE International Solid-State Circuits Conference, pages 380-381, Apr.1997.

发表文章目录

1. 刘海南，周玉梅，“基于 DVS 的动态电压转换器设计”，电子器件，已录用
2. 刘海南，周玉梅，“可控占空比的数字 PWM 发生电路”，集成电路应用，已录用

致 谢

时光如梭，三年的硕士研究生生活即将结束。回首这段充实、紧张而又颇感短暂的经历，我对所有给予过我关心与帮助的人充满了感激之情。

深深地感谢我的导师周玉梅研究员。本论文的选题、写作以及最终完成都得到了周老师的悉心指导。她以渊博的学识和严谨的治学态度令我耳濡目染，受益终生。在生活上，周老师给予我无微不至的关怀，在为人处事的道理上，也从她那里得到很多启迪。在此表示由衷的感激和诚挚的谢意。

感谢叶青研究员在学习期间给予的诸多指导，以及对我的关心和帮助。

感谢李荣敏老师、王豫回老师在学习和日常生活中给予我的热诚关怀和帮助。

感谢黑勇副研究员、吴斌、蒋见花、张月、张弛、霍津哲、高健、张锋、赵坤、张岩、韩飞在我做论文期间，给予的创造性建议和真诚帮助。

感谢微电子所 ASIC 实验室提供的优秀科研和实验环境。感谢所有的师兄兄弟们，大家朝夕相处，关系融洽，形成了很好的学术氛围，在生活上也能互帮互助。在此向董庆祥、王晶、陈博、高大明、满家汉、牛旭花等表示衷心的感谢。

感谢赵冰、汤仙明、邵刚，作为舍友，大家生活得犹如一家人，简陋的宿舍成为了紧张学习之余得以放松的港湾，感谢他们给予我的友情，使得平淡的生活变得如此多姿多彩。

感谢微电子所研究生部的边林芬老师和崔京老师对我生活、学习上的悉心关怀和帮助。

最后，让我永远感激、铭记在心的是我的父母和其他亲人给予我的无私奉献和鼓励，感谢他们为我所付出的一切！