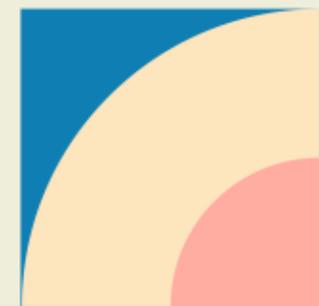




# 一种低功耗高性能并行乘法器的设计与实现

汇报人：

2024-01-15





# 目录

- 引言
- 并行乘法器基本原理与结构
- 低功耗高性能并行乘法器设计
- 实验结果与分析
- 与其他乘法器性能对比
- 结论与展望
- 参考文献
- 附录

01

引言





# 研究背景和意义



01

## 数字化时代需求

随着数字化时代的到来，高性能计算和低功耗设计在各个领域的需求日益增长。

02

## 乘法器的重要性

乘法器作为数字信号处理中的关键模块，其性能直接影响整体系统的运算速度和功耗。

03

## 并行乘法器的优势

相比于传统串行乘法器，并行乘法器具有更高的运算速度和更低的功耗，因此具有广泛的应用前景。



# 国内外研究现状及发展趋势

## 国内外研究现状

目前，国内外学者在并行乘法器设计方面已经取得了一定成果，但仍存在功耗较高、性能较低等问题。

## 发展趋势

随着集成电路技术和计算机体系结构的不断发展，未来并行乘法器将朝着更低功耗、更高性能、更小面积的方向发展。

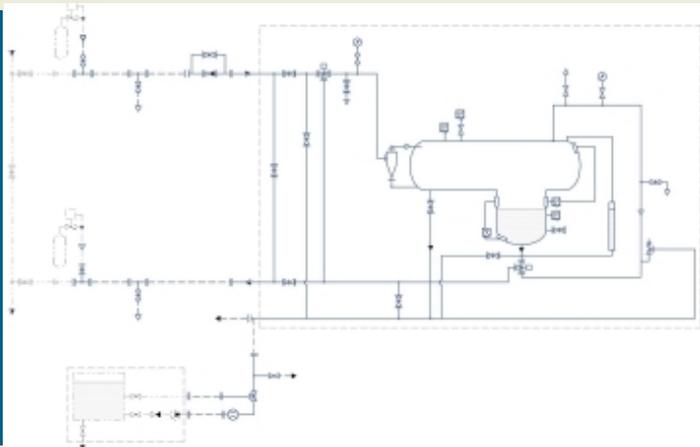




# 研究内容、目的和方法

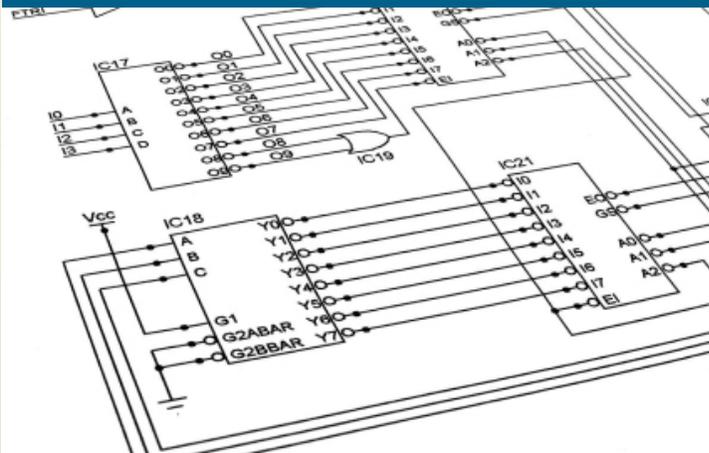
## 研究内容

本研究旨在设计一种低功耗高性能的并行乘法器，通过优化算法和电路结构，降低功耗并提高运算速度。



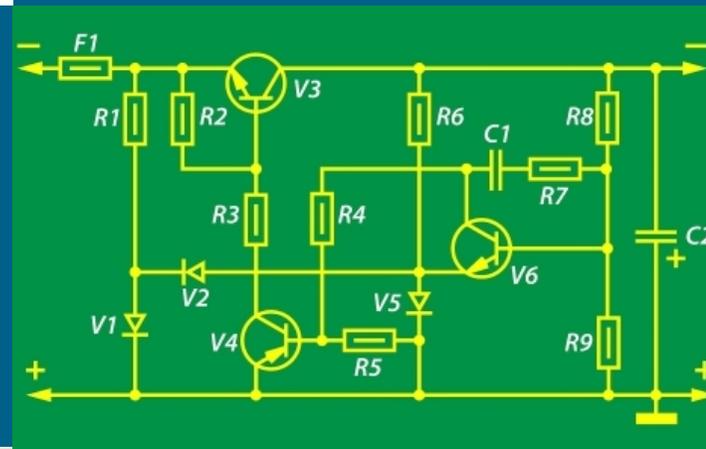
## 研究方法

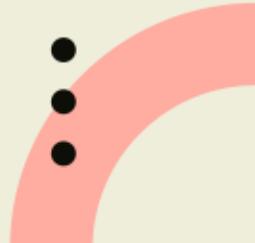
采用理论分析和实验验证相结合的方法，首先进行算法设计和优化，然后进行电路设计和仿真验证，最后进行实际测试和应用分析。



## 研究目的

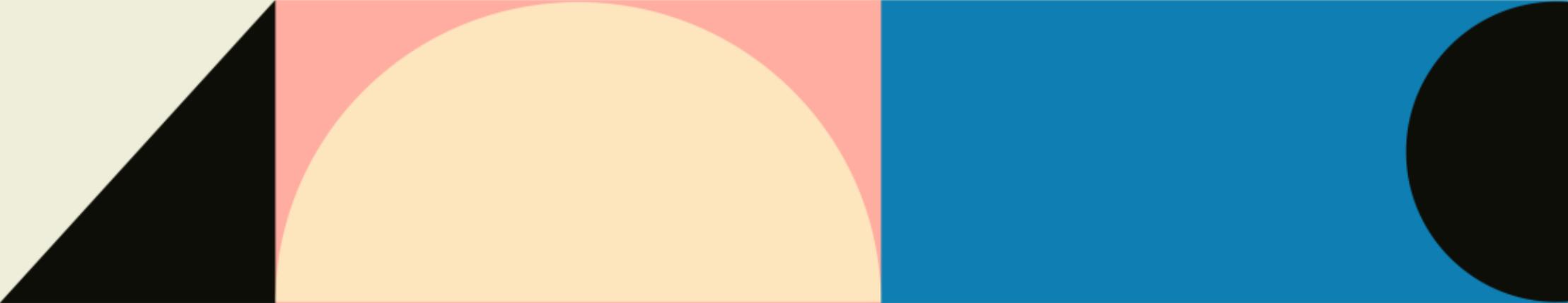
通过本研究，期望为高性能计算和低功耗设计领域提供一种高效、可靠的并行乘法器解决方案。





02

# 并行乘法器基本原理与结构





# 并行乘法器基本原理

Handwritten mathematical work on grid paper showing algebraic steps for solving a system of equations:

$$\frac{x}{2} = \frac{1}{5} + \frac{3}{10}x + c$$
$$\frac{5x + 20 - 10x}{20} = \frac{28 + 6x + 10}{20}$$
$$-10x - 6x = -20 + 28 + 10$$
$$\frac{2x}{12} = \frac{+18}{-12} - 4$$
$$x - 1 = \frac{1}{2}x + \frac{3}{2} - 5x + \frac{3}{2}x$$
$$\frac{-2}{2} = \frac{1x + 3 - 10x + 3x}{2}$$
$$x - 1x + 10x - 3x = +2 + 3$$
$$\frac{2x}{10} = \frac{+5}{+10} + \frac{1}{2}$$

## 并行计算

并行乘法器采用并行计算的方式，将大规模的乘法运算拆分成多个较小的乘法运算，并在多个计算单元上同时进行，从而提高了计算速度。

## 流水线技术

在并行乘法器的设计中，常采用流水线技术，使得乘法运算的连续进行不受单个计算单元的延迟影响，进一步提高了运算效率。



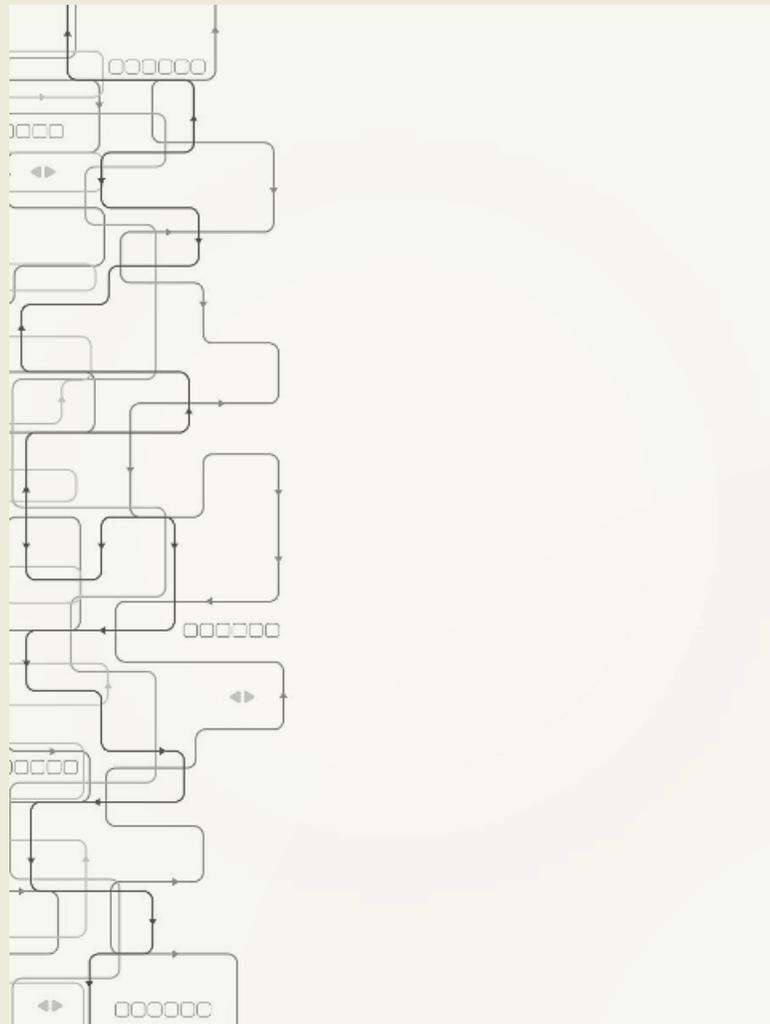
# 常见并行乘法器结构

## 阵列乘法器

阵列乘法器是一种常见的并行乘法器结构，它由多个相同的处理单元组成阵列，每个处理单元负责一部分乘法运算，通过并行计算完成整个乘法过程。

## 树形乘法器

树形乘法器采用树形结构，将乘法运算逐级分解，每级都包含多个并行计算单元。通过逐级合并计算结果，最终得到完整的乘积。





# 性能评价指标



## 吞吐量

吞吐量是衡量并行乘法器性能的重要指标之一，它表示单位时间内完成的乘法运算次数。高吞吐量意味着乘法器具有快速的计算能力。



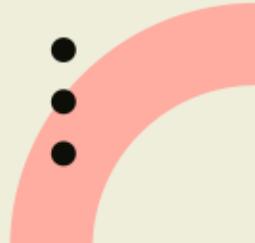
## 延迟

延迟是指从输入乘数到输出乘积所需的时间。对于实时性要求较高的应用，低延迟的并行乘法器更为适用。



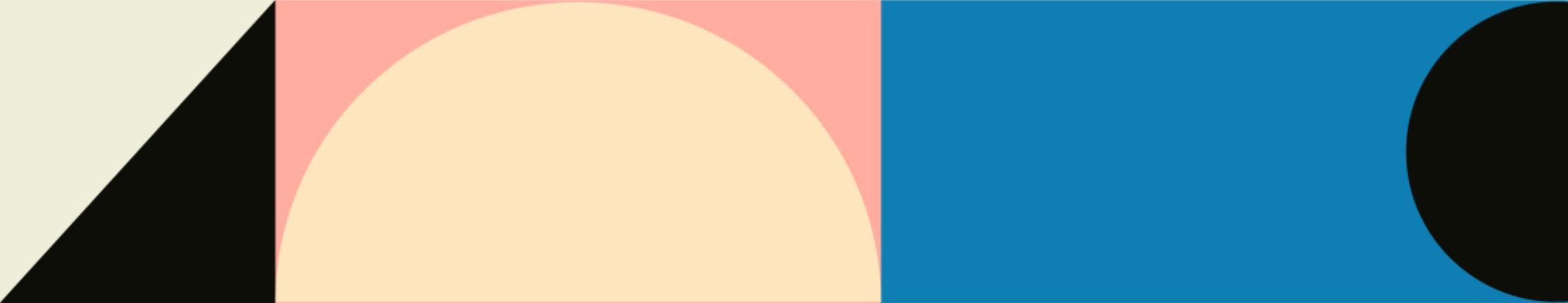
## 能耗

随着绿色计算和低功耗设计的兴起，能耗成为评价并行乘法器性能的关键因素之一。低能耗的乘法器有利于减少散热问题和延长设备使用寿命。



03

# 低功耗高性能并行乘法器设计





# 设计思路与总体架构

## ● 并行计算架构

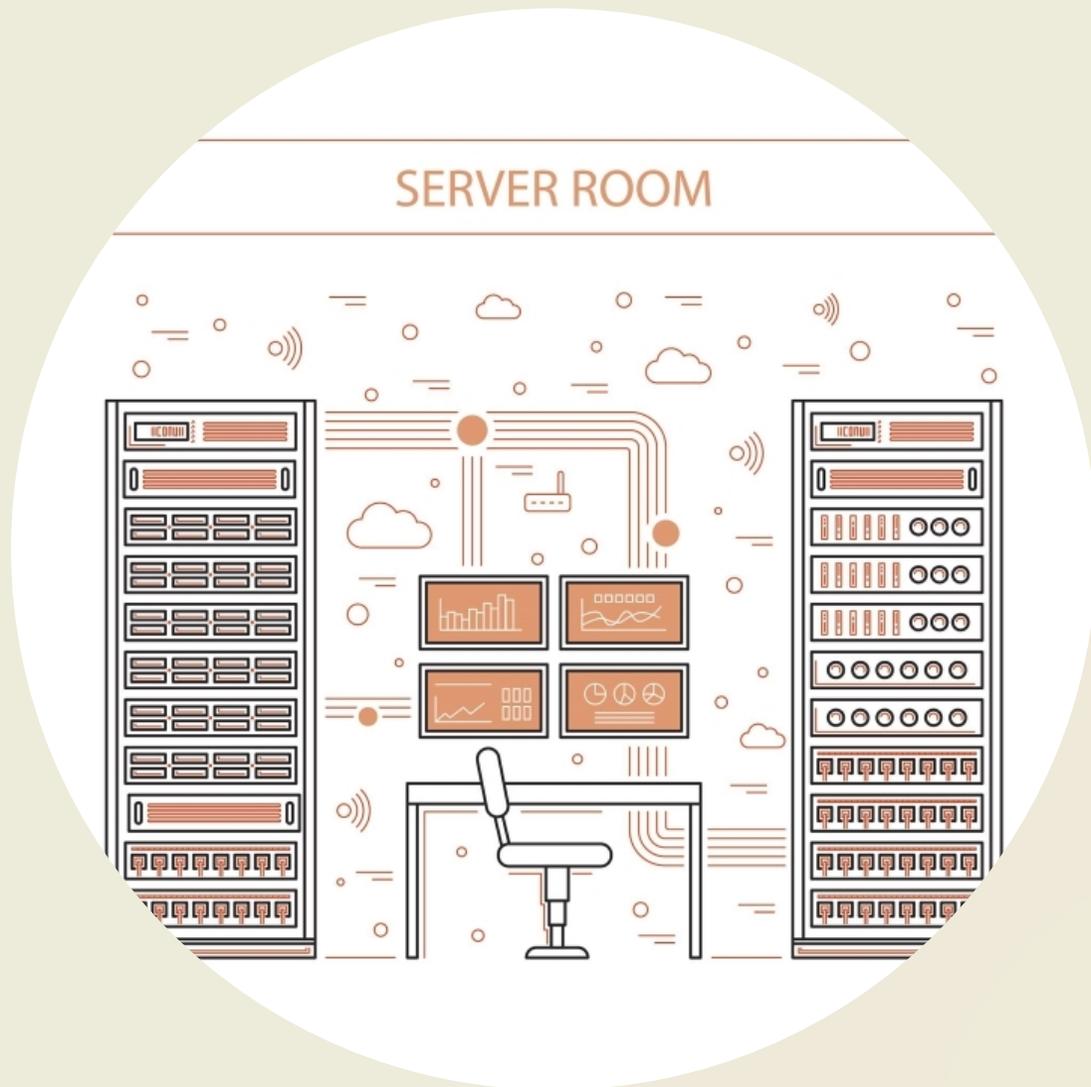
采用多个独立的乘法单元并行计算，提高乘法运算速度。

## ● 流水线设计

将乘法运算拆分为多个阶段，通过流水线方式实现连续的数据处理，提高吞吐量。

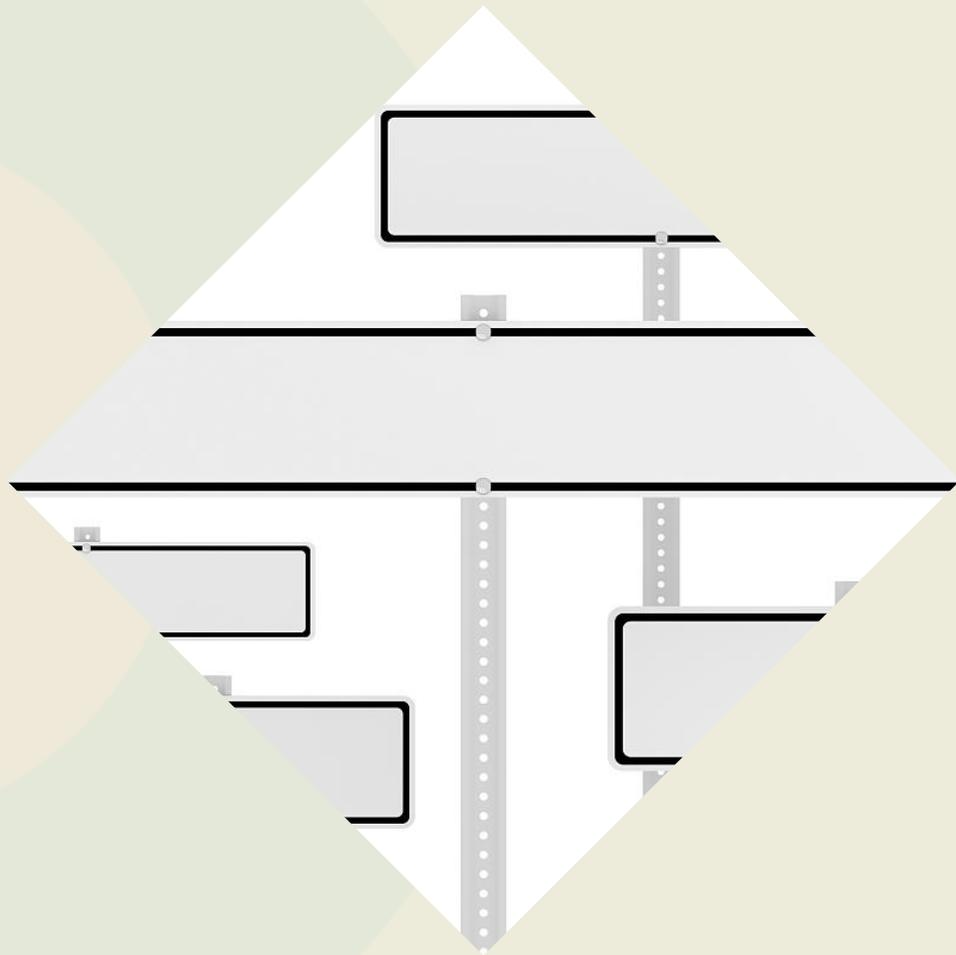
## ● 模块化设计

将乘法器划分为多个功能模块，便于实现和调试。





# 关键模块设计与实现



## 乘法单元设计

采用高性能的乘法算法，如Karatsuba算法或Schönhage-Strassen算法，提高乘法运算速度。

## 控制逻辑设计

实现乘法器的启动、停止、复位等操作，以及数据输入输出的控制。

## 数据通路设计

优化数据通路结构，减少数据传输延迟，提高乘法运算效率。



# 功耗优化策略



## 动态电压频率调整

根据乘法器负载情况动态调整工作电压和频率，降低功耗。

## 门控时钟技术

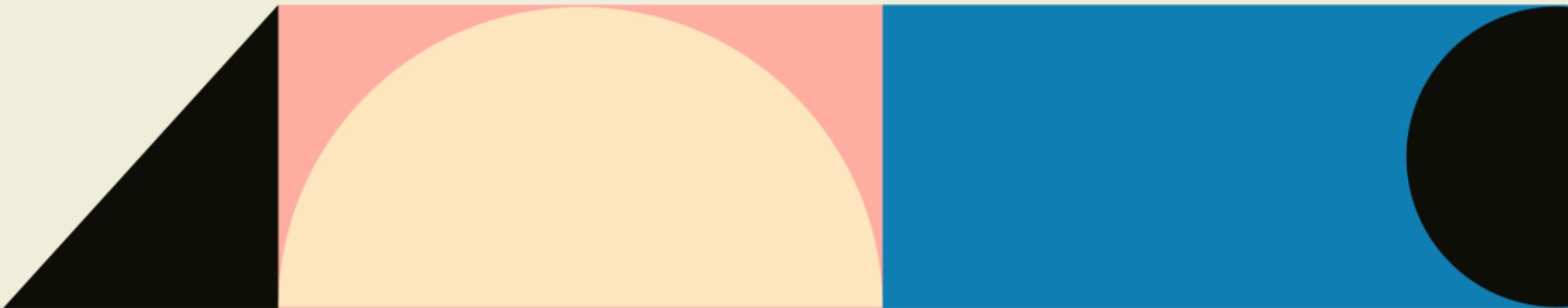
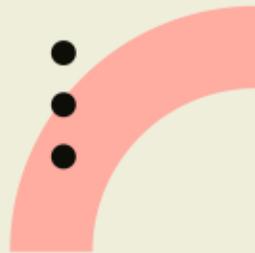
对乘法器中不活跃的部分采用门控时钟技术，减少不必要的功耗。

## 睡眠模式设计

在乘法器空闲时进入睡眠模式，降低静态功耗。

04

# 实验结果与分析





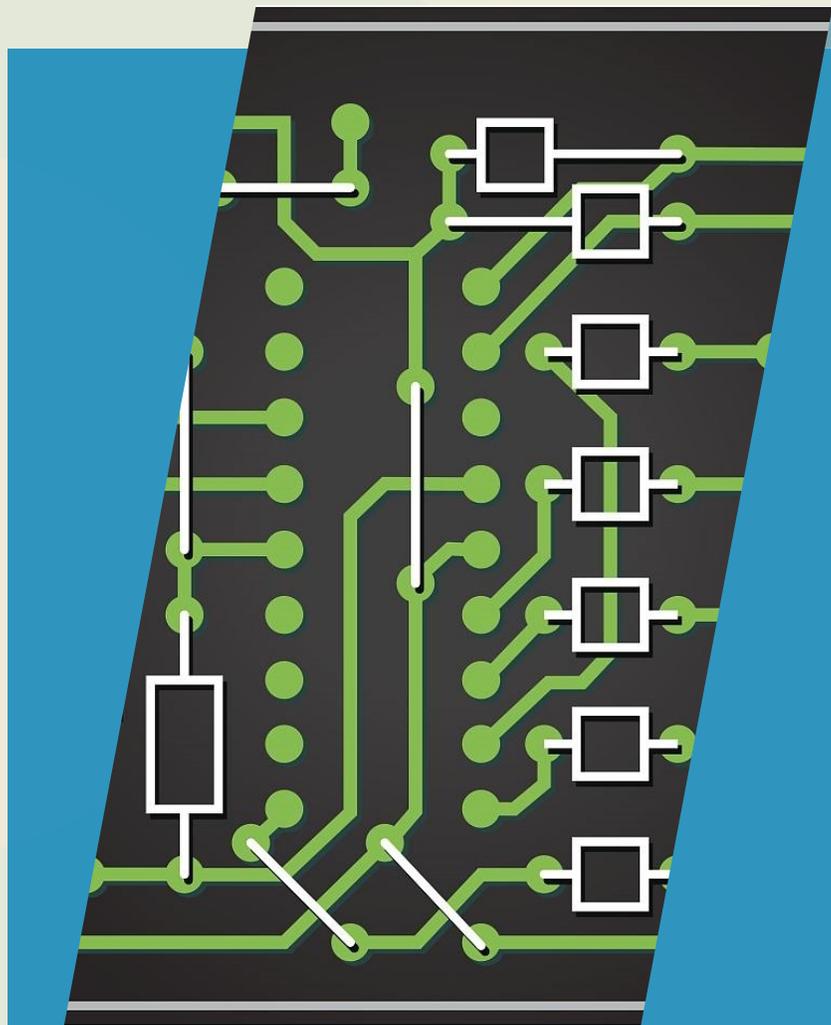
# 实验环境与测试方法

## 实验环境

本实验在高性能计算机上进行，使用专业的集成电路设计软件进行设计和仿真。

## 测试方法

采用随机生成的测试向量对乘法器进行测试，包括正常操作、边界条件和异常情况下的测试。



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：  
<https://d.book118.com/677130105051006116>