

# 计算机系统结构



群名称: 计算机系统结构-邓建-2024  
群 号: 126175420

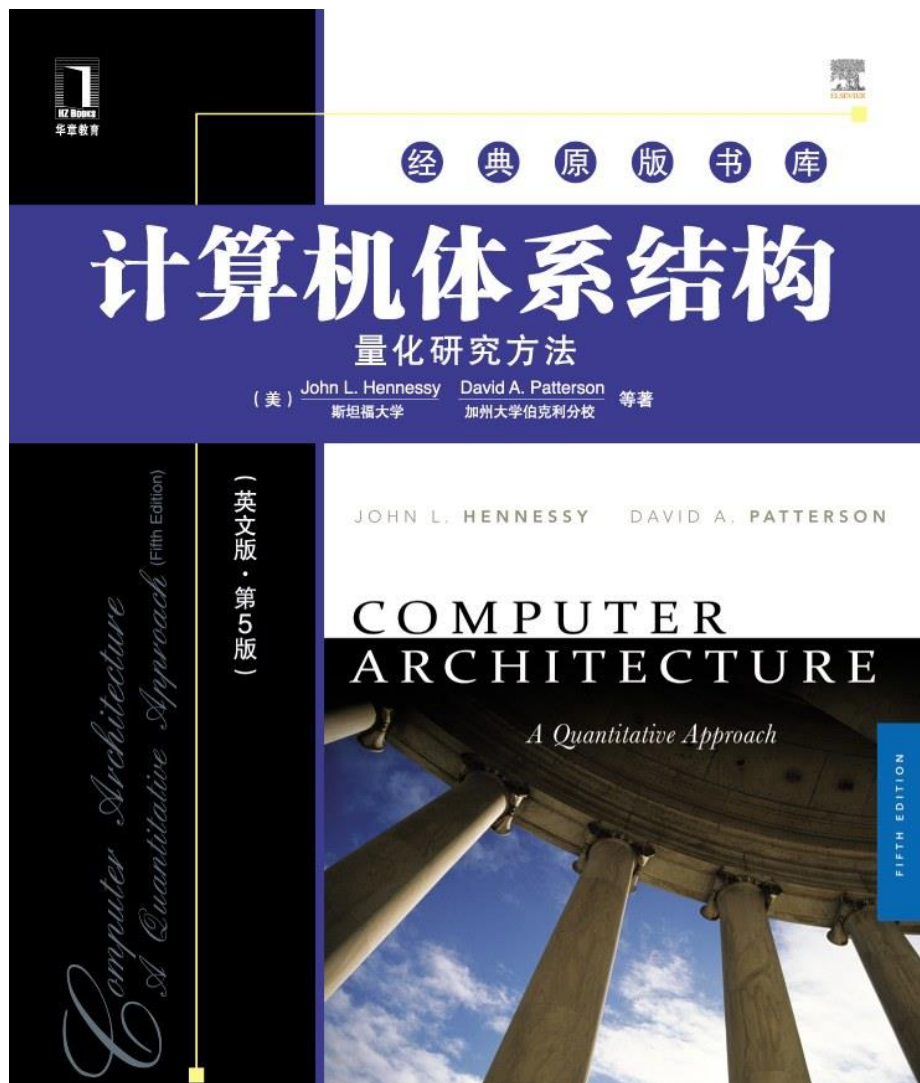
邓建

dengjian@uestc.edu.cn

18980891251@189.cn

# 教材

原书第5版

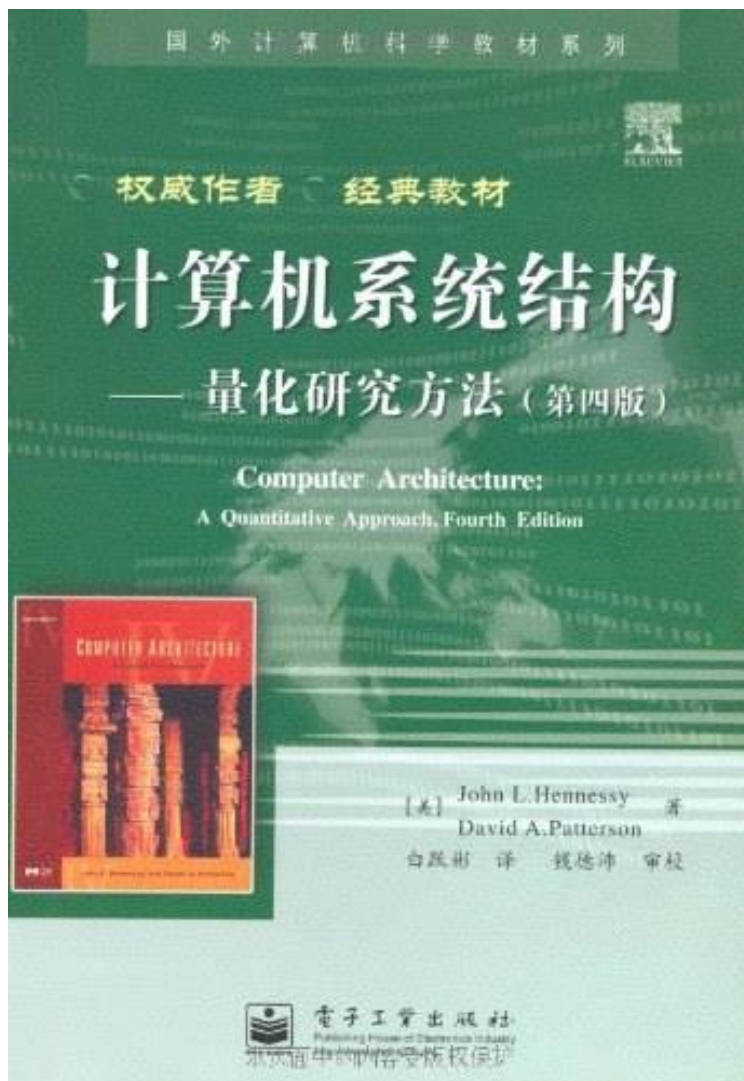


## 计算机体系结构 --量化研究方法

Computer Architecture  
--A quantitative approach

John L. Hennessy,  
David A. Patterson

# 参考书



TURING 图灵计算机科学丛书



[美] 约翰·L. 亨尼西 (John L. Hennessy) 著  
大卫·A. 帕特森 (David A. Patterson) 著

贾洪峰 译 唐忆滨 唐杉 审

## 计算机 体系结构

量化研究方法（第6版）

Computer Architecture: A Quantitative Approach

Sixth Edition



中国工信出版集团

人民邮电出版社  
POSTS & TELECOM PRESS

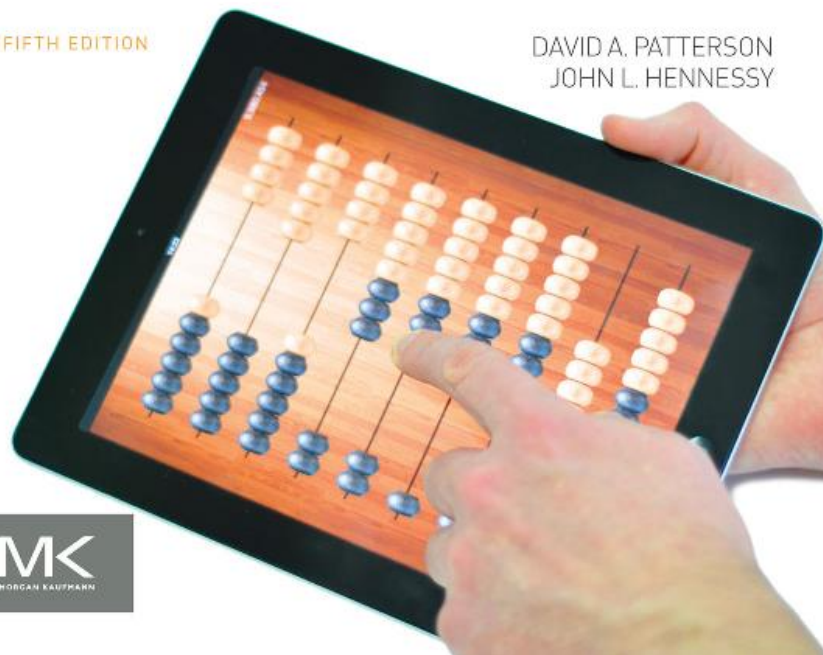
# 参考书

## COMPUTER ORGANIZATION AND DESIGN

THE HARDWARE/SOFTWARE INTERFACE

FIFTH EDITION

DAVID A. PATTERSON  
JOHN L. HENNESSY



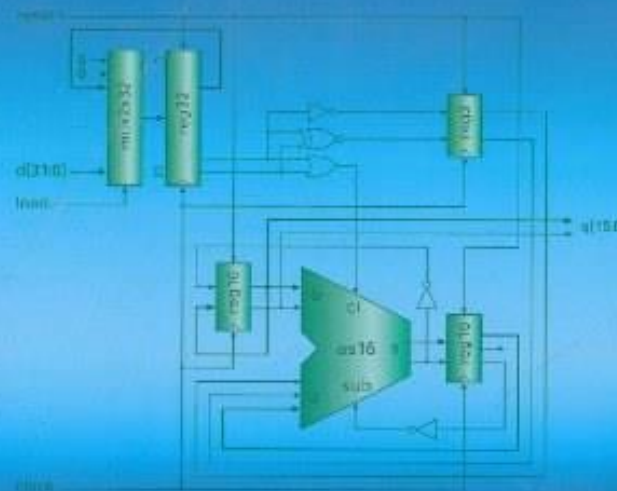
MK  
MORGAN KAUFMANN

Computer Principles and Design in Verilog HDL

## 计算机原理与设计

—— Verilog HDL 版

李亚民 著



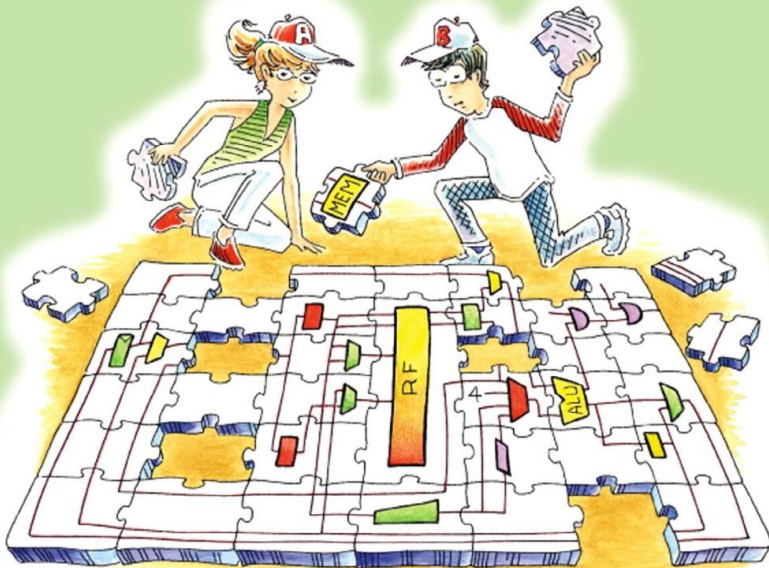
清华大学出版社



# 参考书

## Digital Design and Computer Architecture

SECOND EDITION



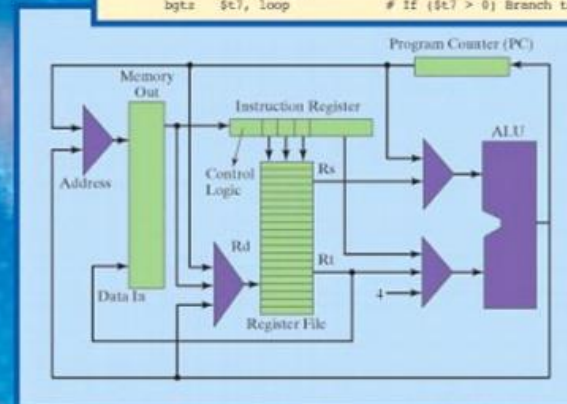
David Money Harris & Sarah L. Harris

MK  
MORGAN KAUFMANN

## MIPS Assembly Language Programming

```
li    $s2, 0           # Load Immediate $s2 = 0
li    $t7, 25          # Initialize loop counter to 25

loop:
add    $s2, $s2, $t7    # $s2 = $s2 + $t7
addi   $t7, $t7, -1     # Decrement loop counter
bgtz   $t7, loop        # If ($t7 > 0) Branch to loop
```



Robert L. Britton

# 课程内容 (32学时)

- Chapter 1 Fundamentals of Quantitative Design and Analysis (page 1-71)
- Chapter 2 Memory Hierarchy Design (page 72-147)
- Chapter 3 Instruction-Level Parallelism and Its Exploitation (page 148-261)
- Chapter 4 Data-Level Parallelism in Vector, SIMD, and GPU Architectures (page 262-343)
- Chapter 5 Thread-Level Parallelism (section 5.1-5.3, page 344-378)
- Chapter 6 Warehouse-Scale Computers to Exploit Request-Level and Data-Level Parallelism (page 432-493)
- Appendix A Instruction Set Principles (page A1-A54)
- Appendix B Review of Memory Hierarchy (page B1-B67)
- Appendix C Pipelining: Basic and Intermediate Concepts (page C1-C88)

- 第1章 量化设计与分析基础 (教材第1章, 6学时)
- 第2章 指令系统原理与实例 (教材附录A, 2学时)
- 第3章 流水线模型机 (教材附录C, 14学时)
- 第4章 存储器层次结构设计 (教材第2章和附录B, 8学时)
- 第5章 多处理器和线程级并行 (教材第5章的5.1-5.3, 2学时)

# 目标:

- ❖ 学习计算机性能评价、量化分析与设计的方法
- ❖ 掌握现代单CPU微处理器设计采用的主要方法和技术:
  - 指令级并行基本技术: 流水线CPU的设计与实现
  - 指令级并行重要技术: 乱序执行、分支预测、推测执行
  - cache与虚拟存储器
- ❖ 了解多处理器与多核系统结构

# 成绩构成：

---

**期末成绩： 70%**

**平时成绩： 30%（课堂练习和课后作业18%，半期12%）**



# 第1章 量化设计与分析基础

## 1.1 引言

## 1.2 计算机的分类

## 1.3 计算机系统结构定义和计算机的设计任务

## 1.4 实现技术的趋势

## 1.5 集成电路功耗的趋势

## 1.6 成本的趋势

## 1.7 可靠性

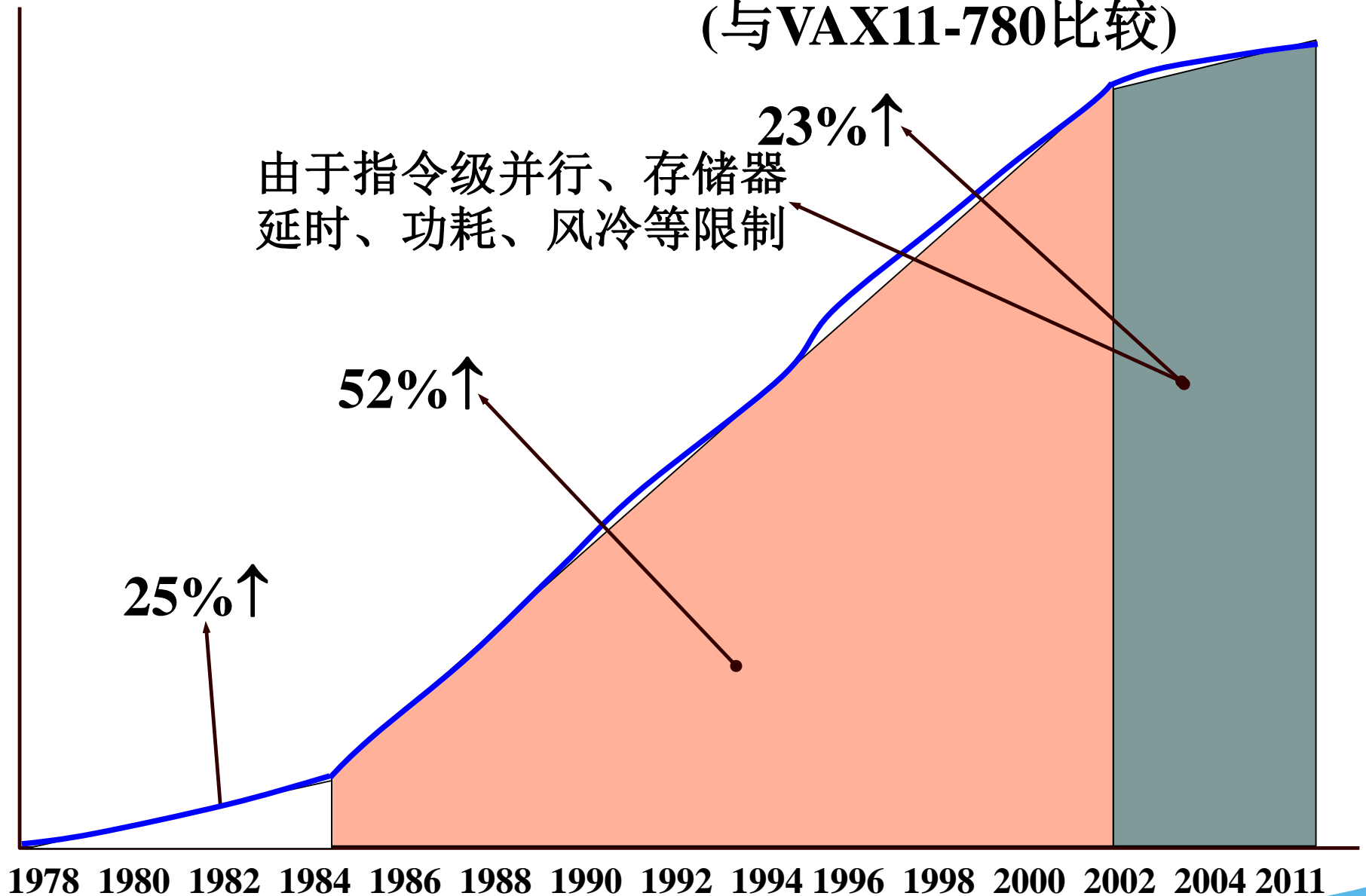
## 1.8 测量、报告和总结计算机性能

## 1.9 计算机设计的量化原则

## 1.10 综合：性能和性价比

# 性能增长趋势

(与VAX11-780比较)



# 1.1 引言

- 25% (1978-1986) : 性能增长主要依赖**实现技术**的进步
- **52% (1986-2003)** : 性能增长依赖**两方面**:
  - \* 实现技术的进步
  - \* **系统结构**革新 (RISC , 指令级并行ILP与Cache)
- 23% (2004-2011) : **指令级并行 ILP开发的限制**, 功耗**限制**, 因此其后性能提升手段出现了以下趋势:
  - **ILP => TLP and DLP** (线程级并行和数据级并行)
  - **更快的单核处理器=> 单芯片多处理器 (多核)**
  - **隐含在编译器和硬件的硬件级并行处理 => 显示的程序级并行**
- 15% (2011-2015)
- 3.5% (2015-2018)

# 60多年计算机性能有如此的改进？

## ■ 两个原因：

### ■ 构建计算机的实现技术发展

- 集成电路IC
- 存储部件(包括RAM 和 DISK)
- 外部设备

### ■ 计算机系统结构设计的改进

- Simple→complex→most complex

(RISC: DEC的VAX被RISC替代, Intel 将x86指令转换为RISC微指令)

- Cache集成在芯片上
- 指令级并行技术

- RISC特征：运算指令不访存、访存用LOAD/STORE指令、指令长度固定和有限的指令数目

# 性能增长的效果

- ❖ 适应各种计算机用户的能力大大增强
- ❖ 设计的各种计算机：基于微处理器
  - Minicomputer => 多微处理器构成的服务器
  - Mainframe => 大量微处理器组成的服务器
  - Supercomputer => 大量服务器构成的系统



# 微处理器的4代（4个十年）

- 1970年代的十年 “Microprocessors”
  - 可编程控制器（Programmable Controller）
  - 单片微处理器（Single-Chip Microprocessors）
  - 个人计算机，Personal Computers (PC)
- 1980年代的十年 “Quantitative Architecture”
  - 指令流水线（Instruction Pipelining）
  - 高速Cache存储器（Fast Cache Memories）
  - 编译器的考虑（Compiler Considerations）
  - 工作站，Workstations
- 1990年代的十年 “Instruction-Level Parallelism”
  - 超标量处理器，Superscalar Processors
  - 静态/动态指令调度（乱序执行），Aggressive Code Scheduling
  - 推测执行微结构，Speculative Microarchitectures
  - 低成本桌面超级计算，Low-Cost Desktop Supercomputing
- 2000年代的十年 “Thread-level/Data-level parallelism”

# 国内微处理器的发展情况

## ➤ 龙芯系列

- 2002年，**龙芯1号**正式问世，它采用**MIPS架构**。采用流水线结构，定点和浮点最高运算速度均超过每秒2亿次，性能与**PentiumII**大致相当，总体上达到了**1997**年的国际水平。
- **龙芯2号**：2004年6月，**64位**的龙芯2号发布，其性能相当于**P4**的水平，比“龙芯1号”性能提高**10至15**倍。
- **龙芯3号**：2009年的**3A1000**是我国首个**四核CPU芯片**，国内首次掌握了多核CPU的片间互连及**Cache**一致性技术。  
**3A1000**的第二次改版于**2012**年流片成功，至今还是龙芯销售的一款重要芯片，尤其是在工控领域。

2015年发射的**北斗**双星搭载的就是龙芯CPU。

# 自主CPU架构-----龙芯架构

- 2020年，龙芯中科推出了龙芯架构（Loongson Architecture，简称LoongArch），包括基础架构部分和向量指令、虚拟化、二进制翻译等扩展部分，近2000条指令。不包含MIPS指令。
- 同原有兼容指令系统相比，在硬件方面更易于高性能低功耗设计，在软件方面更易于编译优化和操作系统、虚拟机的开发。
- 龙芯架构不仅能够确保现有龙芯电脑上应用二进制的无损迁移，而且能够实现多种国际主流指令系统的高效二进制翻译。

## ➤ 海思麒麟系列

- 2009年，华为海思推出了首款移动微处理器**K3V1**，采用ARM架构，**110nm**制造工艺。主要面对中低端市场。
- 2012年，华为海思发布了**K3V2**，是当时全球最小的四核ARM A9架构处理器。集成GC4000的GPU，**40nm**制造工艺。用在华为P6和Mate1等产品上。

其后，华为推出的麒麟处理器全面采用**SoC**架构，即在**单芯片**上集成**CPU、通信模块、音视频解码以及外围电路**等一个完整系统。

- 2014年初，华为发布**麒麟910**（K3V2改进版，四核CPU结构）。麒麟910首次集成华为自研的巴龙Balong710基带，把GPU换成Mali，制造工艺**28nm**。

- 2014年6月，华为发布麒麟920 SoC芯片，28nm制造工艺，八核CPU结构，将4个ARM Cortex-A15和4个Cortex-A7处理器结合在一起，使同一应用程序可以在二者之间无缝切换。集成了协处理器i3，能以极低的功耗运行，持续采集加速计、陀螺仪和指南针等数据，使一些智能应用可以在待机下一直运行。
- 2015年11月华为发布了麒麟950 SoC芯片，八核CPU结构包括4个Cortex-A72和4个Cortex-A53，16nm制造工艺，集成自研Balong720基带、双核14-bit ISP和音视频解码芯片，还集成了i5协处理器，是一款集成度非常高的SoC。
- 2016年10月，华为发布了麒麟960 SoC芯片，八核CPU结构包括4个Cortex-A73和4个Cortex-A53，仍然是16nm制造工艺，GPU为Mali G71 MP8。



- 2017年9月，华为发布了人工智能芯片麒麟970 SoC芯片（八核CPU结构同麒麟960），10nm工艺、集成NPU（Neural Network Processing Unit，神经网络），处理海量数据。集成55亿个晶体管远高于高通和苹果芯片，华为步入顶级芯片厂商行列。
- 2018年8月发布麒麟980 SoC芯片（八核CPU结构包括4个Cortex-A76和4个Cortex-A55），7nm制造工艺，集成69亿个晶体管。全面升级的CPU、GPU、新的双核NPU使其性能更为优秀。
- 2019年9月华为发布了全球首款5G SoC芯片麒麟990 5G（八核CPU结构同麒麟980），内置巴龙5000基带，即内置5G。7nm制造工艺。GPU是16核Mali-G76，NPU在双核基础上增加一个微核。