

DSP的原理与应用

第一讲

概述

主讲：叶剑

电话：13728639620

Email: yejian@hit.edu.cn

1. 课程简介

- 什么是嵌入式系统及其分类
- 为什么学习**DSP28335**
- **DSP28335**的功能模块
- 动手实践

本课程的目标与任务

- 使学生能够掌握DSP设计和应用的基本方法。
- 掌握一整套用于开发和调试嵌入式应用的工具（CCS）。
- 通过本课程的课堂教学和后续的实验操作等环节，着重培养学生们的DSP软、硬件开发的综合分析与设计能力。

考核方式

考核环节	所占分值	考核与评价细则
实验项目及报告	40	(1) 按要求完成实验项目； (2) 完成实验设计，上交实验报告。
作业	20	独立完成DSP28335课后作业
期末考试	40	(1) 卷面成绩40分； (2) 考试命题以大纲中的应知应会内容为主，题型有填空、选择、简答等。

参考书目

- 《TMS320F28335DSP原理、开发及应用》
 - 符晓、朱洪顺编，清华大学出版社.2017年10月第1版
- 《手把手教你学DSP：基于TMS320F28335》
 - 张卿杰，徐友，左楠，卞康君编北京航空航天大学出版社. 2015年1月第1版.
- 《TMS320F28335,TMS320F28334,TMS320F28332, Digital Signal Controllers (DSCs), Data Manual》
 - Texas Instruments. June 2007.

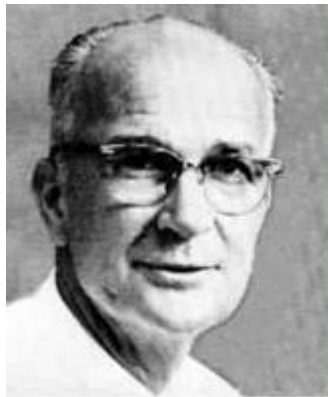
学习方法

- 听课
- 查看参考书和TI的数据手册
- 看参考设计
- 读示例代码
- 请教、讨论、交流
- 循序渐进、系统深入

2. 嵌入式系统的前世今生

混沌之初——晶体管之父、硅谷之父

肖克莱



肖克莱, W. B.

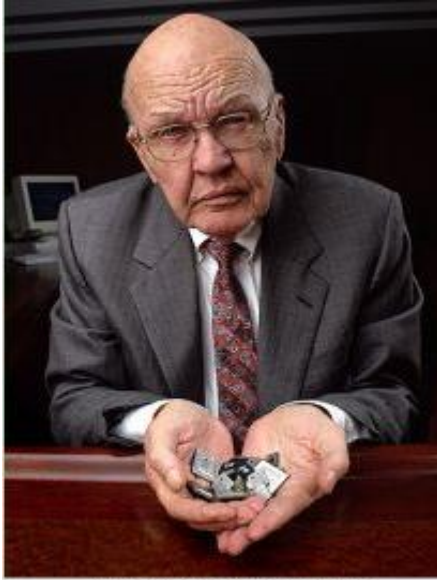
- 1、良好的启蒙教育-赢在起跑线
- 2、先有伯乐，而后有千里马
- 3、点接触晶体检波器的解释
- 4、时势造英雄-点接触晶体管的诞生
- 5、英雄造时势-多项晶体管技术的发明
- 6、功成身退——硅谷播火种

硅谷摇篮——仙童沉浮 “硅谷八叛逆”



- 1、八个天才的叛逆-仙童降生
- 2、第一个集成电路专利-仙童成长
- 3、成也萧何、败也萧何——
蒲公英播种、八叛逆出走
- 4、种子发芽、蒲公英芳华不再——
老迈仙童

硅谷之外——德州仪器 杰克·基尔比



Courtesy Texas Instruments

- 1、器件霸主-德州仪器
- 2、杰克·基尔比



- 1954年 生产首枚商用晶体管
- 1958年 TI工程师Jack Kilby发明首块集成电路(IC)
- 1967年 发明手持式电子计算器
- 1971年 发明单芯片微型计算机
- 1973年 获得单芯片微处理器专利
- 1978年 推出首个单芯片语言合成器，首次实现低成本语言合成技术
- 1982年 推出单芯片商用数字信号处理器(DSP)

.....

嵌入式系统的引入

- 由于信息技术、网络技术的高速发展和后PC时代的到来，嵌入式系统已经广泛渗透到科学研究、工程设计、军事技术、各类产业和商业文化艺术以及人们的日常生活中。嵌入式系统应用的例子比比皆是。可以说，嵌入式系统无所不在、无处不在。

应用举例（一）

➤ 消费电子



应用举例（二）

▶ 信息家电



应用举例 (三)

➤ 汽车电子



应用举例（四）

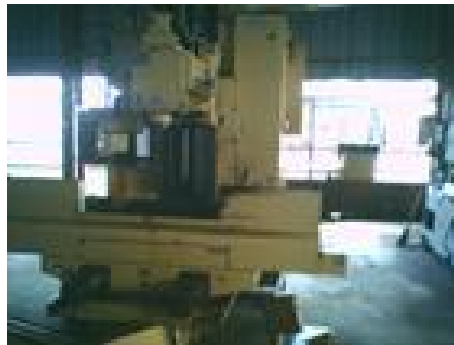
➤ 机器人

- 智能玩具
- 工业机器人
- 军用机器人



应用举例（五）

➤ 工业国防



前景

- 嵌入式系统的应用范围如此之广，涉及面如此之大，以至于有些学者断言 *嵌入式技术将成为后PC时代的主宰。*

何为嵌入式系统

- 简单的讲，就是嵌入到对象体中的**专用计算机系统**。
- 嵌入式系统三要素：
 - **嵌入性**：嵌入到对象体系中，有对象环境要求。
 - **专用性**：软、硬件按对象要求裁减。
 - **计算机**：实现对象的智能化功能。

定义

- **IEEE（Institute of Electrical and Electronics Engineers）的定义：**

devices used to control, monitor, or assist the operation of equipment, machinery or plants（用于控制、监视或者辅助操作设备和机器的装置）

- **目前国内一个普遍被认同的定义是：**

以应用为中心、以计算机技术为基础，软、硬件可裁剪，适应应用系统对功能、可靠性、成本、体积、功耗等严格要求的专用计算机系统。

3. 嵌入式系统的内涵

➤ 从技术角度

- 嵌入式系统是将应用程序、操作系统和计算机硬件集成在一起的系统。

➤ 从系统角度

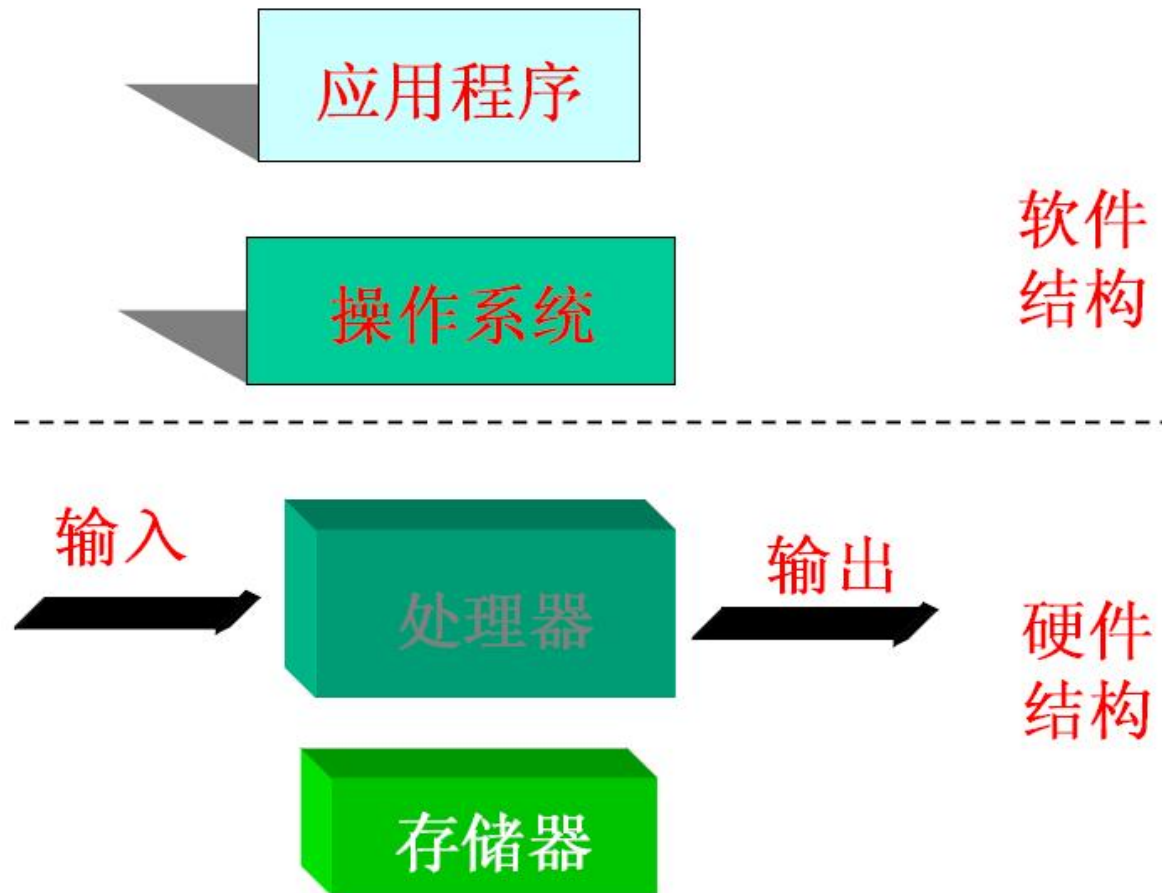
- 嵌入式系统是设计完成复杂功能的硬件和软件，并使其紧密耦合在一起的计算机系统。

注：软件+硬件

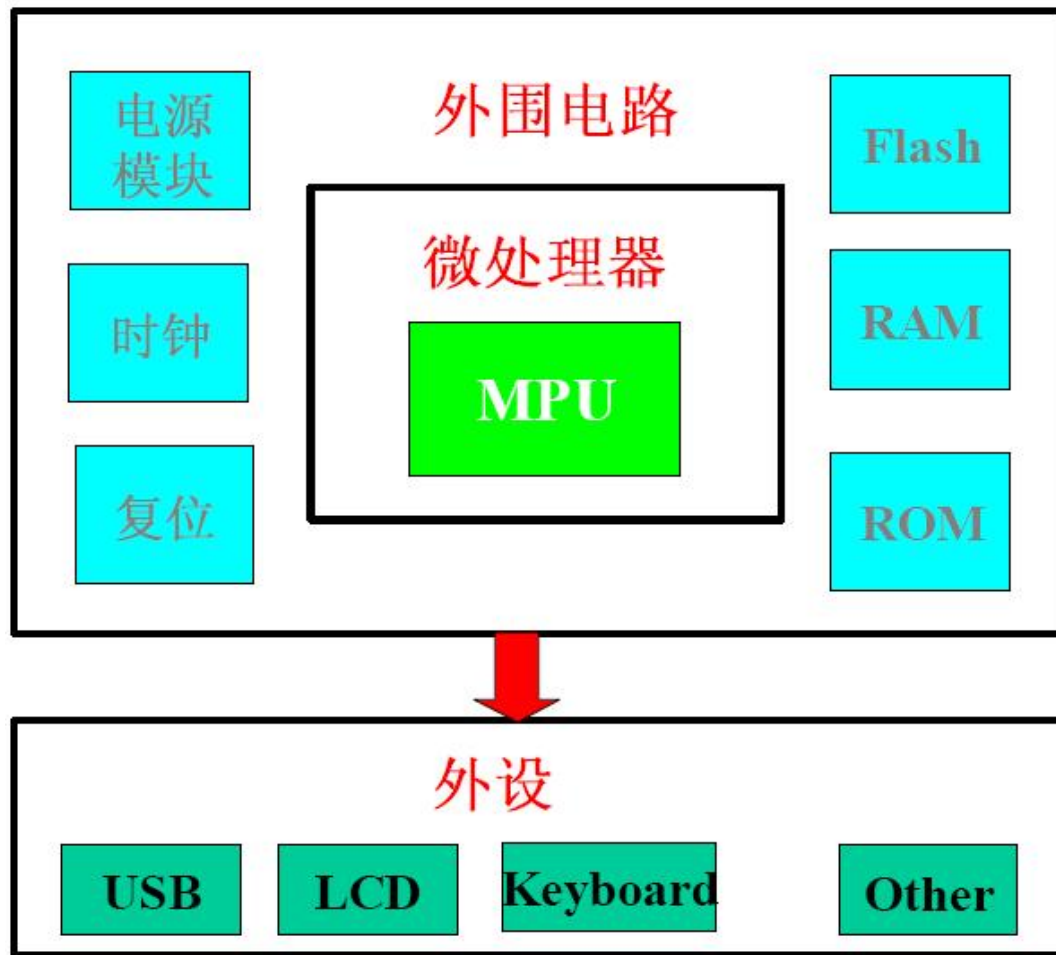
4. 嵌入式系统外延

- 广义的讲，凡是带有智能处理器的**专用软硬件系统**都可称为嵌入式系统。

5. 嵌入式系统组成



嵌入式系统的硬件组成



6. 特点（对通用计算机）

- 专用性
 - 采用专门的处理器
 - 功能算法的专用性
- 小型化（资源有限）
 - 结构紧凑、坚固可靠、计算资源有限
- 软硬件设计一体化
 - 硬件与软件的依赖性强，一般需要协同设计
 - 应用软件与操作系统的一体化设计
- 需要交叉开发环境
 - 本身资源受限，开发由宿主机完成

7. 嵌入式系统历史

- 20世纪70年代单片机的出现，这时的应用只是使用8位的芯片执行一些单线程的程序，还谈不上“系统”的概念。
- 20世纪80年代出现商业级的嵌入式“操作系统”（一个实时核），并在其上编写嵌入式应用软件。
 - 获取更短的开发周期
 - 更高的开发效率
 - “嵌入式系统”真正出现

商业化的嵌入式实时内核开始出现，如VTRX32，PSOS，VxWorks等
- 20世纪90年代以后，随着对实时性要求的提高，软件规模不断上升，实时核逐渐发展为实时多任务操作系统（RTOS），并作为一种软件平台逐步成为目前国际嵌入式系统的主流。

8. 嵌入式系统的分类

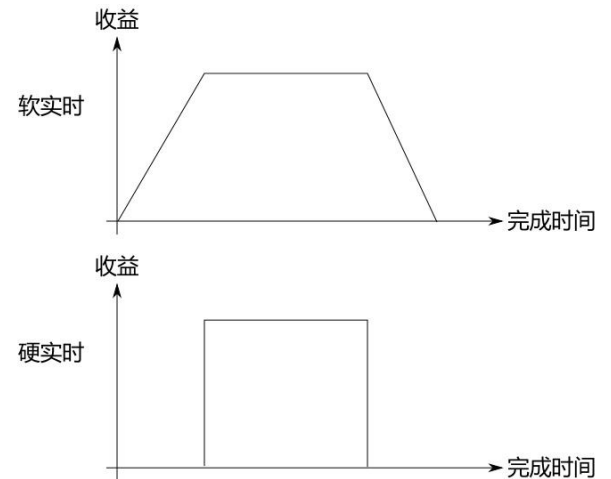
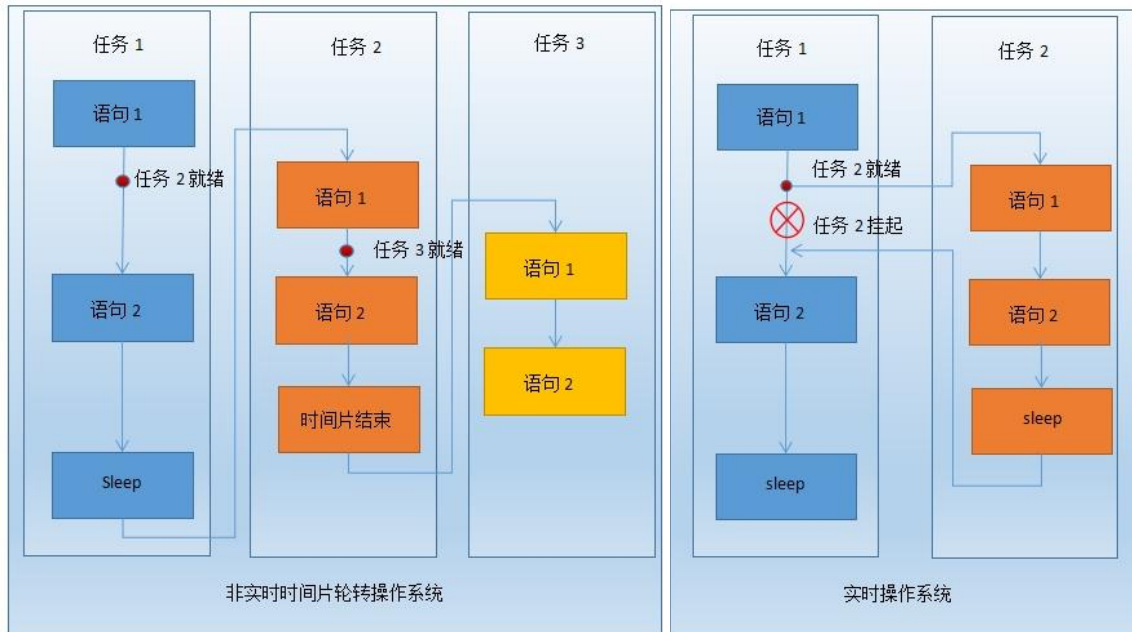
按表现形式分（硬件范畴）：

- 系统级：各种类型的工控机。
- 板级：各种类型的带CPU的主板。
- 片级：各种以单片机、DSP、微处理器为核心的产品。

嵌入式系统的分类

按实时性要求分（软件范畴）：

- 非实时系统（Linux/Windows/OSX）
- 软实时系统（消费类产品）
 - ❑ 超时会导致性能下降
- 硬实时系统（工业和军工系统）
 - ❑ 超时会导致系统失败



9. 片级嵌入式系统的分类

按芯片封装功能丰富程度可分为：

- 嵌入式微处理器MPU（Micro Processor Unit）
- 嵌入式微控制器MCU（Micro Controller Unit）
- 嵌入式DSP（Digital Signal Processor）
- 片上系统SoC（System on Chip）

嵌入式微处理器

嵌入式微处理器是由通用计算机中的CPU演变而来。嵌入式微处理器可谓是通用计算机中CPU的微缩版。相对于通用CPU，嵌入式微处理器具有体积小、功耗少、成本低的优点，当然在速度上也慢一些。典型的如32位的ARM。

嵌入式微控制器

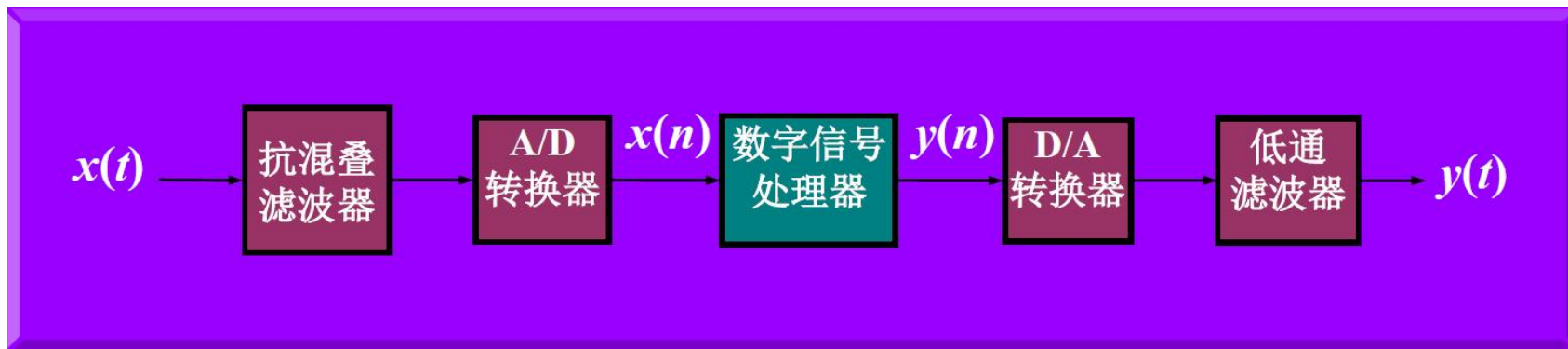
- 嵌入式**微控制器**又称**单片机**。
- 单片机是指随着大规模集成电路的出现及其发展，将计算机的CPU、RAM、ROM、定时计数器和多种I/O接口集成在一片芯片上，形成芯片级的计算机，为不同的应用场合做不同组合控制。
- 微控制器是目前嵌入式系统工业的主流。
- 微控制器的片上外设资源丰富，适合用于控制，因此称为微控制器。
- 与嵌入式微处理器的区别（大体上）
 - ❑ 微控制器——存储器在**片内**
 - ❑ 微处理器——存储器在**片外**

DSP

DSP (digital signal processor) 是一种**独特**的微处理器，有自己的完整指令系统，是以数字信号来处理大量信息的器件。一个数字信号处理器在一块不大的**芯片内**包括有控制单元、运算单元、各种寄存器以及一定数量的存储单元等等，在其**外围**还可以连接若干存储器，并可以与一定数量的外部设备互相通信，有软、硬件的全面功能，本身就是一个微型计算机。

DSP工作原理

DSP的工作原理是接收模拟信号，转换为0或1的数字信号，再对数字信号进行修改、删除、强化，并在其他系统芯片中把数字数据解译回模拟数据或实际环境格式。它的**强大数据处理能力**和**高运行速度**，是最值得称道的两大特色。



DSP特点

DSP芯片，由于它运算能力很强，速度很快，体积很小，而且采用软件编程具有高度的灵活性，因此为从事各种复杂的应用提供了一条有效途径。其主要应用是实时快速地实现各种数字信号处理算法。根据数字信号处理的要求，DSP芯片一般具有如下主要特点：

1. 在一个指令周期内可完成一次乘法和一次加法（乘法器MUL/累加器ACC）；

DSP特点

2. 程序和数据空间分开，可以同时访问指令和数据（哈佛结构）；
3. 片内具有快速RAM，通常可通过独立的数据总线在两块中同时访问；
4. 具有低开销或无开销循环及跳转的硬件支持；
5. 快速的中断处理和硬件I/O支持；

DSP特点

6. 可以并行执行多个操作;
7. 支持流水线操作，使取指、译码和执行等操作可以重叠执行。

当然，与通用微处理器相比，DSP芯片的其他通用功能相对较弱些。

DSP 生产厂商

目前，世界上生产DSP的厂商多如牛毛，但是真正具有实力的就要数TI公司了。美国德州仪器（Texas Instruments, TI）是最著名的DSP芯片生产厂商，其产品应用也最广泛，TI公司生产的TMS320系列DSP芯片广泛应用于各个领域。TI公司在1982年成功推出了其第一代DSP芯片TMS32010，这是DSP应用历史上的一个里程碑，从此，DSP芯片开始得到真正的广泛应用。由于TMS320系列DSP芯片具有价格低廉、简单易用、功能强大等特点，所以逐渐成为目前最有影响、最为成功的DSP系列处理器。

DSP 分类

目前，TI公司在市场上主要有三大系列产品：

面向数字控制、运动控制的**TMS320C2000**系列，主要包括 TMS320C24x/F24x、TMS320LC240x/LF240x、TMS320C24xA/LF240xA、**TMS320C28xx/F28xx**等。

DSP分类

目前，TI公司在市场上主要有三大系列产品：

面向低功耗、手持设备、无线终端应用的**TMS320C5000**系列，主要包括TMS320C54x、TMS320C54xx、TMS320C55x等。

面向高性能、多功能、复杂应用领域的**TMS320C6000**系列，主要包括TMS320C62xx、TMS320C64xx、TMS320C67xx等。

ARM

ARM（Advanced RISC Machines），既可以认为是一个公司的名字，也可以认为是对一类微处理器的通称，还可以认为是一种技术的名字。

1991年ARM公司成立于英国剑桥，主要出售芯片设计技术的授权。

ARM最大的优势在于速度快、低功耗、芯片集成度高，基本上外围加上电源和驱动接口就可以做成一个小系统了。

ARM架构

Main article: *List of ARM microarchitectures*

Architecture ↕	Core bit-width ↕	Cores	
		Arm Holdings ↕	Third-party
ARMv1	32 ^[a 1]	ARM1	
ARMv2	32 ^[a 1]	ARM2, ARM250, ARM3	Amber, STORM Open Soft Core ^[40]
ARMv3	32 ^[a 2]	ARM6, ARM7	
ARMv4	32 ^[a 2]	ARM8	StrongARM, FA526, ZAP Open Source Processor Core ^[41]
ARMv4T	32 ^[a 2]	ARM7TDMI, ARM9TDMI, SecurCore SC100	
ARMv5TE	32	ARM7EJ, ARM9E, ARM10E	XScale, FA626TE, Feroceon, PJ1/Mohawk
ARMv6	32	ARM11	
ARMv6-M	32	ARM Cortex-M0, ARM Cortex-M0+, ARM Cortex-M1, SecurCore SC000	
ARMv7-M	32	ARM Cortex-M3, SecurCore SC300	
ARMv7E-M	32	ARM Cortex-M4, ARM Cortex-M7	
ARMv8-M	32	ARM Cortex-M23, ^[42] ARM Cortex-M33 ^[43]	
ARMv7-R	32	ARM Cortex-R4, ARM Cortex-R5, ARM Cortex-R7, ARM Cortex-R8	
ARMv8-R	32	ARM Cortex-R52	
ARMv7-A	32	ARM Cortex-A5, ARM Cortex-A7, ARM Cortex-A8, ARM Cortex-A9, ARM Cortex-A12, ARM Cortex-A15, ARM Cortex-A17	Qualcomm Krait/Scorpion, PJ4/Sheeva, Apple Swift
ARMv8-A	32	ARM Cortex-A32	
	64/32	ARM Cortex-A35, ^[48] ARM Cortex-A53, ARM Cortex-A57, ^[49] ARM Cortex-A72, ^[50] ARM Cortex-A73 ^[51]	X-Gene, Nvidia Project Denver 1/2, Cavium Thunder X, ^[52] ^[53] ^[54] AMD K12, Apple Cyclone/Typhoon/Twister/Hurricane/Zephyr/Monsoon/Mistral, Qualcomm Kryo, Samsung M1/M2 ("Mongoose") ^[55] /M3 ("Meerkat")
ARMv8.1-A	64/32	TBA	ThunderX2 ^[58]
ARMv8.2-A	64/32	ARM Cortex-A55, ^[59] ARM Cortex-A75, ^[60] ARM Cortex-A76, ^[61] Cortex-A65, Neoverse E1, ARM Cortex-A77	HiSilicon 980, ^[62] Nvidia Carmel, Samsung M4 ("Cheetah"), ^[63] Fujitsu A64FX (ARMv8 SVE 5)
ARMv8.3-A	64/32	TBA	Apple Vortex/Tempest
ARMv8.4-A	64/32	TBA	
ARMv8.5-A	64/32	TBA	

ARM应用

V • T • E

Application ARM-based chips

[ARM Holdings](#) • [ARM architecture](#) • [List of ARM microarchitectures](#) • [List of applications of ARM cores](#) • [ARM Cortex-A](#) • [ARM Cortex-R](#) • [ARM Cortex-M](#)

Application processors (32-bit)	Cortex-A5	Actions ATM702x • Amlogic M805/S805 , T82x • Atmel SAMA5D3 • InfoTM iMAPx820 , iMAPx15 • Qualcomm Snapdragon
	Cortex-A7	Allwinner A2x , A3x , A83T , H3 , H8 • NXP i.MX7 , QorIQ LS10xx , NXP i.MX6UL • Broadcom VideoCore BCM2836 , BCM23550 • PXA1920 , 1500 mini plus • MediaTek MT65xx • Qualcomm Snapdragon 200 , 400
	Cortex-A8	Allwinner A1x • Apple A4 • Freescale i.MX5 • Rockchip RK291x • Samsung Exynos 3110(S5PC110) , S5PV210 • Texas Instru DM38x • ZiiLABS ZMS-08
	Cortex-A9	Actions ATM702x , ATM703x • Altera Cyclone V , Arria V/10 • Amlogic AML8726 , MX , M6x , M801 , M802/S802 , S812 , T86x • i.MX6 • HiSilicon K3V2 , 910's • InfoTM iMAPx912 • Leadcore LC1810 , LC1811 • Marvell Armada 1500 mini • MediaTek MT65xx • Renesas EMMA EV2 , R-Car H1 , RZ/A • Rockchip RK292x , RK30xx , RK31xx • Samsung Exynos 4 421x , 441x • ST-Ericsson N950 • Texas Instruments Sitara AM4xxx • VIA WonderMedia WM88x0 , 89x0 • Xilinx Zynq-7000 • ZiiLABS ZMS-20 , ZMS-40
	Cortex-A15	Allwinner A80 • HiSilicon K3V3 • MediaTek MT8135/V • Nvidia Tegra 4 , K1 • Renesas R-Car H2 • Samsung Exynos 5 52xx , 5800 • Texas Instruments Sitara AM5xxx
	Cortex-A17	MediaTek MT6595 , MT5595 • MStar 6A928 • Rockchip RK3288
	ARMv7-A compatible	Apple A6 , A6X , S1 , S1P , S2 , S3 • Broadcom Brahma-B15 • Marvell P4J • Qualcomm Snapdragon S1 , S2 , S3 , S4 Plus , S4 Pro , S5
	Others	Enhanced Cortex-A8 • Enhanced Cortex-A9 • Cortex-A12 • Cortex-A32

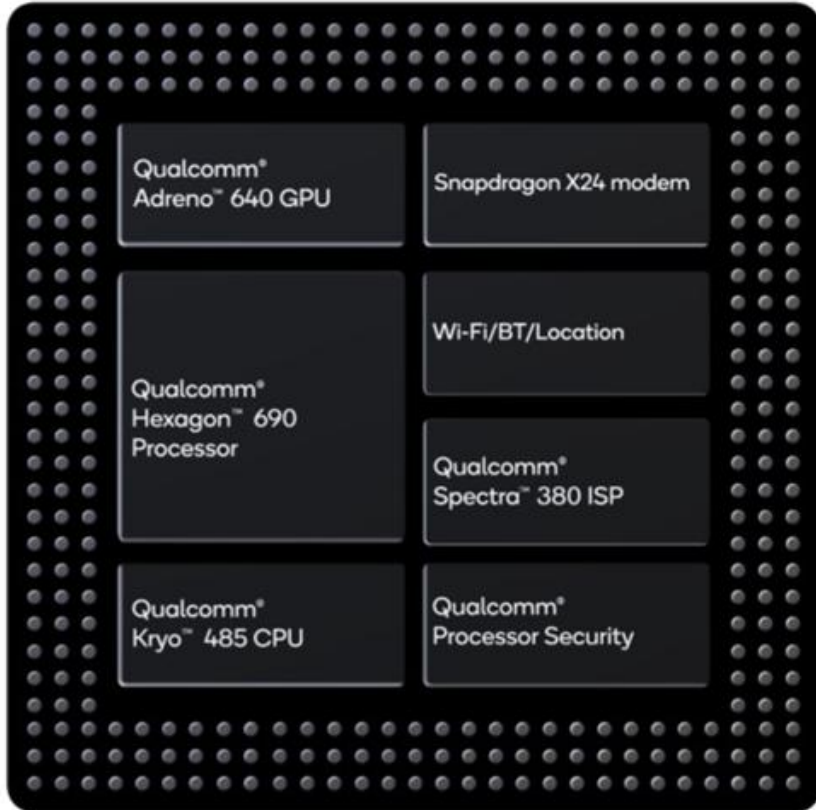
ARM应用

Application processors (64-bit)	Cortex-A35	NXP i.MX8X • MediaTek MT6799, MT8516 • Rockchip RK3308
	Cortex-A53	Actions GT7, S900, V700 • Allwinner A64, H5, H64, R18 • Altera Stratix 10 • Amlogic S9 Family, T96x • Broadacom PXA1928, Mobile PXA1908/PXA1936 • MediaTek MT673x, MT675x, MT6762, MT6763, MT6765, MT6771, MT6775, MT6795, Helio A2x, P1x, P2x, P3x, G2x, G3x • NXP ARM S32, QorIQ LS1088, LS1043, i.MX8M • Qualcomm Snapdragon 800, RK3328, RK3368 • Samsung Exynos 7 75xx, 78xx • Spreadtrum SC9860/GV, SC9836 • Texas Instruments Sitara
	Cortex-A55	Samsung Exynos 850 • Unisoc SC9863A
	Cortex-A57	AMD Opteron A1100-series • NXP QorIQ LS20xx • Nvidia Tegra X1 and Tegra X2 • Qualcomm Snapdragon 800
	Cortex-A72	HiSilicon Kirin 95x, Kunpeng 916 • MediaTek Helio X2x, MT817x • MStar 6A938 • Qualcomm Snapdragon 800, QorIQ LS1028A, i.MX8
	Cortex-A73	Qualcomm Snapdragon 460, 63x, 66x, 835 • Samsung Exynos 78xx, 79xx, 96xx • HiSilicon Kirin 710, 960, 970
	Cortex-A75	Qualcomm Snapdragon 670, 71x, 845, 850 • Samsung Exynos 9820, 9825 • MediaTek Helio P65, P9x, G70, G70T
	Cortex-A76	HiSilicon Kirin 810, 820, 980, 985, 990 • Qualcomm Snapdragon 480, 67x, 72x, 73x, 76x, 855(+), 7c, 8x and 8cx
	Cortex-A77	MediaTek Dimensity 1000 • Qualcomm Snapdragon 690, 750G, 865(+) and 870 • HiSilicon Kirin 9000(E) • NXP i.MX8M
	Cortex-A78	MediaTek Dimensity 1100, 1200 • Qualcomm Snapdragon 888 • Samsung Exynos 1080, 2100
	Cortex-X1	Qualcomm Snapdragon 888 • Samsung Exynos 2100
	ARMv8-A compatible	Apple A7, A8, A8X, A9, A9X, A10, A10X, A11, A12, A12X, A12Z, A13, A14, M1 • Applied Micro X-Gene • Cavium ThunderX2 (Denver2), Tegra Xavier (Carmel), • Qualcomm Kryo, Falkor • Samsung Mongoose

ARM应用

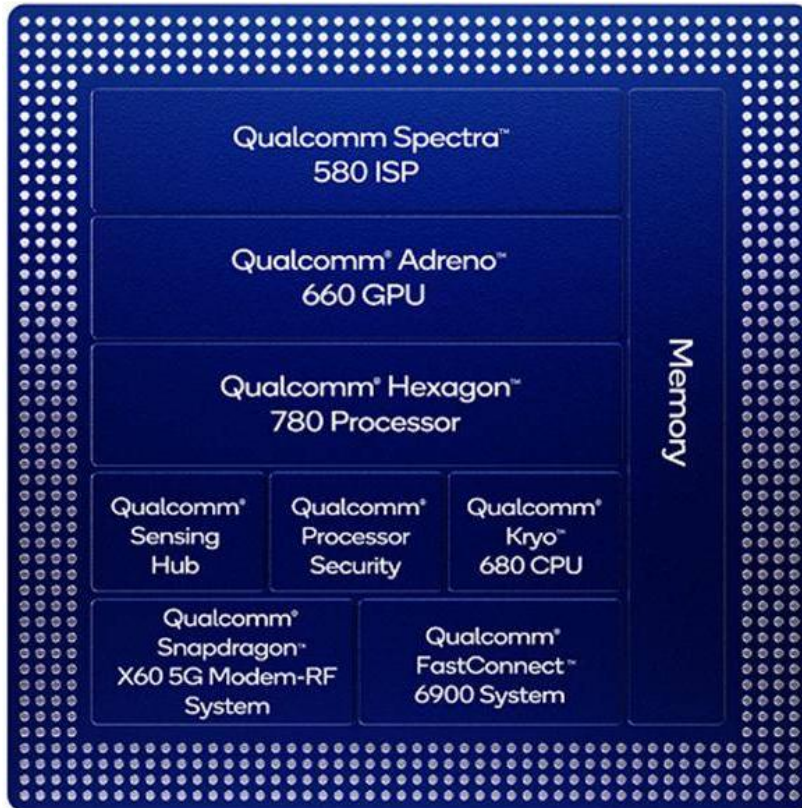


骁龙855



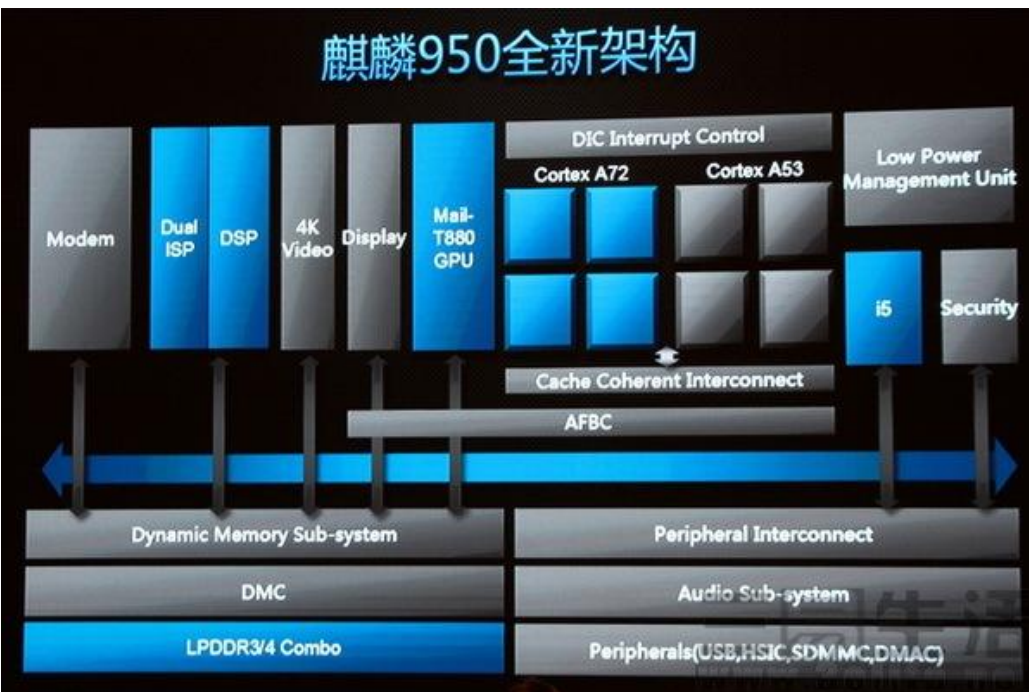
- 骁龙855架构（SoC）
- Hexagon 690 DSP
- Spectra 380 ISP，一种特殊的DSP
- Kryo 485 CPU， ARM 架构

骁龙888



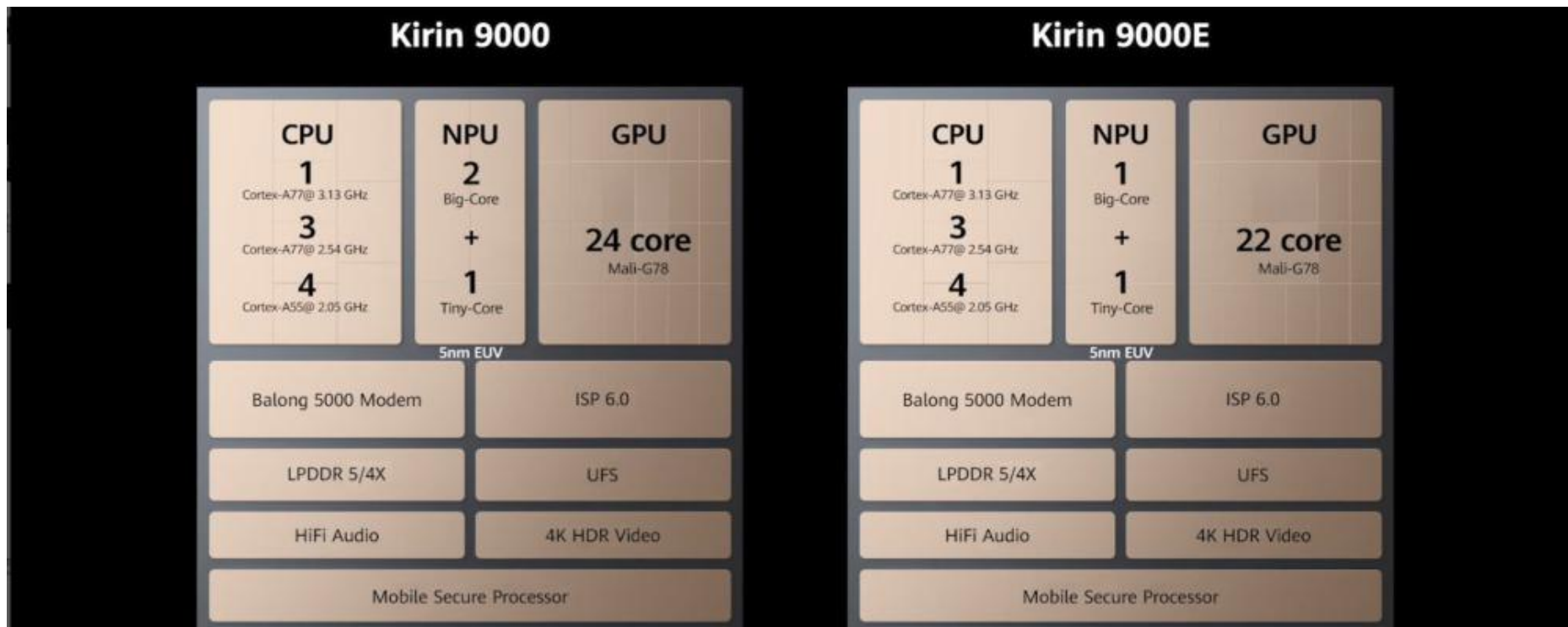
- 骁龙888架构（SoC）
- Hexagon 780 DSP
- Spectra 580 ISP，一种特殊的DSP
- Kryo 680 CPU， ARM 架构

麒麟980



- 麒麟980架构 (SoC)
- Cortex-A76, ARM
- Image signal processor (ISP): Improved dual 14-bit ISP 1000 Mp/s with standalone digital signal processor (DSP)

麒麟9000



- 麒麟9000架构（SoC）
- 1颗频率为3.13GHz的A77超级大核，3颗2.54GHz的A77大核，以及4颗2.05GHz的A55小核，ARM

骁龙855 VS麒麟980

SoC	骁龙855	骁龙845	麒麟980	苹果A12 Bionic
CPU	<p>1 x Kryo 485 Gold (基于A76) @ 2.84GHz 1x 512KB L2 Cache</p> <p>3 x Kryo 485 Gold (基于A76) @ 2.41GHz 3x 256KB L2 Cache</p> <p>4 x Kryo 485 Silver (基于A55) @ 1.78GHz 4x 128KB L2 Cache</p> <p>共享2ML3 Cache</p>	<p>4 x Kryo 385 Gold (基于A75) @ 2.803GHz 4x 256KB L2 Cache</p> <p>4 x Kryo 385 Silver (基于A55) @ 1.766GHz 4x 128KB L2 Cache</p> <p>共享2ML3 Cache</p>	<p>2 x HiSilicon Cortex-A76 @ 2.6GHz 2x 512KB L2 Cache</p> <p>2 x HiSilicon Cortex-A76 @ 1.92GHz 2x 512KB L2 Cache</p> <p>4 x ARM Cortex-A55 @ 1.805GHz 4x 128KB L2 Cache</p> <p>共享4ML3 Cache</p>	<p>2 x Vortex @ 2.5GHz 4 x Tempest @ 1.59GHz</p>
GPU	Adreno 640 @ 585MHz	Adreno 630 @ 710MHz	Mali-G76 MP10 @ 720MHz	4-core "G11P" @ >~1.1GHz
内存	4x16bit, LPDDR4x @ 2133MHz	4x16bit, LPDDR4x @ 1866MHz	LPDDR4x @ 2133MHz	LPDDR4x
调制器	X24调制解调器 LTE最大下载/上传 2Gbps/316Mbps	X20调制解调器 LTE最大下载/上传 1.2 Gbps/150 Mbps	Kirin 980 Integrated LTE LET最大下载/上传 1.4Gbps/200Mbps	英特尔 XMM7560 调制调节器
视频解码	H.265 (HEVC), H.264 (AVC), HLG, HDR10, HDR10+, VP8, VP9	HDR10, HLG, 和 H.265 (HEVC)	4K video, HDR 10	HEVC、H.264、MPEG-4 Part 2、Motion JPEG、HDR10

5nm 旗舰芯片

				
芯片	A14	骁龙 888	Exynos 1080	麒麟 9000
工艺	5nm EUV	5nm	5nm EUV FFET	5nm
CPU	3.01GHz+1.80GHz	Cortex-X1 3xCortex-A78 4xCortex-A55	Cortex-A78@2.8GHZ 3xCortex-A78 4xCortex-A55	Cortex-A77@3.13GHz 3xCortex-A77 4xCortex-A55
GPU	自研四核心	Adreno 660	Mali-G78	Mali-G78
NPU	神经网络单元	新第六代Qualcomm AI引擎	高性能NPU	双大核+1小内核
跑分	约580000分左右	740000分左右	约693600分	693605分
存储	苹果高速闪存	LPDDR5	LPDDR5/LPDDR4x	LPDDR5/LPDDR4x
Modem	2G/3G/4G/5G (包含sub-6 GHz和毫米波)	2G/3G/4G/5G (包含sub-6 GHz和毫米波)	2G/3G/4G/5G (包含sub-6 GHz和毫米波)	2G/3G/4G/5G (包含sub-6 GHz和毫米波)
Wi-Fi	Wi-Fi 6	Wi-Fi 6E	Wi-Fi 6	Wi-Fi 6+
视频拍摄	杜比视界 4K 60帧 10bit HDR	每秒捕获120帧, 且每帧都是1200万像素	支持拍摄10-bit色域视频	4K HDR 视频拍摄
影像	三摄系统+激光LiDAR	每秒可处理 27 亿像素, 提供更强的影像算力	最高支持2亿像素 最多6个摄像头同时开启 最多三摄同时操作	超感知徕卡电影影像系统

DSP和ARM的区别

- ARM, 1) 强的事务管理功能; 2) 速度和数据处理能力一般; 3) 外围接口比较丰富; 4) 标准化和通用性做的很好; 5) 低功耗。所以适合用在一些**消费电子产品**。
- DSP主要是用来计算的, 比如进行加密解密、调制解调等, 优势是**强大的数据处理能力和较高的运行速度**。由于其在控制算法等方面很擅长, 所以适合用在对控制要求比较高的场合, 比如军用导航、电机驱动等方面。

发展趋势

- 是ARM取代DSP，还是DSP挤掉ARM？
- 今天的手机生产采用的是**双核**方式：**DSP芯片**处理通信，如调制解调器功能和语音处理等；一块**通用处理器**（通常是ARM设计的RISC处理器）负责处理手机上运行的各种程序，如用户界面和控制协议堆栈等。

SoC

- SoC构成可以是系统级芯片控制逻辑模块、微处理器CPU内核模块、DSP模块、嵌入的存储器模块、和外部进行通讯的接口模块、电源提供和功耗管理模块,对于一个无线SoC还有射频前端模块。

SoC

CPU仅占SoC面积的~15%左右

基本计算

移动要求
时刻在线



创新的
用户体验

完整集成的移动平台

10. 嵌入式系统的应用

- 工业控制
- 交通管理
- 信息家电
- 家庭智能管理系统
- POS网络及电子商务，电子政务
- 环境检测
- 机器人
- ...

11. 嵌入式系统的发展趋势

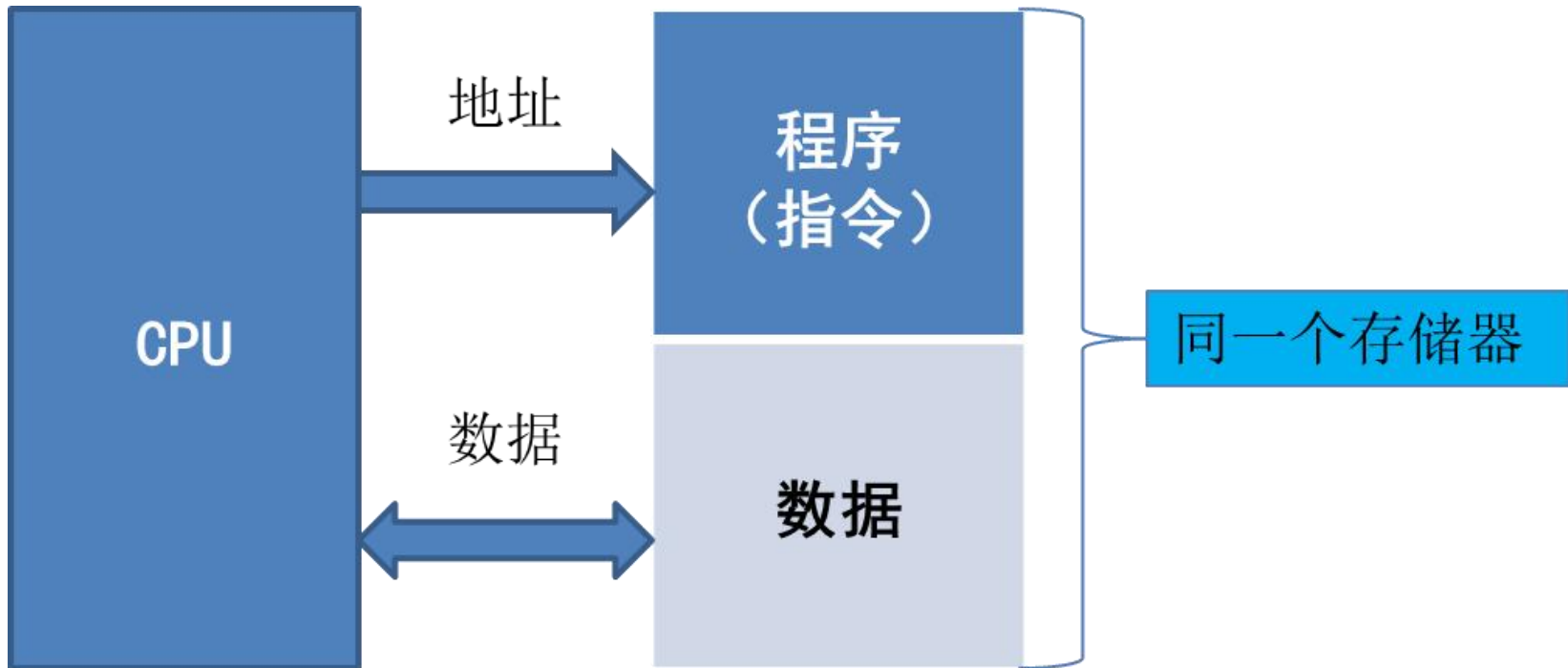
- 嵌入式应用软件的开发需要强大的开发工具和操作系统的支持
- 通过互联网相互连接成为必然趋势
- 支持小型电子设备实现小尺寸、低功耗和低成本
- 提供精巧的多媒体人机界面

基于哈佛结构的流水线操作

12.冯诺依曼结构

- 冯·诺依曼结构是一种将**程序指令存储器**和**数据存储器**合并在一起的存储器结构。
- 取指令和取操作数都在**同一总线**上，通过**分时复用**的方式进行。
- 缺点是在高速运行时，**不能达到同时取指令和取操作数**，从而形成了传输过程的瓶颈。

12.冯诺依曼结构

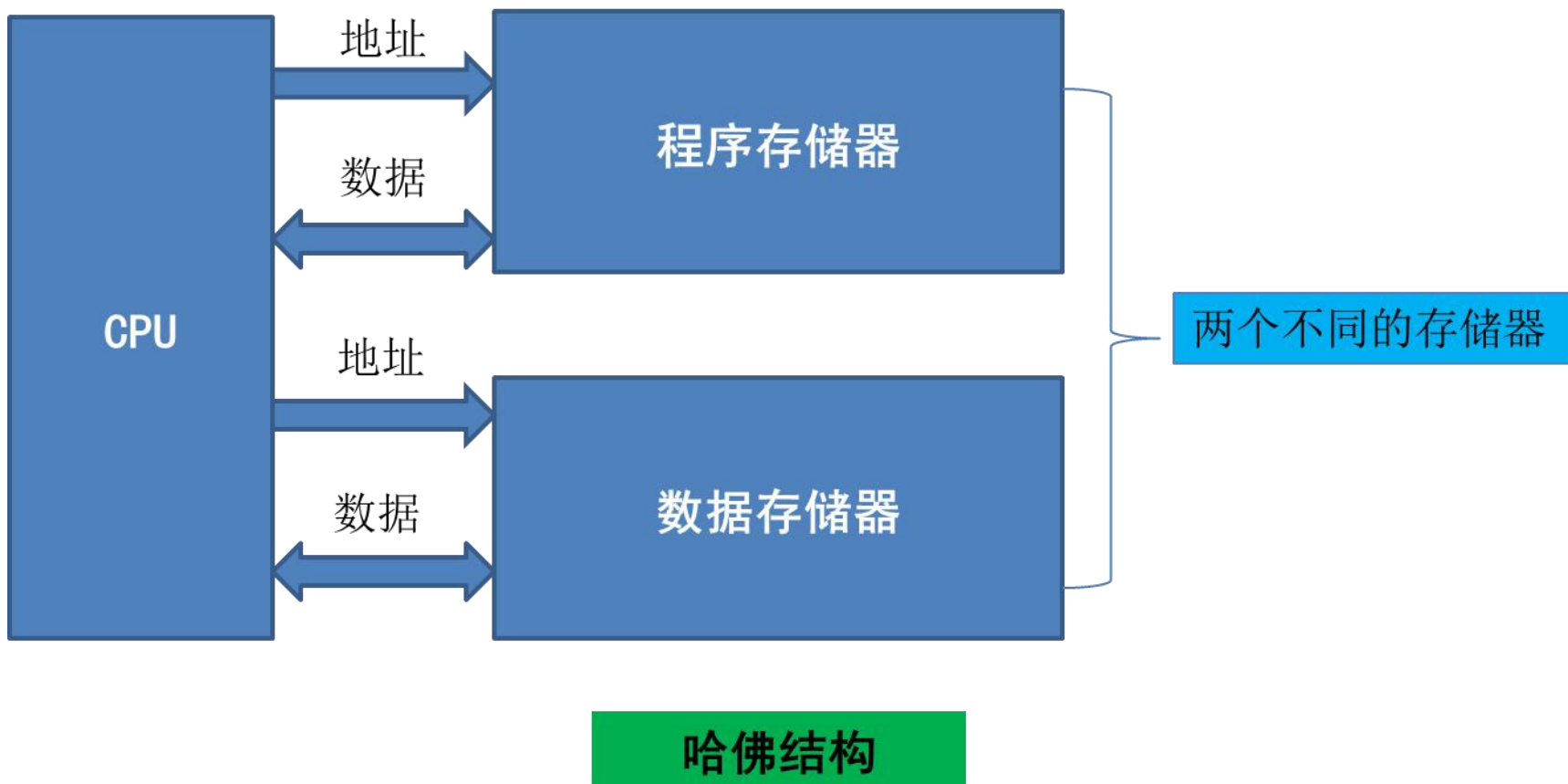


冯诺依曼结构

13.哈佛结构

- 哈佛结构是一种将**程序指令存储**和**数据存储**分开的存储器结构。
- 程序存储器和数据存储器是**两个独立的存储器**，每个存储器独立编址、独立访问，目的是为了减轻程序运行时的访存瓶颈。
- 哈佛结构的CPU首先到程序指令储存器中读取程序指令内容，解码后得到数据地址，再到相应的数据储存器中读取数据，并进行下一步的操作（通常是执行）。
- 程序指令储存和数据储存分开，允许在**一个机器周期**同时获得指令和数据。

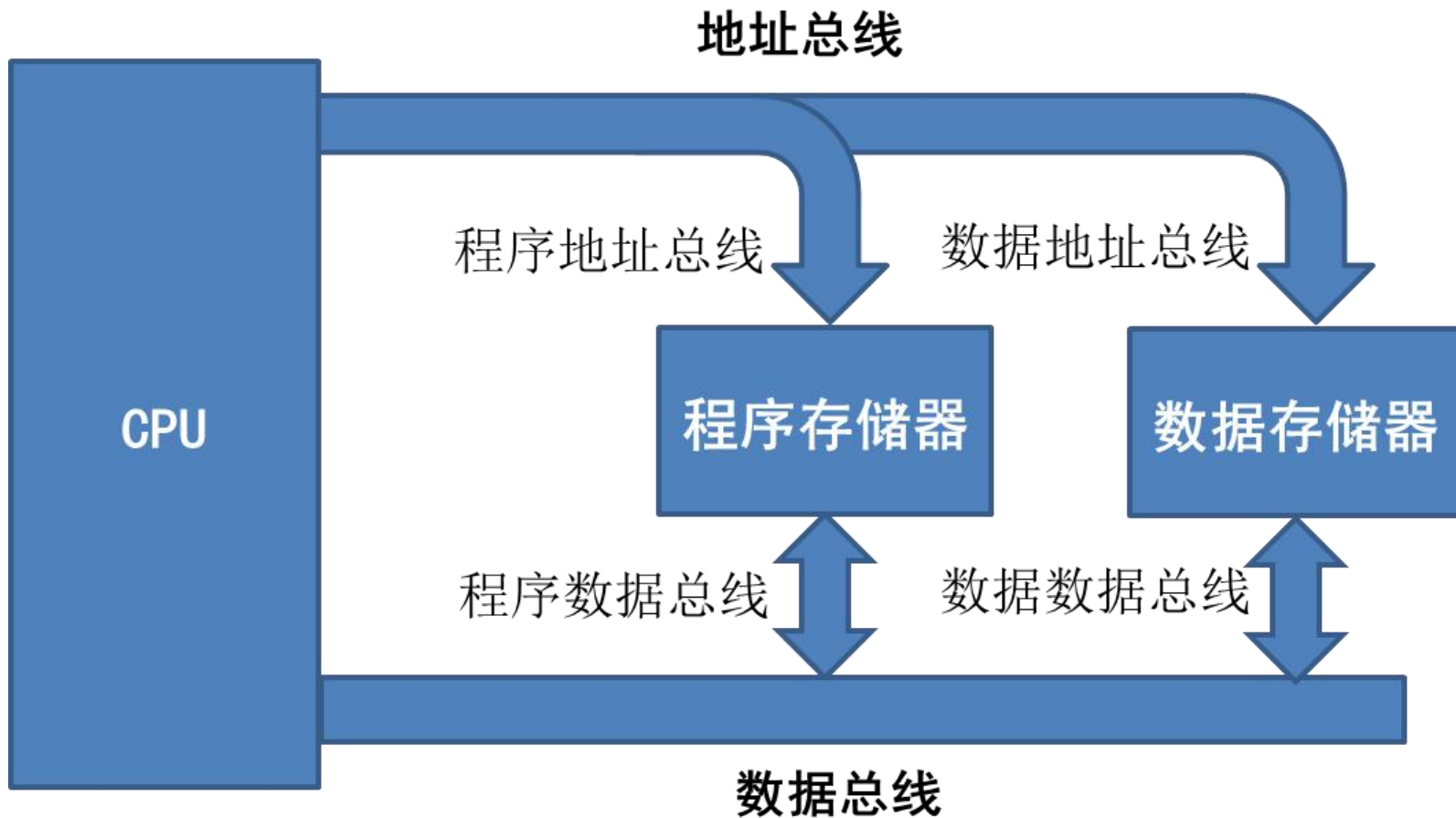
13.哈佛结构



14.改进型哈佛结构（拓展了解）

- 改进型哈佛结构虽然也使用两个不同的存储器：**程序存储器**和**数据存储器**。
- 它把两个存储器的**地址总线合并了**，**数据总线也进行了合并**，即原来的哈佛结构需要4条不同的总线，改进后需要两条总线。

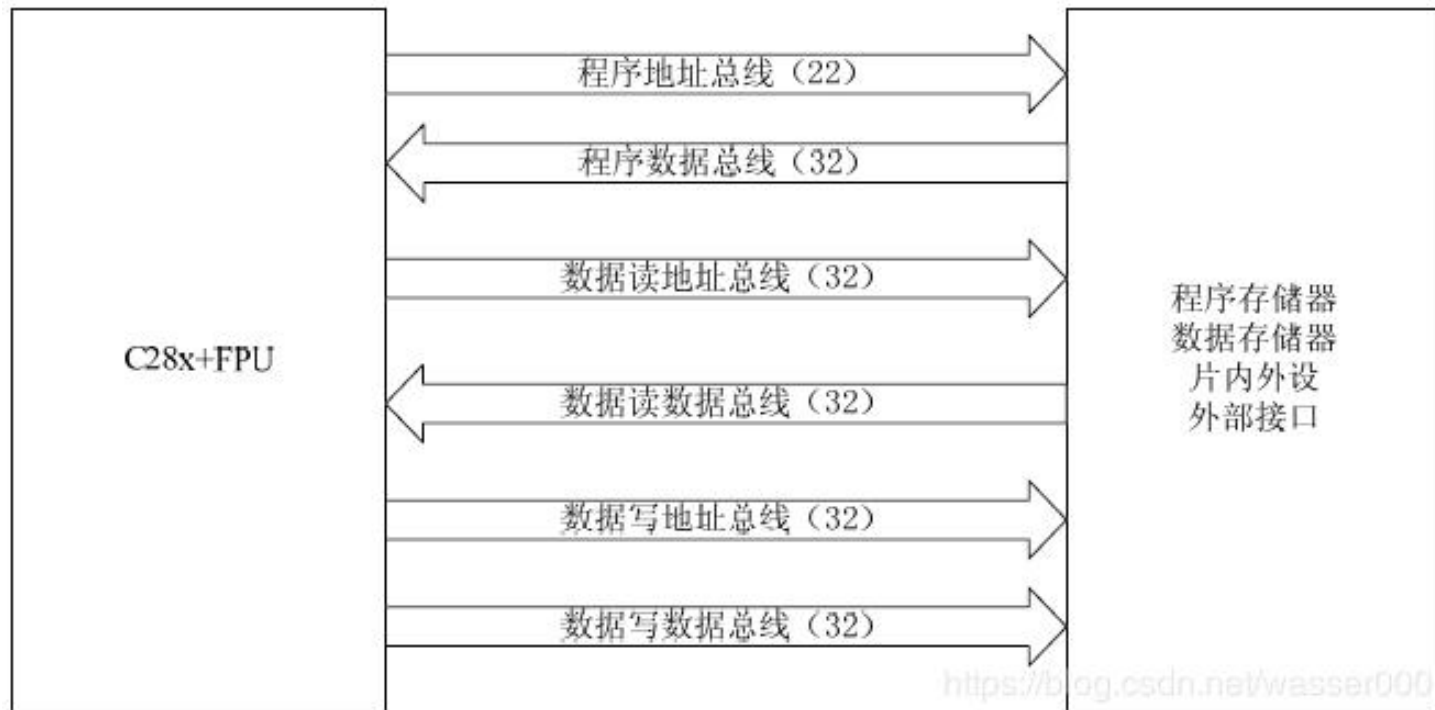
14.改进型哈佛结构（拓展了解）



改进型哈佛结构

DSP28335的哈佛结构

- 程序读总线：22位地址线与32位数据线
- 数据读/写总线：32位地址线与32位数据线



DSP28335的哈佛结构

- 程序读总线：22位地址线与32位数据线
- 数据读/写总线：32位地址线与32位数据线

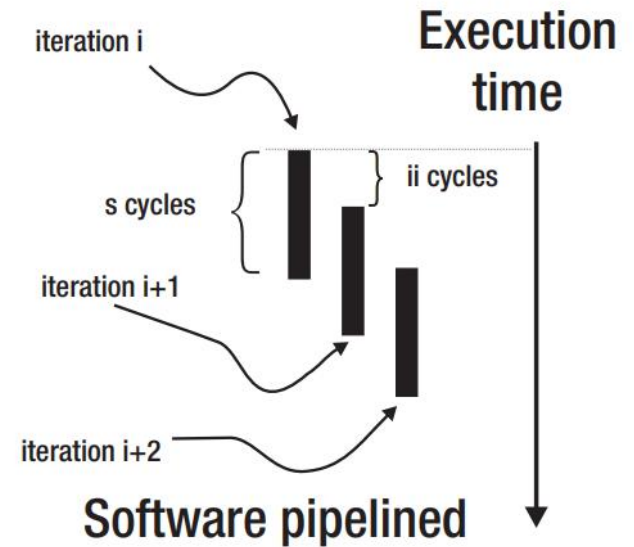
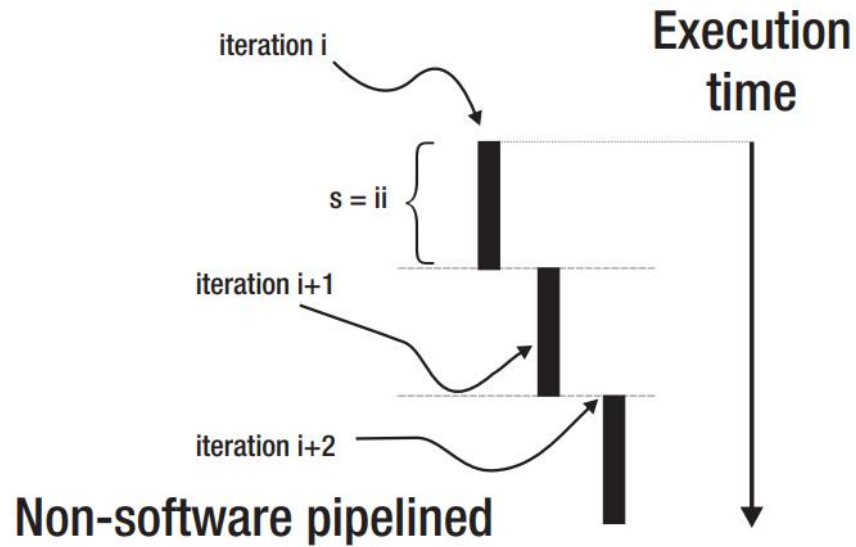
3.2.2 **Memory Bus (Harvard Bus Architecture)**

As with many DSC type devices, multiple busses are used to move data between the memories and peripherals and the CPU. The C28x memory bus architecture contains a program read bus, data read bus and data write bus. The program read bus consists of 22 address lines and 32 data lines. The data read and write busses consist of 32 address lines and 32 data lines each. The 32-bit-wide data busses enable single cycle 32-bit operations. The multiple bus architecture, commonly termed Harvard Bus, enables the C28x to fetch an instruction, read a data value and write a data value in a single cycle. All peripherals and memories attached to the memory bus will prioritize memory accesses. Generally, the priority of memory bus accesses can be summarized as follows:

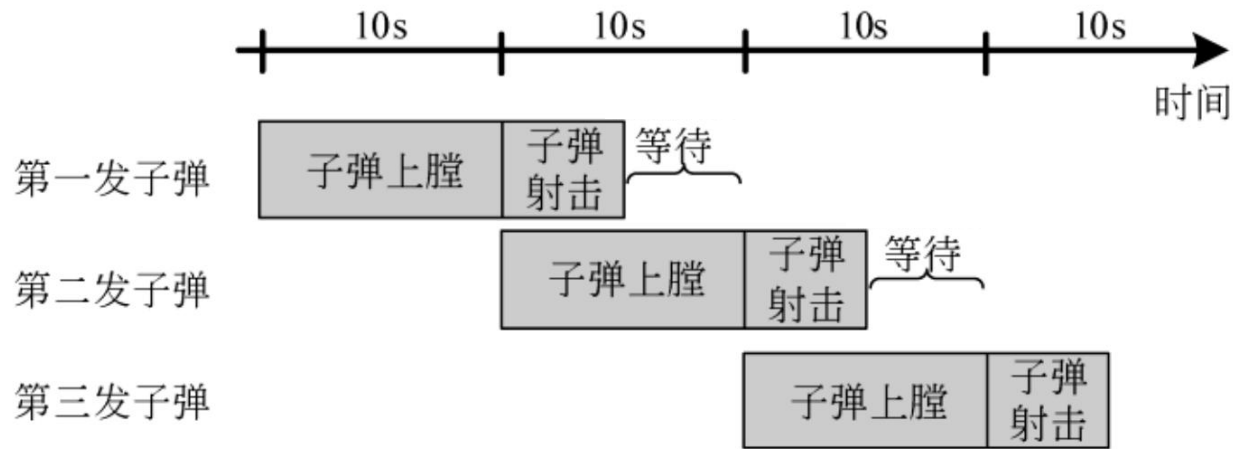
问题

- 哈佛结构，改进型哈佛结构哪个效率高？
- Modified Harvard architecture应该怎么翻译？

15. 流水线

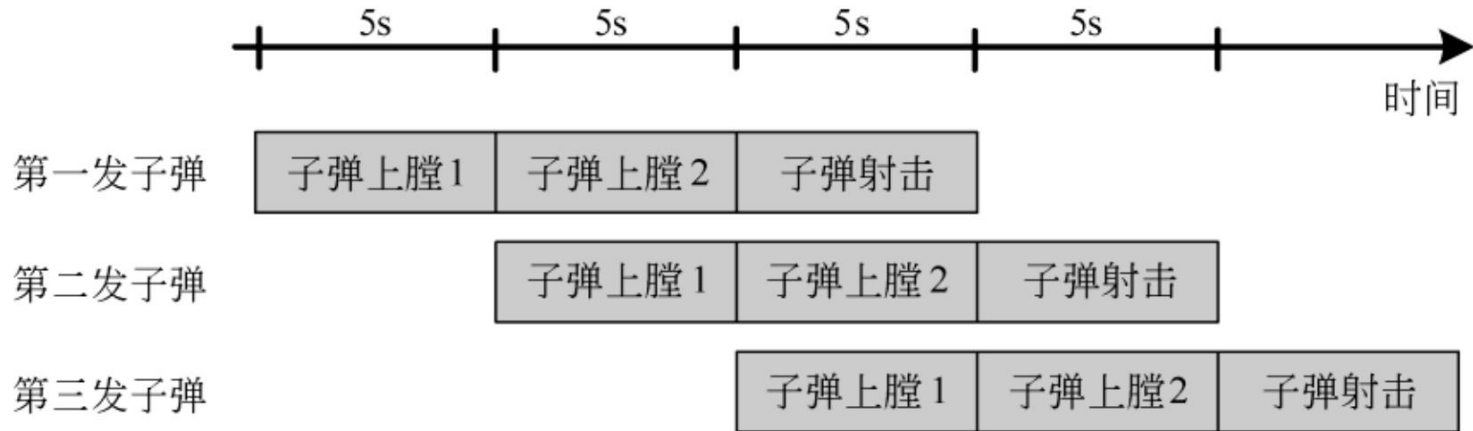


不规则的流水线



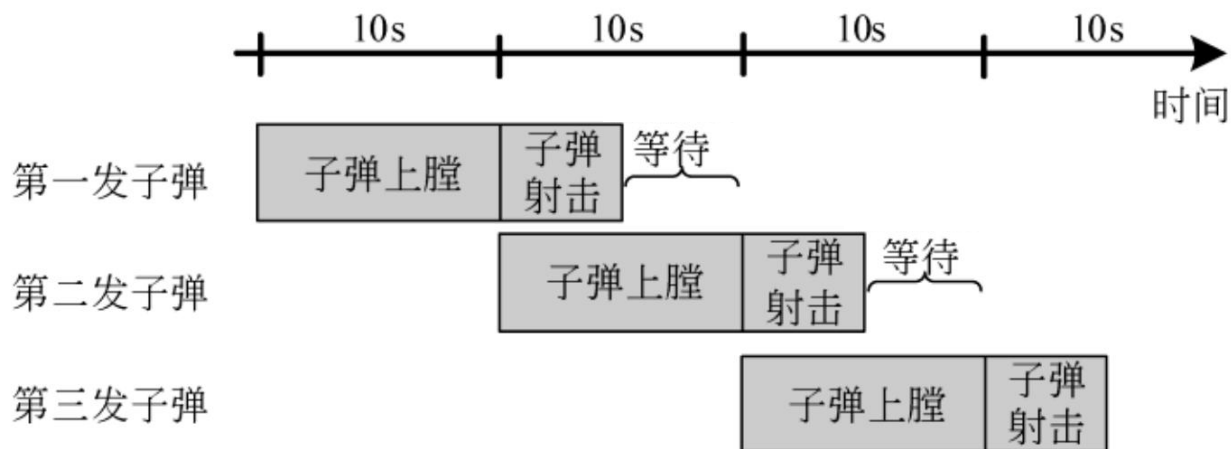
- 并非指令每一步的操作时间都是等长的。
- 长节拍的步骤会导致流水线效率下降（短节拍的步骤要等待长节拍的步骤执行完毕）。

规则的流水线

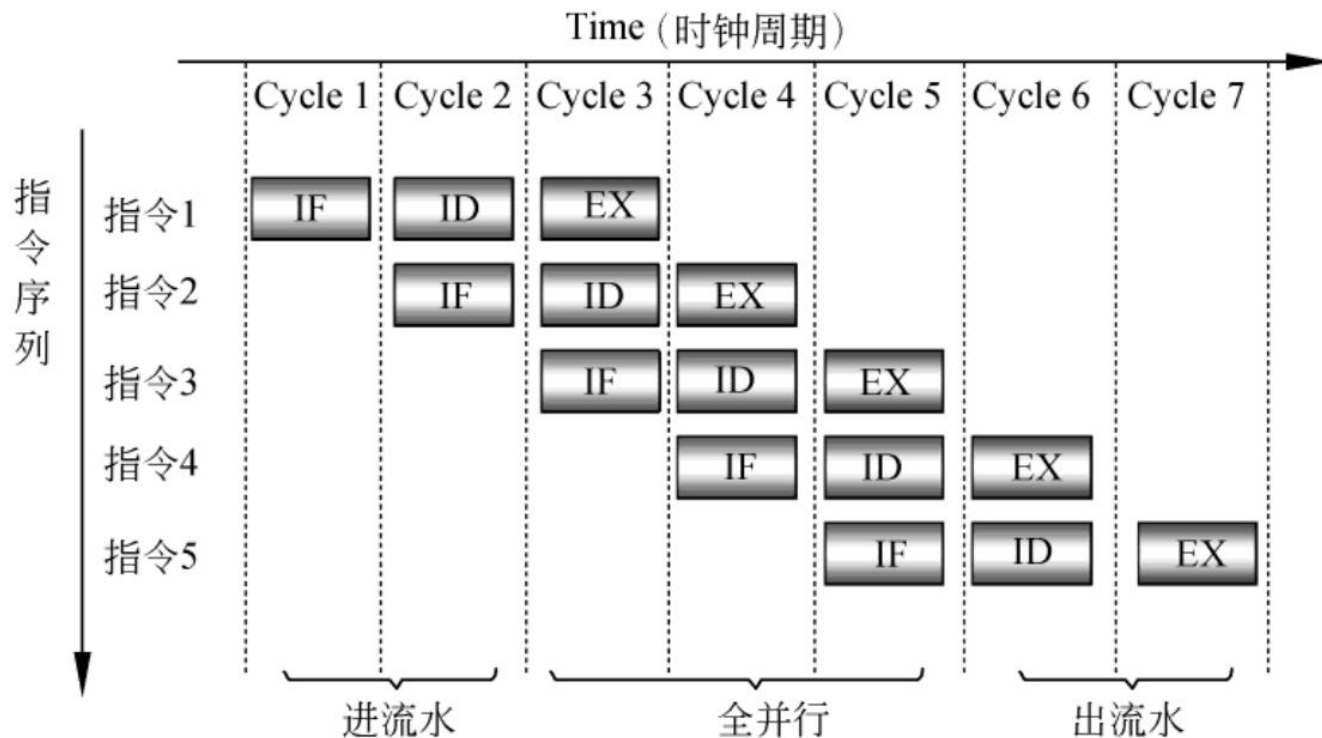


- 方法：拆分长节拍。
- 长节拍的步骤会导致流水线效率下降（短节拍的步骤要等待长节拍的步骤执行完毕）。

不规则 VS 规则



ARM7 三级流水线



流水线节拍	描述
IF	Instruction Fetch, 取指
ID	Instruction Decode, 译码
EX	Execute, 执行

ARM9 五级流水线

指令序列	流水时钟数								
	1	2	3	4	5	6	7	8	9
指令 i	IF	ID	EX	MEM	WB				
指令 i +1		IF	ID	EX	MEM	WB			
指令 i +2			IF	ID	EX	MEM	WB		
指令 i +3				IF	ID	EX	MEM	WB	
指令 i +4					IF	ID	EX	MEM	WB

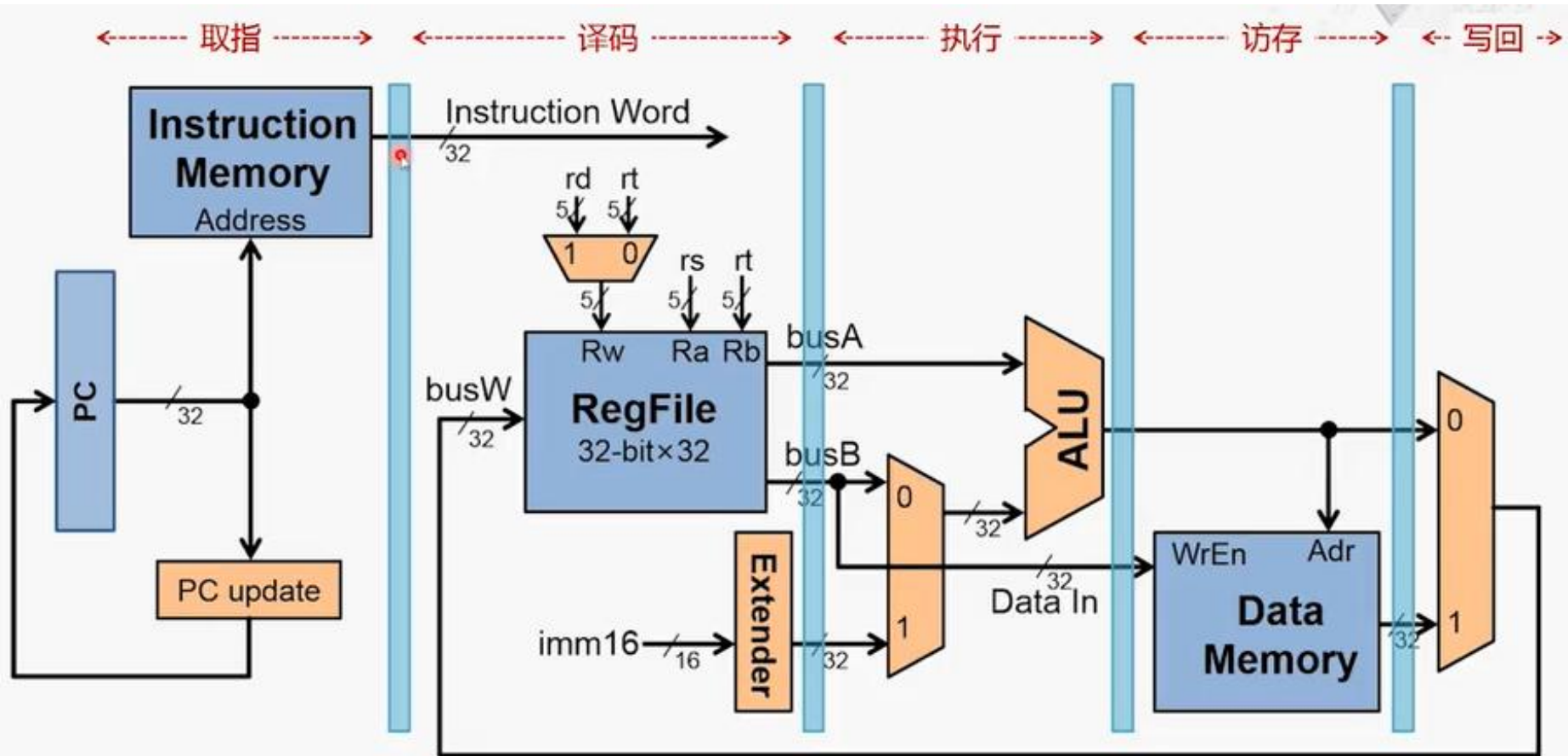
- 横坐标：表示时间，即各个任务或指令在流水线中 所在该时刻所对应的子过程
- 纵坐标：表示某个任务或某条指令，即流水线依次 处理的 任务或指令

ARM9 五级流水线

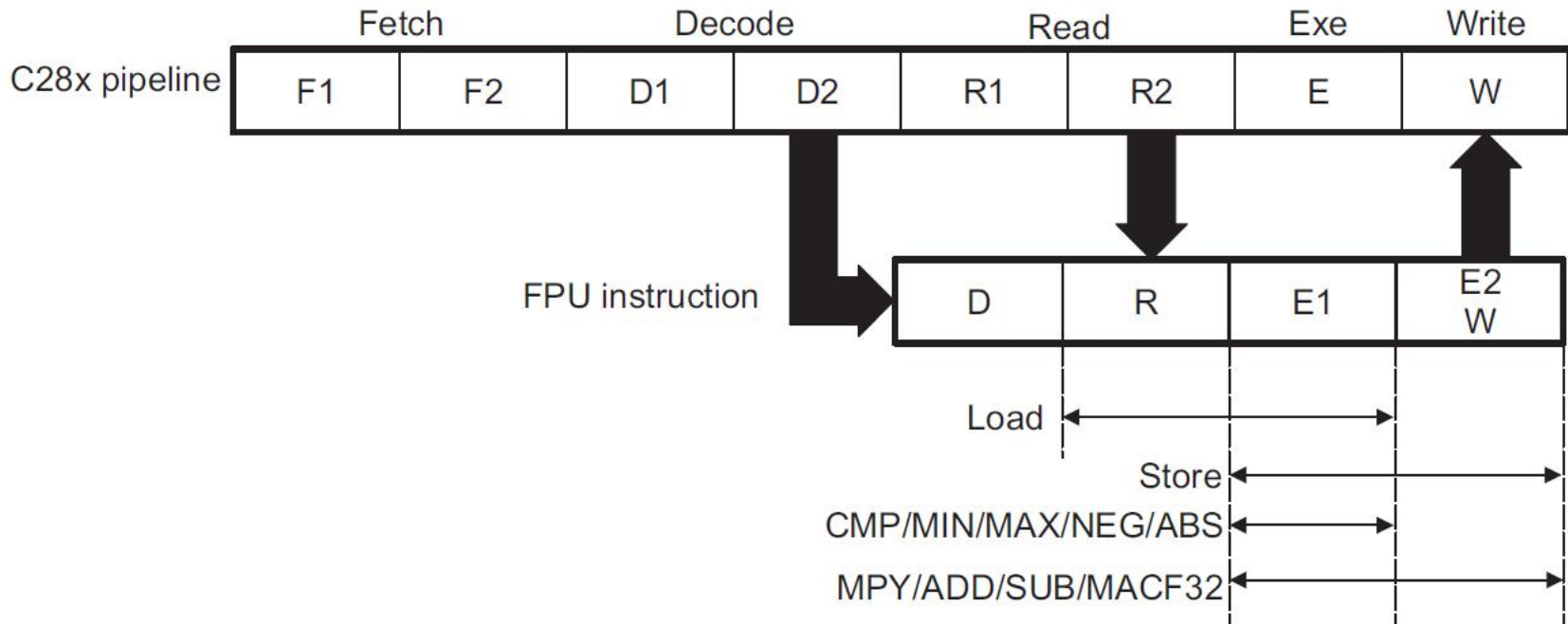
指令序列	流水时钟数								
	1	2	3	4	5	6	7	8	9
指令i	IF	ID	EX	MEM	WB				
指令i +1		IF	ID	EX	MEM	WB			
指令i +2			IF	ID	EX	MEM	WB		
指令i +3				IF	ID	EX	MEM	WB	
指令i +4					IF	ID	EX	MEM	WB

- **IF: Instruction Fetch**,取指令,用到部件:指令存储器
- **ID: Instruction Decode**,译码, 用到部件: 指令译码器
- **EX: Execute**,执行, 用到部件: ALU, 算术逻辑单元。
- **MEM: Memory Access**,访存, 从数据存储器中读。用到部件: 数据存储器。
- **WB: Write Back**,写回, 将数据写到寄存器中。用到部件: 寄存器堆写口。

ARM9 五级流水线



DSP28335 八级流水线



- **F1: Generate instruction address at program bus address lines.**
- **F2: Read the instruction from program bus data lines.**
- **D1: Decode instruction.**
- **D2: Calculate address information for operand(s) of the instruction.**
- **R1: Load operand(s) address to data bus address lines.**
- **R2: Read operand from data bus data lines.**
- **E: Execute the instruction.**
- **W: Write back result to data memory.**

处理器流水线深度的变化

1986年, R2000: 5级

1988年, R3000: 5级

1991年, R4000 (64位): 8级

1993年, Pentium: 5级

取指	译码	地址生成	执行	回写
----	----	------	----	----

1995年, Pentium Pro: 12级

IF1	IF2	IF3	ID1	ID2	RAT	ROB	DIS	EX	WB	RR	RET
-----	-----	-----	-----	-----	-----	-----	-----	----	----	----	-----

1997年, ARM9: 5级

取指	译码	执行	访存	写回
----	----	----	----	----

2002年, ARM11: 8级

取指1	取指2	译码	发射	执行1	执行2	执行3	写回
-----	-----	----	----	-----	-----	-----	----

2004年, Pentium 4(Prescott) : **31级**

2006年, Core 2 Duo(Merom) : 14级

2008年, Core i7(Nehalem) : 16级

2009年, Cortex-A8 : 13级

2010年, Cortex-A9 : 11级

2011年, Cortex-A15 : 15级

2013年, Core i7(Haswell) : 14级

2013年, Cortex-A57 : 15级

问题

- 可以无限的划分流程，而无限的提高效率吗？

谢谢!



群名称: 23年DSP的原理与应用

群 号: 364687084