第1章 计算机基础

> 课件来源：以 2024 版《第一章 计算机基础2.pptx》（74 页，ch1_v2024）为骨架；2022 版《第一章 计算机基础.ppt》（66 页，ch1_v2022）仅补充主版本没有的内容（各代微处理器的特点与应用，2022 版 p.12-16）。本章是全书地基：数制转换、原码/反码/补码与溢出判断是必考计算题，BCD/ASCII、冯·诺依曼结构、三总线、名词术语是填空/选择/简答高频点。课程考核中期末考试占 60%、闭卷（课件 p.4）。

0. 本章考点清单

考点	常见题型	重要度
十进制↔二进制/十六进制转换（除基取余、乘基取整、四位分组）	计算	★★★
原码/反码/补码表示与真值互求、表数范围	计算、填空	★★★
补码加减运算、溢出判断 OV=C7⊕C6、进位与溢出区别	计算、选择	★★★
BCD 码（8421）与 ASCII 码（7 位、128 字符、查表）	填空、选择	★★★
按权展开式求十进制值	计算	★★
二进制四则运算与逻辑运算（AND/OR/NOT/XOR）	计算	★★
冯·诺依曼设计思想、五大部件、存储程序	简答	★★
程序执行过程（取指/执行、IP/IR/ID 微操作）	简答	★★
三总线 AB/DB/CB 的方向与位数含义	填空、选择	★★
微处理器/微型计算机/微型计算机系统三级组成	填空	★★
存储单元地址与内容、读/写操作、容量单位换算	填空	★★
进位 CY 与溢出 OV 的概念辨析	选择	★★
名词术语（位/字/字长/字节/指令/指令系统/程序）与技术指标	填空	★
计算机发展史（ENIAC、四代计算机、微处理器各代）	选择	★
计算机特点、分类、应用、发展趋势	选择	★
摩尔定律、冯·诺依曼体系缺陷	简答	★

1. 章节导引与计算机发展概述

1.1 课程与本章结构（课件 p.1-6）

本课程讲授微型计算机系统组成、8086 CPU 结构、指令系统、汇编语言程序设计、存储器、输入输出和中断（课件 p.3）。考核：平时作业 16% + 阶段测试 8% + 实验 16% + 期末闭卷考试 60%（课件 p.4）。本章三大块：1.1 计算机发展概述、1.2 运算基础、1.3 计算机系统的组成及程序执行过程（课件 p.6）。

1.2 第一台计算机 ENIAC（课件 p.7-8）

- 名称：ENIAC（Electronic Numerical Integrator And Computer，电子数字积分计算机）。
- 诞生：1946 年 2 月 14 日，美国宾夕法尼亚大学（次日正式公布）；研制背景是二战弹道射表计算（课件 p.7）。
- 规模数据：占地约 170 m²、重 30 英吨、耗电 150 kW、造价 48 万美元；17468 根电子管；运算速度每秒 5000 次加法或 400 次乘法（课件 p.8）。

1.3 计算机的四代划分（课件 p.9-11）

代别	年代	器件	内存/外存	软件与速度要点
第一代	1945~1958	电子管	水银延迟线内存、磁鼓外存	机器语言，50 年代中期出现汇编语言；科学计算与军事
第二代	1958~1964	晶体管	磁芯内存、磁盘外存	每秒百万次；高级语言与编译系统；信息处理
第三代	1964~1971	中、小规模集成电路	—	每秒千万次；操作系统、结构化程序设计、计算机网络
第四代	1971 至今	大规模集成电路	—	微型计算机出现，性能价格比大突破（课件 p.10）

80 年代起研制第五代"智能计算机"：打破固有体系结构，具有思维、推理、判断能力（课件 p.11）。

1.4 微型计算机与微处理器的发展（课件 p.12-16）

- 微型计算机诞生于 20 世纪 70 年代；特点：体积小、重量轻、功耗低、可靠性高、价格便宜、使用方便、软件丰富；核心是微处理器（CPU），**每出现一个新的微处理器就产生新一代微型计算机**（课件 p.12）。
- 2024 版各代微处理器（课件 p.13-15）：

代别	年代	代表产品	字长	集成度	时钟	指令周期
第一代	1971-1973	Intel 4004、8008	4 位 / 8 位	约 2000 管/片	1 MHz	20 μs
第二代	1973-1975	Intel 8080、M6800	8 位	5000 管/片	2 MHz	2 μs
第三代	1975-1977	Intel 8085、Z80、M6802	8 位	1 万管/片	2.5-5 MHz	1 μs
第四代	1978-1980	Intel 8086、Z8000、M6809	16 位	3 万管/片	5 MHz	0.5 μs
第五代	1980 以后	（16 位）；1983 后 80386、68020	16 位 →32 位	10 万→15-50 万管/片	10→16 MHz	0.2→0.1 μs
第六代	2005 至今	酷睿 Core 2 Duo(2006)、Core i7(2008,45nm 四核)、二代 i3/i5/i7(2010)	—	—	—	—

新一代还采用超标量流水线、高速缓存等新技术（课件 p.14）；未来继续向多核 CPU 发展（课件 p.16）。

- **2022 版补充**（主版本没有，2022 版 p.12-16）：各代特点与应用——第一代指令系统简单、运算功能较差、价格低廉，面向家电、计算器和二次仪表；第二代（8080/8085、MC6800、Z80、R6502）指令系统比较完善、速度提高一个数量级、寻址能力增强，面向家电、智能仪表、工业控制；第三代（8086/8088、80186、80286、Z8000、MC68000，16 位）指令系统丰富、多级中断、多种寻址方式、段式存储结构，用于工业控制；第四代（80386、80486，32 位）内存 1 MB 以上、32 位总线、多任务多用户，用于办公自动化、网络环境；第五代（Pentium、Itanium，64 位）外部数据线 64 位、地址总线 32 位以上、多媒体与通信能力，用于办公自动化、网络服务器。
- 注意：两版"代"的划分口径不同（2024 版把 8086 算第四代，2022 版把 8086/80286 算第三代、386/486 算第四代），答题时按所用课件口径，不要混记。

1.5 特点、分类、应用、趋势（课件 p.17-21）

- 主要特点五个：自动性、高速性、逻辑性、通用性、准确性（课件 p.17）。
- 分类（课件 p.18-19）：从原理上分数字计算机、模拟计算机；从用途上分专用、通用；从字长上分 4/8/16/32/64 位机；从结构上分位片机、单片机、单板机、微机系统。结构分类定义：位片机只有一位；单片机在一块芯片中包含运算器、控制器、存储器和 I/O 接口；单板机把 CPU、存储器、I/O 接口、键盘及显示电路做在一块印刷线路板上；微机系统以运算器和控制器为核心，加大规模集成电路存储器、I/O 接口、系统总线，再配外设、电源、辅助电路和软件（课件 p.19）。
- 应用：航空航天、科学研究、家用电器等（课件 p.20）。
- 发展趋势：①两极化（巨型机代表国家经济实力与科学水平，微型机标志科学普及程度）；②多媒体技术；③网络化；④智能化；⑤非冯·诺依曼体系结构，由集中顺序控制变为并行处理（课件 p.21）。

易错警示：单片机≠单板机≠微机系统，包含关系是"芯片级→板级→系统级"；"每新一代微处理器→新一代微机"是常考判断句。

2. 进位计数制与数制转换

2.1 概念精讲（课件 p.22-25）

按进位方法计数称为进位计数制。R 进制数有两个特点：有 R 个数字符号 0~R-1；逢 R 进一。R 称基数，R^i 称位权（权）。按权展开式：

```
S = a(n-1)a(n-2)…a1a0.a(-1)a(-2)…a(-m)
  = a(n-1)×R^(n-1) + … + a1×R¹ + a0×R⁰ + a(-1)×R⁻¹ + … + a(-m)×R⁻ᵐ
```

- 十进制：0-9，逢十进一。例 $(758.75)_{10} = 7\times10^2 + 5\times10^1 + 8\times10^0 + 7\times10^{-1} + 5\times10^{-2}$（课件 p.23）。
- 二进制：0、1，逢二进一。例 $(10110.101)_2 = 1\times2^4 + 0\times2^3 + 1\times2^2 + 1\times2^1 + 0\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3} = (22.625)_{10}$（课件 p.24）。
- 十六进制：0-9、A-F，逢十六进一。例 $(1AE.4)_{16} = 1\times16^2 + 10\times16^1 + 14\times16^0 + 4\times16^{-1} = (430.25)_{10}$。课件 p.25 原文写作 $(1AF.4)_{16} = 1\times16^2 + 10\times16^1 + 15\times16^0 + 4\times16^{-1} = (430.25)_{10}$，**课件原文如此，存疑：按展开系数 1、10、15、4 求和应为 431.25；对照例 1-2（p.29）与 p.31，该数应为 1AE.4H = 430.25**（2022 版 p.24 同误）。

2.2 三种数制对照表（表 1-1，课件 p.26）

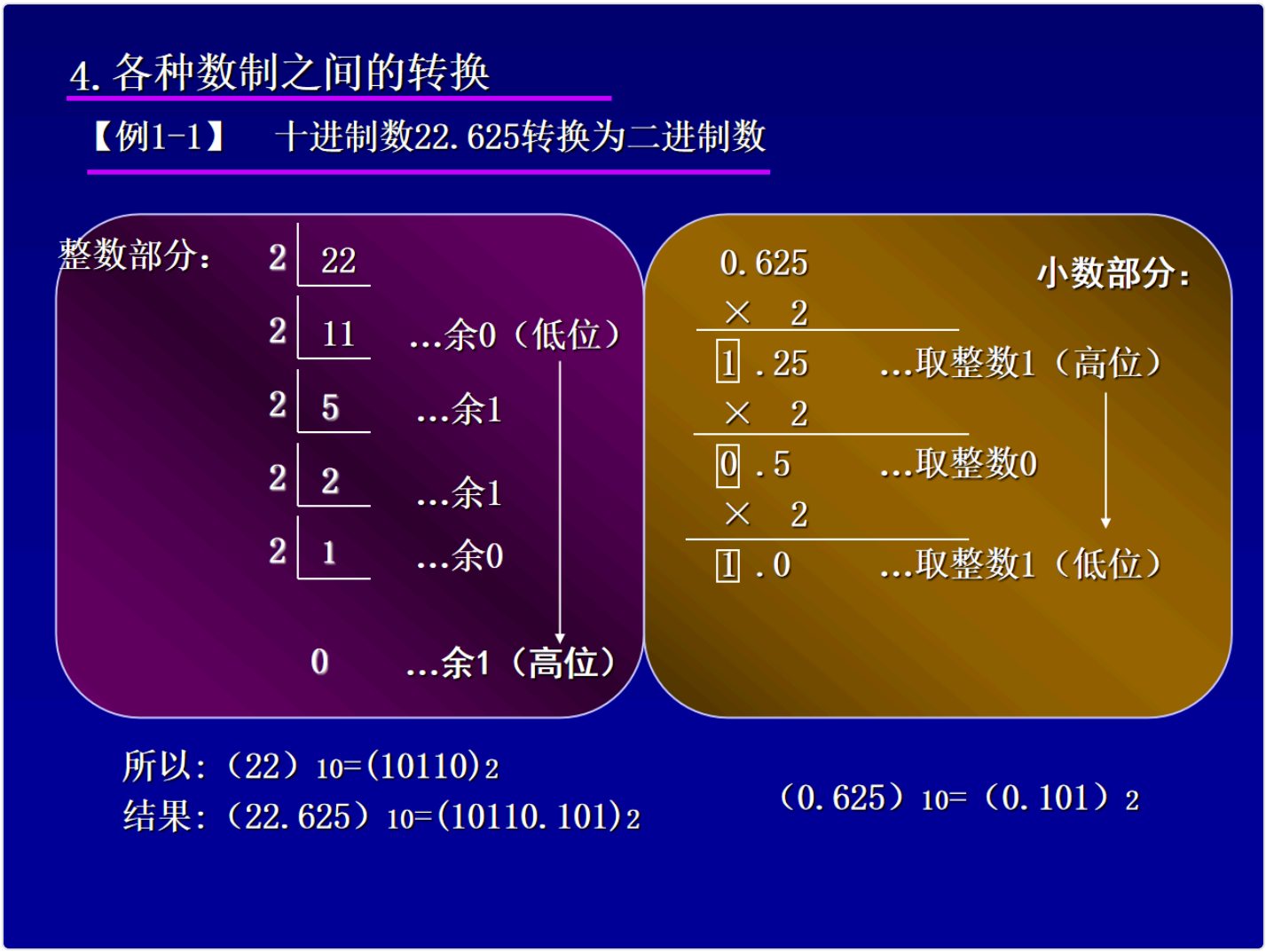
十进制	二进制	十六进制	十进制	二进制	十六进制
0	0000	0	9	1001	9
1	0001	1	10	1010	A
2	0010	2	11	1011	B
3	0011	3	12	1100	C
4	0100	4	13	1101	D
5	0101	5	14	1110	E
6	0110	6	15	1111	F
7	0111	7	16	10000	10
8	1000	8	17	10001	11

2.3 转换方法（课件 p.27-31）

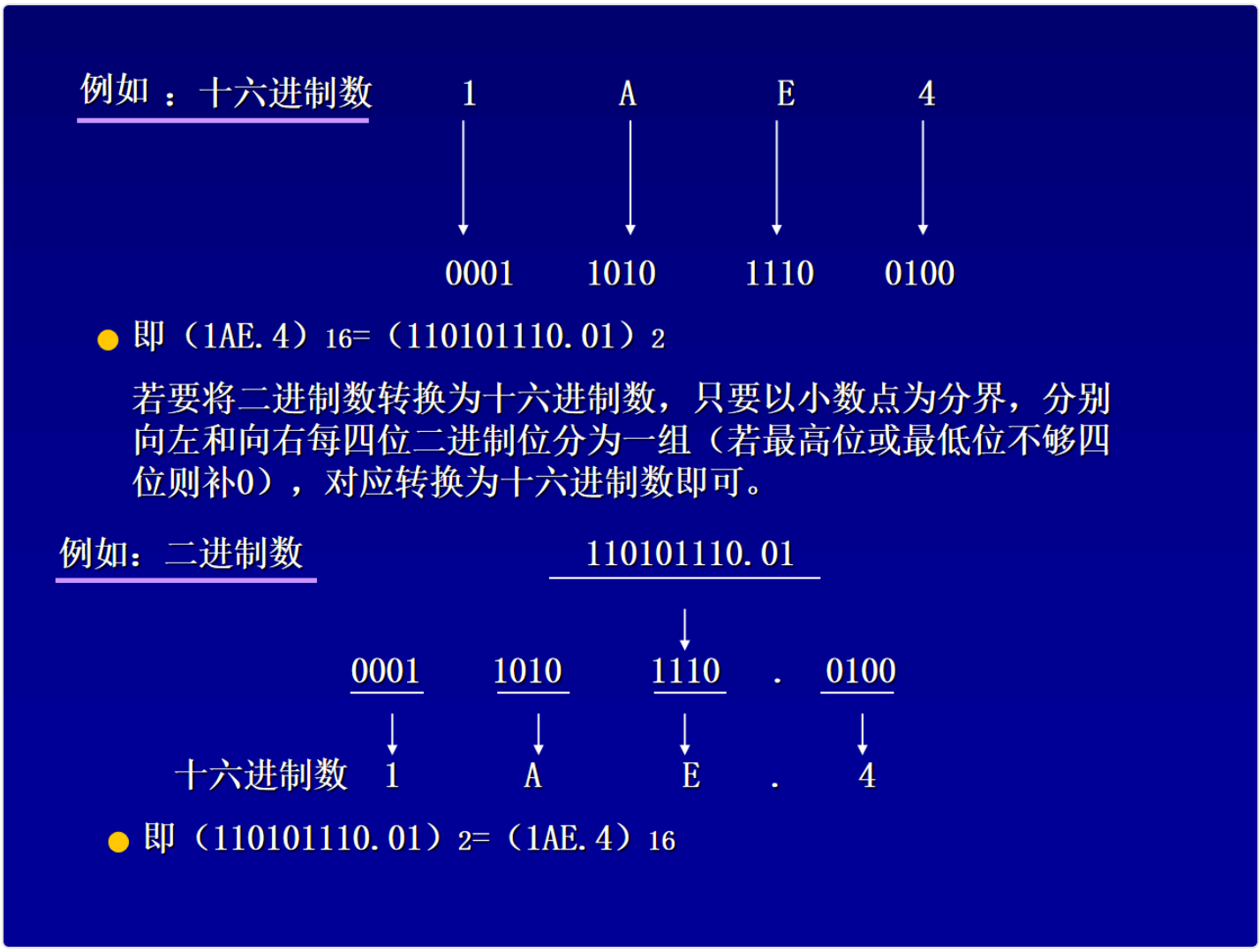
1. 二/十六进制 → 十进制：按权展开求和。

2. 十进制 → 二/十六进制：整数部分**除基取余**——不断除以基数直至商为 0，每次取余数，**从低位排向高位**（先取得的余数是低位）；小数部分**乘基取整**——用基数乘小数部分直至小数为 0 或达到精度，每次取整数，**从高位排向低位**（课件 p.27）。
3. 注意（课件 p.30）：整数部分每次只求整数商；小数部分如 0.3 永远乘不到 0，只能按精度取位数，结果不精确。
4. 二进制 ↔ 十六进制：以小数点为界，向左、向右每 4 位一组，不足 4 位补 0（整数部分向左补、小数部分向右补），一组对一位十六进制（课件 p.31）。

【例 1-1】$(22.625)_{10}$ → 二进制：整数 22 除 2 取余依次 0,1,1,0,1（低→高）得 10110；小数 0.625×2=1.25 取 1、0.25×2=0.5 取 0、0.5×2=1.0 取 1（高→低）得 .101；结果 $(10110.101)_2$（课件 p.28）。



【例 1-2】$(430.25)_{10}$ → 十六进制：430÷16=26 余 14→E，26÷16=1 余 10→A，1÷16=0 余 1，得 1AE；0.25×16=4.0 取 4；结果 $(1AE.4)_{16}$（课件 p.29）。

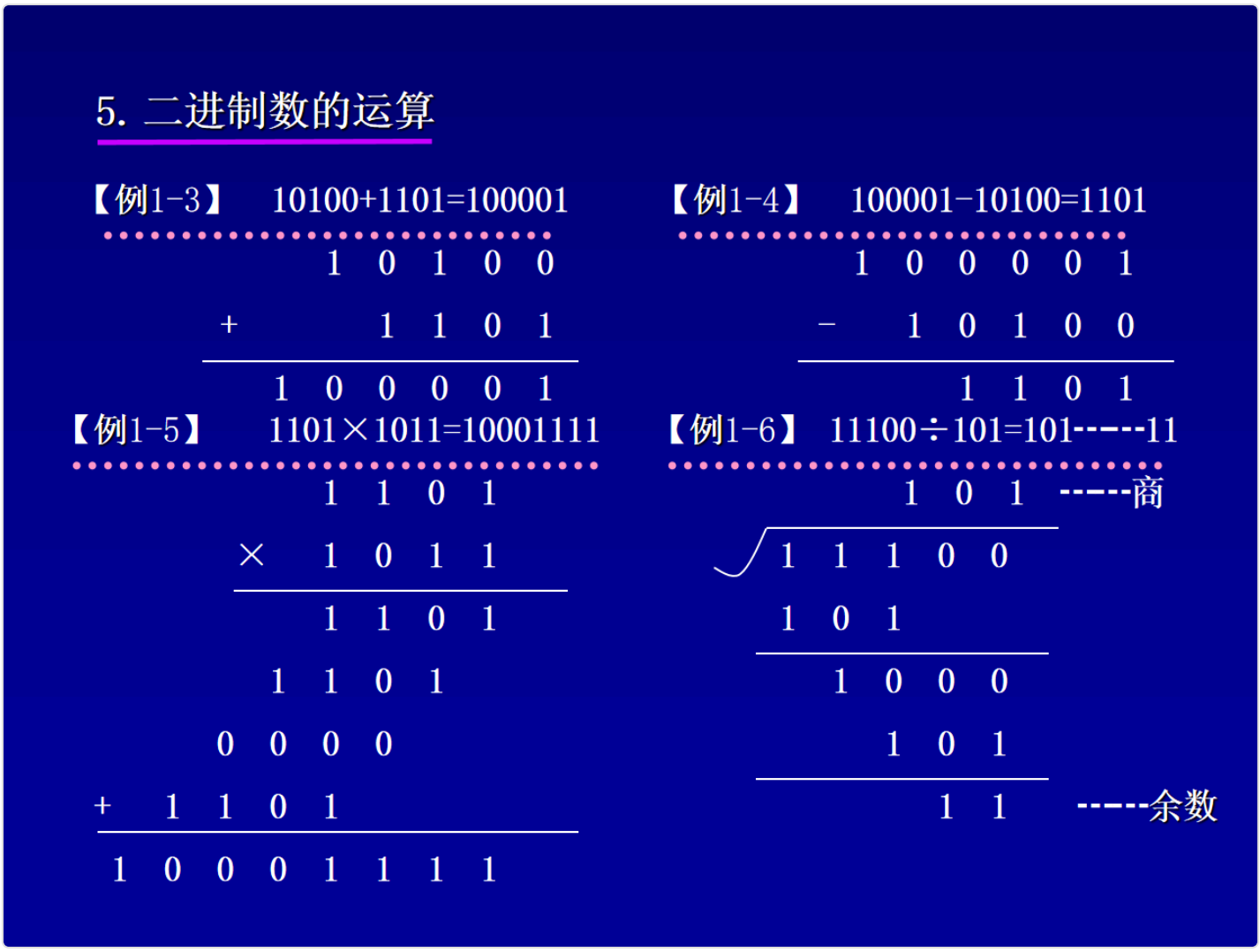


易错警示：余数"倒着读"、整数"顺着读"方向相反，最易丢分；分组补 0 只能补在整数部分的最左端和小数部分的最右端，不能在小数末尾随意补 0 改变数值分组。

3. 二进制数的运算与逻辑运算

3.1 四则运算（课件 p.32，例 1-3~1-6）

- 加法：10100 + 1101 = 100001（逢二进一）。
- 减法：100001 - 10100 = 1101（借一当二）。
- 乘法：1101 × 1011 = 10001111（部分积错位相加）。
- 除法：11100 ÷ 101 = 101 ..... 11（商 101、余 11）。



验算（十进制）：20+13=33、33-20=13、13×11=143、28÷5=5 余 3，与二进制结果一致。

3.2 逻辑运算（课件 p.33-34，例 1-7~1-10）

按位运算、位间无进位借位：AND 全 1 才 1；OR 有 1 即 1；NOT 逐位取反；XOR 相同为 0、不同为 1。

例题	算式	结果
例 1-7	10100101 AND 10001011	10000001
例 1-8	10100101 OR 10001011	10101111
例 1-9	NOT 10100101	01011010
例 1-10	10100101 XOR 10001011	00101110

易错警示：逻辑运算逐位独立，不要套用算术加法的进位；XOR 常用于判异或溢出（见第 6 节）。

4. 二进制编码：BCD 码与 ASCII 码

4.1 BCD 码（课件 p.35-36）

BCD 码是"二进制编码的十进制数"：本质是十进制（10 个符号、逢十进位），但每一位十进制数字用 4 位二进制编码表示。例：65 → 0110 0101；BCD 01001001.0111 → 49.7（课件 p.35）。

关键结论：BCD 码与二进制之间不能直接转换，必须以十进制为中间桥梁（BCD→十进制→二进制，反之亦然）（课件 p.35）。

表 1-2 8421 BCD 编码表（课件 p.36）：

十进制	8421BCD	十进制	8421BCD
0	0000	6	0110
1	0001	7	0111
2	0010	8	1000
3	0011	9	1001
4	0100	10	0001 0000
5	0101	11	0001 0001

8421 码只使用 0000~1001 十个码组，1010~1111 六个码组不用（由表 1-2 直接可得）。

4.2 ASCII 码与汉字编码（课件 p.37-38）

- ASCII（American Standard Code for Information Interchange，美国标准信息交换码）采用 **7 位二进制编码，共 $2^7 = 128$ 个字符**，含数码 0-9、英文字母及可打印字符（课件 p.37）。
- 课件 p.38 给出 ASCII 表：左列为 4 位"行"码（低 4 位 b3b2b1b0），顶行为 3 位"列"码（高 3 位 b6b5b4），字符编码 = 列码拼接行码。例：'A' 位于行 0001、列 100 → 100 0001B = 41H；'0' 行 0000、列 011 → 011 0000B = 30H；'a' 行 0001、列 110 → 61H。

行 列	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	、	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	”	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	’	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	¦
1101	CR	GS	—	=	M	]	m	}
1110	SO	RS	•	>	N	↑	n	~
1111	SI	US	/	?	O	_	o	DEL

- 汉字编码采用国标码 GB18030-2000，单、双、四字节混合编码，"每个字节的最高位为 1"以区分汉字和 ASCII 码（课件 p.37）。（补充常识：单字节部分与 ASCII 兼容、最高位为 0，"最高位为 1"指汉字编码字节。）

易错警示：ASCII 是 7 位编码、128 个字符，不是 8 位 256 个；在机内占 1 字节时最高位补 0（补充常识）。BCD 的 0110 0101 是 65 的 BCD，而 65 的二进制是 1000001，两者完全不同。

5. 带符号数的表示：原码、反码、补码

5.1 机器数与真值（课件 p.39）

机器数：数在计算机中的二进制表示形式，位数通常为 8 的倍数，最高位作符号位（0 表示 +、1 表示 -）。真值：机器数代表的实际数值。例：X1=+84=+1010100B → [X1]机=01010100B；X2=-84 → [X2]机=11010100B（课件 p.39）。带符号数有三种表示：原码、反码、补码（课件 p.40）。

5.2 原码（课件 p.40-41）

定义（n 为位数）：X≥+0 时 [X]原 = X；X≤-0 时 [X]原 = 2^{n-1} - X。即符号位 0/1、数值位不变。 例：8 位时 [+124]原 = 01111100，[-124]原 = 11111100；D7 为符号位、D6~D0 为数值位；16 位时 D15 为符号位（课件 p.41）。 范围：8 位原码 -127~+127；16 位原码 -32767~+32767（课件 p.41）。0 有 +0、-0 两种原码。

5.3 反码（课件 p.42-43）

定义：X≥+0 时 [X]反 = X；X≤-0 时 [X]反 = 2^n - 1 + X。即正数反码同原码；负数反码 = 原码除符号位外按位取反。 特点：[+0]反=00000000、[-0]反=11111111（0 有两种表示）；8 位范围 -127~+127，16 位 -32767~+32767；最高位为符号位（课件 p.42）。 例（课件 p.43）：+4 → 00000100；-4：原 10000100 → 反 11111011；-31：原 10011111 → 反 11100000。已知反码求真值：[X]反=00000101 → X=+5；[Y]反 =1111110 → Y=-1。**课件原文如此，存疑：1111110 只有 7 位，按本章 8 位约定应写 11111110，结论 Y=-1 不变**（2022 版 p.40 此处另有 9 位原码笔误，2024 版已改正）。

5.4 补码（课件 p.44-46）

- 模的概念：n 位二进制计数范围 0~2^n-1，加 2^n 或减 2^n 结果不变，2^n 即模；钟表模为 12，8+10=6 (mod 12) 与 8-2=6 (mod 12) 等价——**减一个数等于加它的补数**，这是补码把减法变加法的原理（课件 p.44）。
- 定义：[X]补 = (2^n×i + X) MOD 2^n；X≥+0 时 [X]补 = X；X≤-0 时 [X]补 = 2^n + X = [X]反 + 1，即负数原码除符号位外求反加 1（课件 p.45）。
- 求法例：[+3]补=00000011；[-3]补=[-3]反+1=11111100+1=11111101；[+0]补=00000000；[-0]补 =11111111+1=00000000（进位丢弃）→ **0 的补码唯一**（课件 p.45）。
- 特点（课件 p.46）：8 位补码真值范围 **-128~+127**，16 位 **-32768~+32767**；符号位为 0 时其余位即真值的二进制；符号位为 1 时其余位**不是**真值的二进制，需"按位取反、末位加 1"才得绝对值。
- 减法变加法：[X±Y]补 = [X]补 + [±Y]补。例 X=64-9=64+(-9)：[64]补=01000000B，[-9]补 =11110111B，相加得 1 00110111，丢弃模 2^8 的进位后 00110111B=+55（课件 p.46）。

5.5 三码对照（表 1-3，课件 p.47）

十进制	原码	反码	补码
-128	无	无	10000000
-127	11111111	10000000	10000001
-126	11111110	10000001	10000010
-2	10000010	11111101	11111110
-1	10000001	11111110	11111111
-0	10000000	11111111	00000000
+0	00000000	00000000	00000000
+1	00000001	00000001	00000001
+127	01111111	01111111	01111111

表1-3　数的表示方法

十进制数	二进制数	原码	反码	补码
−128	−10000000	————	————	10000000
−127	−1111111	11111111	10000000	10000001
−126	−1111110	11111110	10000001	10000010
…	……	……	……	……
−2	−0000010	10000010	11111101	11111110
−1	−0000001	10000001	11111110	11111111
−0	−0000000	10000000	11111111	00000000
+0	+0000000	00000000	00000000	00000000
+1	+0000001	00000001	00000001	00000001
+2	+0000010	00000010	00000010	00000010
…	……	……	……	……
+126	+1111110	01111110	01111110	01111110
+127	+1111111	01111111	01111111	01111111

易错警示：①原码/反码 0 有两种、补码 0 唯一，故补码多表示一个 -128（10000000 是补码的 -128，不是 -0）；②负数补码求真值必须"取反加 1"，直接读数值位必错；③正数三码相同。

6. 补码运算的溢出及判断

6.1 溢出概念（课件 p.48）

溢出指运算结果超出规定长度的表数范围，此处特指带符号补码运算溢出。n 位补码范围 $-2^{n-1} \sim +2^{n-1}-1$。8 位例：60+100=160>+127 溢出；(-60)+(-100)=-160<-128 溢出；60+(-100)=-40 不溢出（课件 p.48）。

6.2 判断方法（课件 p.49）

1. 符号判断法：两正数相加得负、两负数相加得正 → 溢出；正负相加永不溢出。
2. 双进位异或法：**OV = C7 XOR C6**（C7 为符号位产生的进位、C6 为次高位向符号位的进位）。OV=1 溢出，OV=0 无溢出。
3. 进位与溢出的区别：进位 CY 指最高位向更高位的进位（CY=C7）；**有进位不一定有溢出，无进位不一定无溢出**，二者是不同概念（课件 p.49）。例：64-9 的补码加法 CY=1 但 OV=0；60+100 CY=0 但 OV=1。

易错警示：溢出针对带符号数；无符号数运算看进位 CY 不看 OV（补充常识）。丢弃模进位是补码运算的正常操作，不等于溢出。

7. 数的定点和浮点表示（大纲列目，补充常识）

两版课件目录与小结均列有"1.2.4 数的定点和浮点表示"（课件 p.6、p.74），但均无展开正文页，考试概率低。衔接常识（补充常识）：定点数约定小数点位置固定（纯整数或纯小数）；浮点数形如 $N = M \times R^E$，由尾数 M 和阶码 E 两部分组成，小数点位置随阶码浮动，表数范围大但精度由尾数位数决定。复习以课件实际展开内容为准。

8. 计算机硬件系统组成及程序执行过程

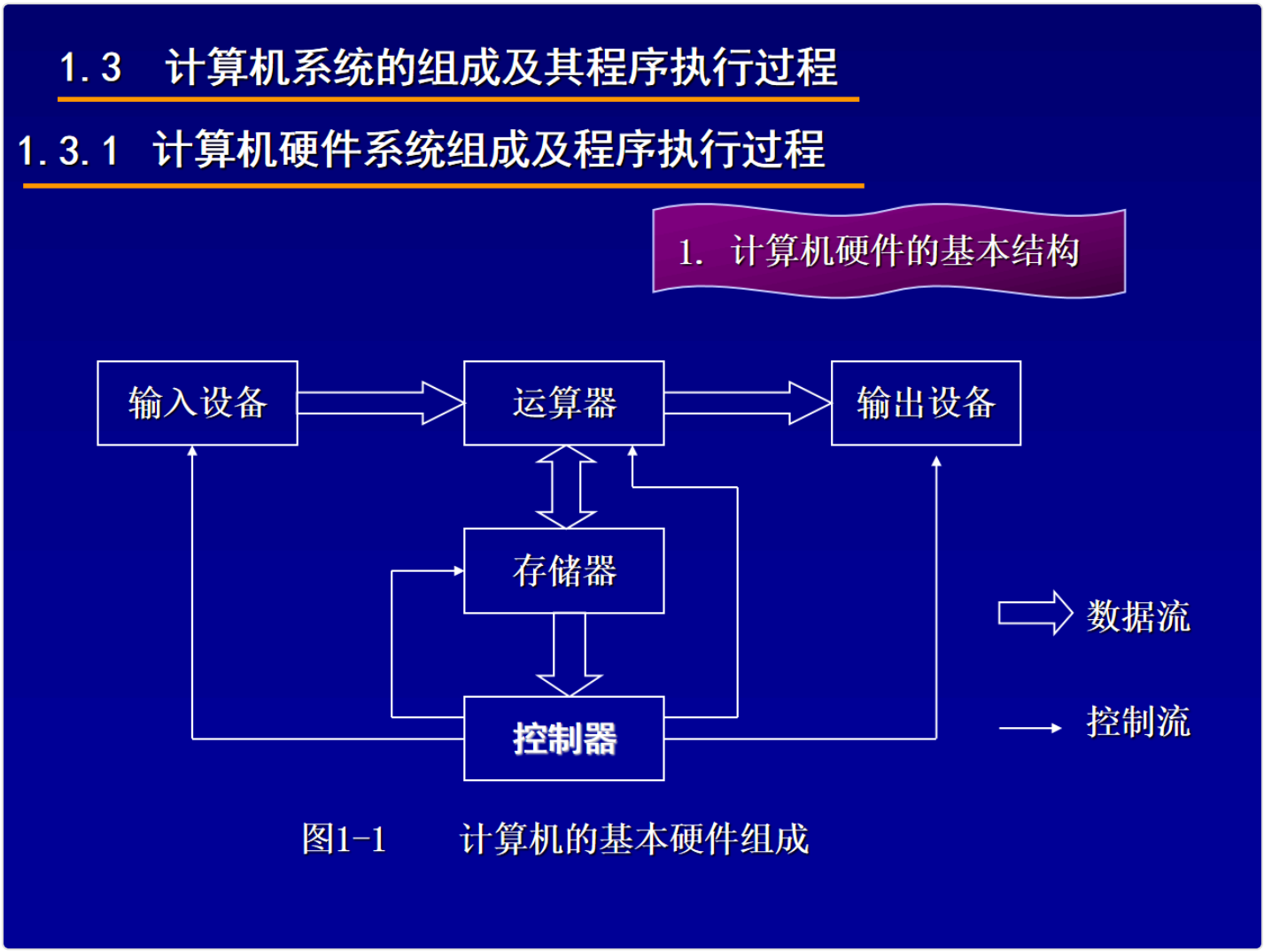
8.1 冯•诺依曼设计思想（课件 p.50）

三条：①采用二进制表示指令和数据；②程序（指令序列）和数据预先存入存储器，执行时自动、连续地逐条取出指令执行；③硬件由运算器、控制器、存储器、输入设备、输出设备五大部分组成。核心是"**存储程序**"和"**程序控制**"，即"集中顺序控制"；目前大多数计算机仍遵从该体系（课件 p.50）。

8.2 摩尔定律与体系缺陷（课件 p.51-52）

- 摩尔定律：每隔 18 个月 IC 集成度翻一倍、性能提升一倍、价格降低一半；已持续 40 多年。两个客观限制：光速（信号传播延迟、频率过高信号畸变）与材料原子特性（晶体管不能小于一个硅原子）（课件 p.51）。
- 冯·诺依曼体系缺陷五点（课件 p.52）：①指令数据同存一个存储器，过分依赖存储器（存储器瓶颈）；②指令按执行顺序存放、由 PC 指明地址，串行执行影响速度；③存储器按地址线性编址，与高级语言按名字访问变量存在"冯·诺依曼语义间隔"；④为算术逻辑运算而生，非数值处理发展缓慢；⑤控制驱动方式，指令数据出错机器不会主动修改，难以应对模糊信息。

8.3 五大部件与数据流/控制流（课件 p.53-54）



数据流：输入设备→运算器→输出设备，存储器与运算器双向交换；控制流：控制器向各部件发控制信号（图 1-1，课件 p.53-54）。构成微型计算机的关键是把各部件有机连接起来（课件 p.54）。

8.4 运算器与存储器（课件 p.55-58）

- 运算器：完成加减乘除算术运算与逻辑运算的电路 + 多个寄存器；在控制信号指挥下完成运算、暂存操作数/结果、数据传送（课件 p.55）。
- 存储单元以 8 bit（1 字节）为单位；每个单元有唯一编号即**地址**，从 0 开始顺序编排；地址与内容是两个概念（图 1-2：地址 0 内容 20、地址 1 内容 31、地址 2 内容 -18、地址 n 内容 186）（课件 p.55）。

1.3.1 计算机硬件系统组成及程序执行过程

2. 运算器、存储器和控制器

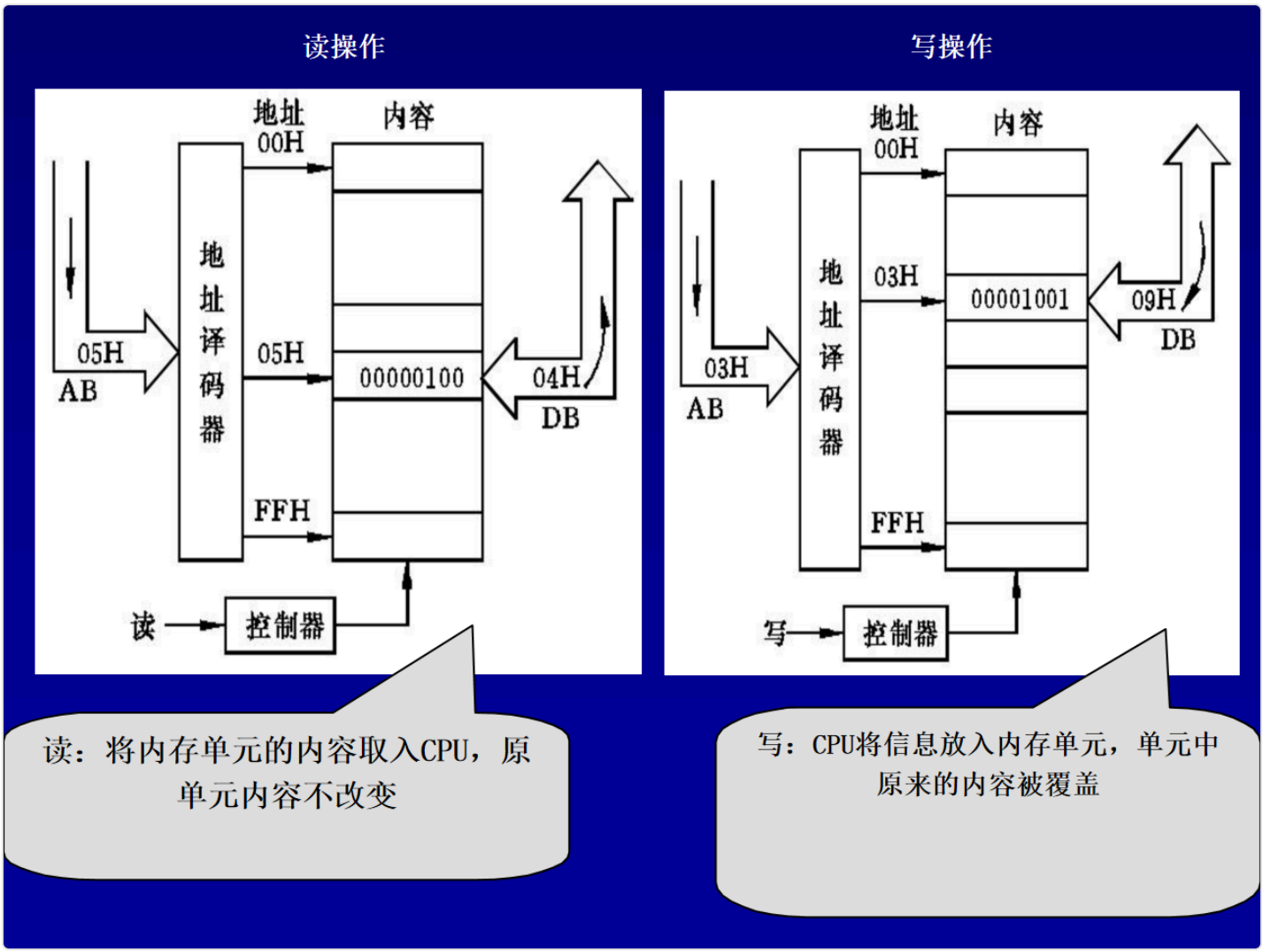
●运算器主要包括能完成加、减、乘、除算术运算及逻辑运算的电路以及多个寄存器。在控制信号的指挥下，运算器完成诸如算术运算、逻辑运算、暂存操作数或运算结果以及数据传送等工作。

地址	内 容
0	20
1	31
2	-18
.	.
.	.
.	.
n	186

图1-2　存储单元的地址与内容

●在微型机中存储单元通常以8bit即1字节为单位。为了能唯一确定并找到任一个存储单元，计算机对每一存储单元都指定一个唯一的编号，称之为存储单元的地址，地址通常从0开始顺序编排。即图1-2所示。

- 存储器指标：存/取周期（存取一个代码的时间，一般 10 ns~50 ns，越快越好）；存储容量 = 存储单元总数量（课件 p.56）。
- 容量单位：字节 Byte（B）= 8 位；1 KB = 1024 B = 2^{10} B；1 MB = 1024 KB = 2^{20} B；1 GB = 1024 MB。例：8 KB 即 64 Kb（B 是字节、b 是位），该 RAM 有 8 K 字节（课件 p.56）。
- 地址与内容实例：地址 08F04H 的单元内容为 10110110B（课件 p.57）。
- 读操作：把内存单元内容取入 CPU，**原单元内容不变**；写操作：CPU 把信息放入内存单元，**原内容被覆盖**（课件 p.58）。图中读：AB 送 05H 经地址译码选中单元，内容 00000100 经 DB 以 04H 送 CPU；写：AB 送 03H、DB 送 09H，单元 03H 内容变为 00001001。



8.5 控制器与简单计算机硬件结构（课件 p.59-60）

控制器组成：指令指针寄存器 IP、指令寄存器 IR、指令译码器 ID、控制信号发生器（课件 p.59）。

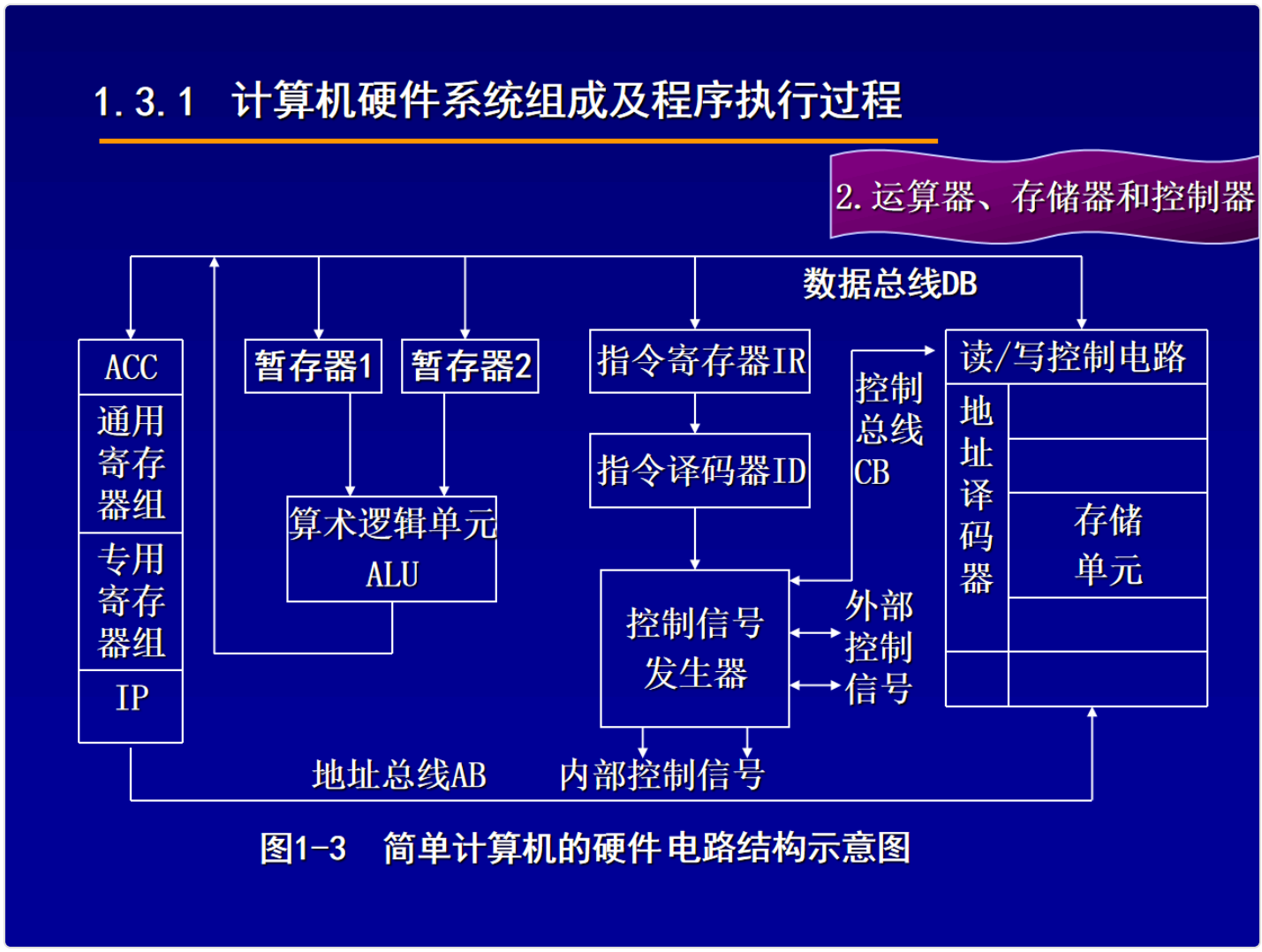
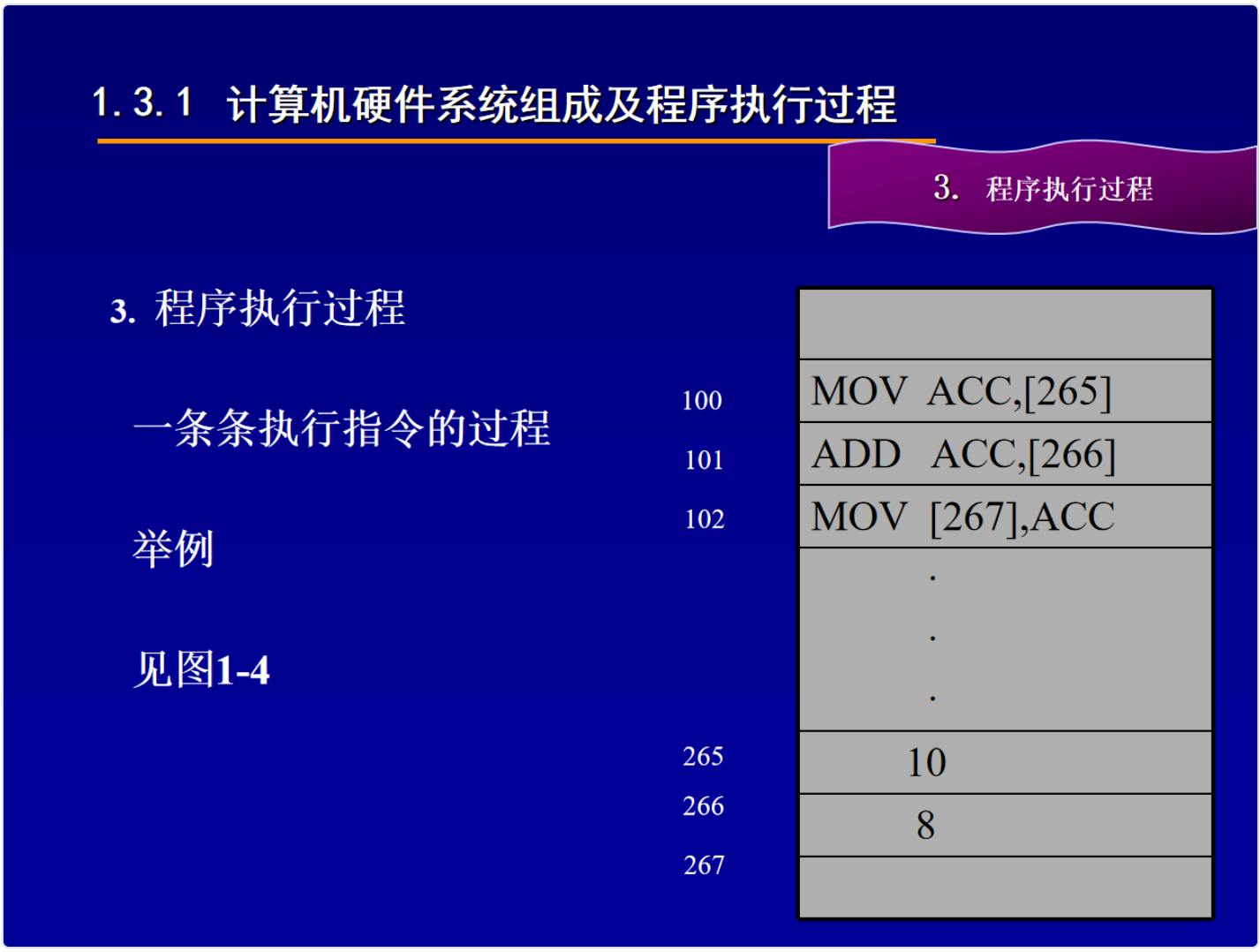


图 1-3 要点：ACC、通用寄存器组、专用寄存器组、IP 挂数据总线 DB；暂存器 1、暂存器 2 送 ALU；IR→ID→控制信号发生器；存储器侧有读/写控制电路、地址译码器、存储单元；IP 经地址总线 AB 送地址；控制信号发生器经控制总线 CB 及内部/外部控制信号指挥全机（课件 p.60）。

8.6 程序执行过程（课件 p.61-62）

程序执行就是一条条执行指令的过程。模型机示例（图 1-4）：地址 100 存 MOV ACC,[265]，101 存 ADD ACC,[266]，102 存 MOV [267],ACC；数据区 265 单元=10、266 单元=8、267 单元待存结果。



每条指令分两个阶段（课件 p.62）： 1. 取指阶段：IP→M（送地址），IP+1→IP，M→IR（取指令入指令寄存器）。 2. 分析执行阶段：IR→ID 译码后执行。 - MOV ACC,[265]：①addr→M；②M→ACC。 - ADD ACC,[266]：①addr→M；②M→暂存器1；③ACC→暂存器2；④ALU 执行 ADD；⑤ALU 结果→ACC。 - MOV [267],ACC（按同一模式推导）：①addr→M；②ACC→M（写操作）。

执行完三条指令后 (267)=10+8=18，IP 由 100 依次增至 103。

8.7 控制器设计（课件 p.63）

按产生微操作控制信号的方式分：传统组合逻辑控制器、微程序控制器。微程序技术原理：把指令执行分解为若干微操作，用微指令把微操作编成微程序，执行微程序即完成微操作、即执行了指令（课件 p.63）。

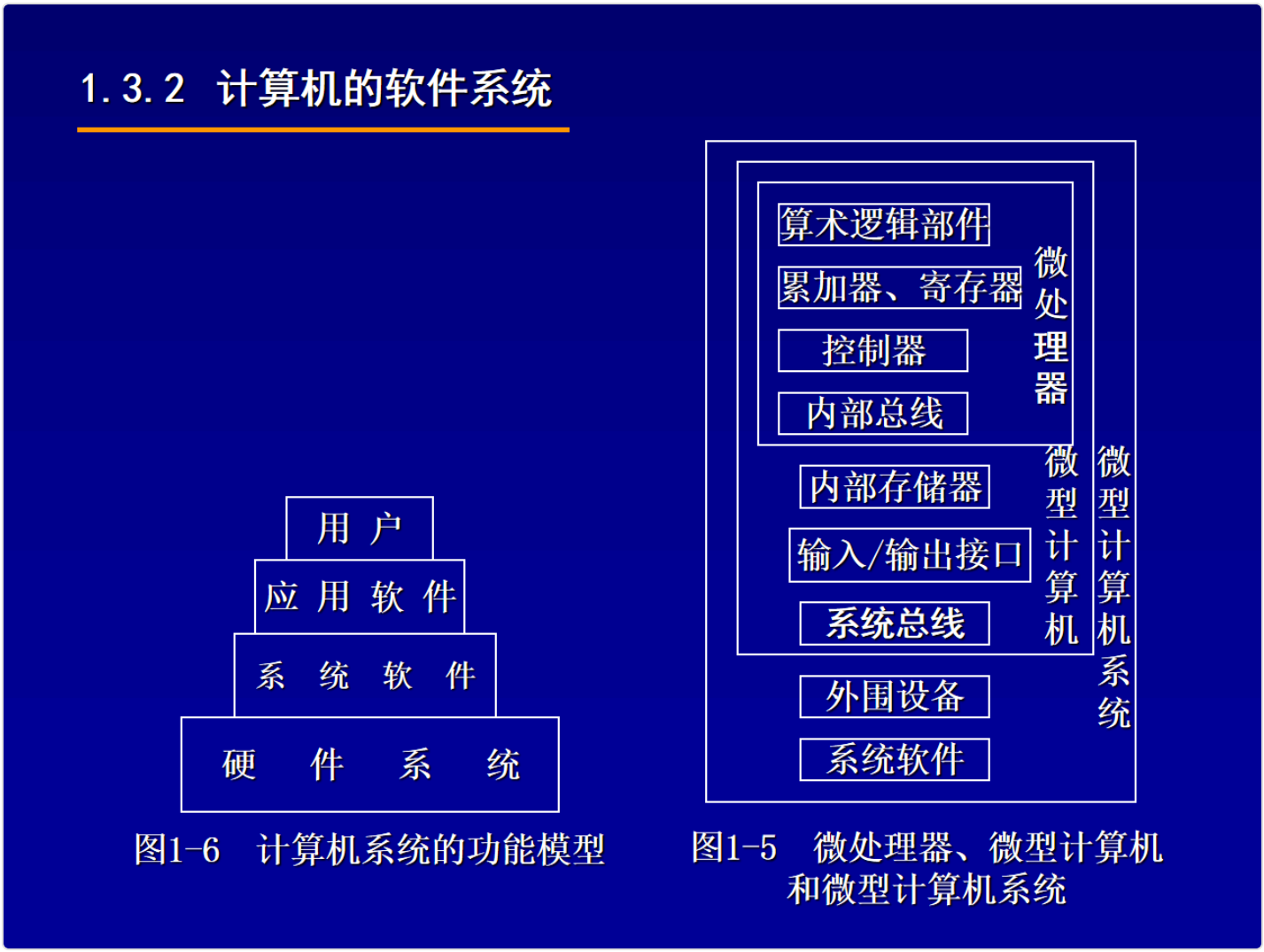
易错警示：取指阶段 IP 自动 +1 指向下一条指令；IR 存"当前指令"、ID 负责译码、IP 存"下一条指令地址"，三者分工常考填空；读不改内容、写覆盖内容。

9. 软件系统与微型计算机系统的三级组成

9.1 系统软件与应用软件（课件 p.64-66）

- 系统软件：由机器设计者提供、使用和管理计算机的软件，包括各种语言的汇编/解释/编译程序、监控管理程序、操作系统、调试程序、故障诊断程序、程序库（课件 p.64）。
- 应用软件：用户用各种语言编制的解决各种问题的软件，如文字处理、财务管理、银行管理软件（课件 p.65）。
- 层次关系（图 1-6）：用户→应用软件→系统软件→硬件系统，软件是用户与硬件的接口，软件之间逐层依赖；硬件是物质基础、软件扩大功能，二者结合才是完整计算机系统（课件 p.65-66）。

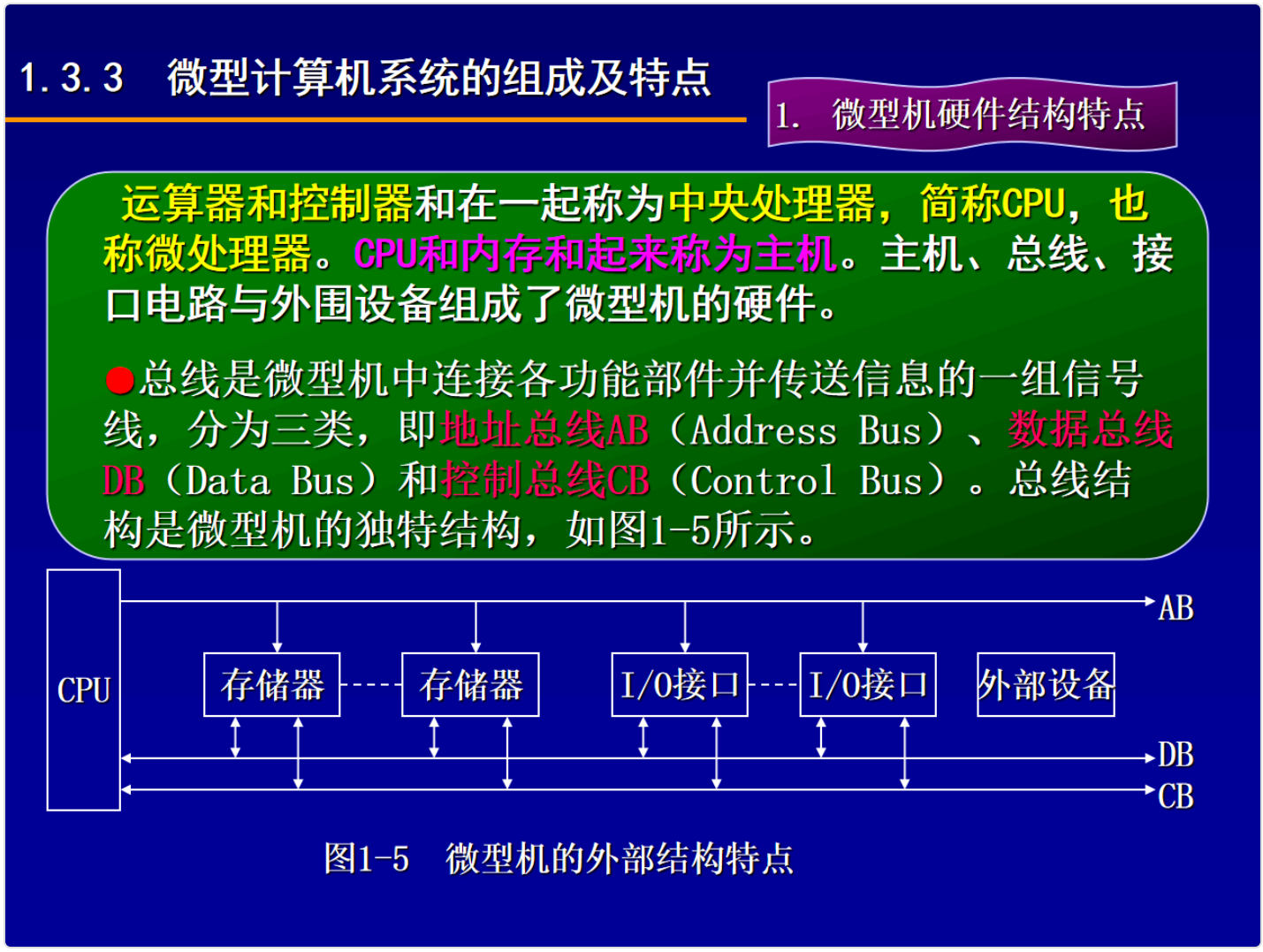
9.2 微处理器、微型计算机、微型计算机系统（课件 p.66-67）



- 微处理器 = 算术逻辑部件 + 累加器/寄存器 + 控制器 + 内部总线。
- 微型计算机 = 微处理器 + 内部存储器 + 输入/输出接口 + 系统总线。
- 微型计算机系统 = 微型计算机 + 外围设备 + 系统软件（图 1-5；课件 p.67 亦概括为微处理器→加内存储器、总线、I/O 接口→再加外部设备与软件）。

9.3 微型机硬件结构特点：总线（课件 p.68-69）

- 运算器 + 控制器 = 中央处理器 CPU（微处理器）；CPU + 内存 = **主机**；主机 + 总线 + 接口电路 + 外围设备 = 微型机硬件（课件 p.68）。
- 总线是连接各功能部件并传送信息的一组信号线，分三类：地址总线 AB（Address Bus）、数据总线 DB（Data Bus）、控制总线 CB（Control Bus）；总线结构是微型机的独特结构（课件 p.68）。



总线	传送内容	方向	位数含义
AB 地址总线	地址信息	单向（CPU 发出）	决定 CPU 可直接寻址的内存范围
DB 数据总线	数据（也可是编码、指令代码、状态量、控制量）	双向	与微处理器位数对应，反映运算与传送能力
CB 控制总线	控制信号	按信号各有方向	—

（课件 p.69）

9.4 微型机硬件/软件系统组成（课件 p.70-71）

硬件系统一般由主机和外围设备组成（课件 p.70）；软件系统分系统软件与应用软件（课件 p.71）。

易错警示：三级组成是"包含"关系不是并列关系；DB 上传的不一定是数值；AB 位数决定寻址范围（如 20 根 AB → 2^{20} = 1 MB），DB 位数才对应字长。

10. 常用名词术语与主要技术指标

10.1 名词术语（课件 p.72）

术语	定义
位 bit	最基本最小的数据单位，只有 0/1 两种状态
字节 Byte	8 位二进制数
字	作为一个整体参与运算、处理和传送的一串二进制数，信息的基本单位
字长	每个字包含的二进制位数；通常等于数据总线位数和通用寄存器位数
指令	指挥计算机进行基本操作的命令
指令系统	计算机所能执行的全部指令的集合
程序	完成某一任务的指令（或语句）的有序集合

10.2 主要技术指标（课件 p.73）

字长、内存储器容量、运算速度、外部设备配备（课件 p.73）。

易错警示：字长 = DB 位数 = 通用寄存器位数，与 AB 位数无关；"字"不等于"字节"，字长随机器而定。

11. 易错点与高频陷阱汇总

1. BCD 码与二进制不能直接按权转换，必须经十进制桥梁（课件 p.35）。
2. 除基取余的余数从低位往高位读（倒序），乘基取整的整数从高位往低位读（顺序），方向相反（课件 p.27-28）。
3. 二↔十六分组补 0：整数部分补在最左、小数部分补在最右，分组以小数点为界（课件 p.31）。
4. 0.3 等小数乘基取整永远乘不尽，只能按精度截断，结果不精确（课件 p.30）。
5. 原码、反码的 0 有 ±0 两种表示，补码 0 唯一；因此 8 位补码比原码/反码多表示 -128（课件 p.42、p.45-47）。
6. 10000000 作为补码是 -128，作为原码/反码无定义（课件 p.47）。
7. 负数补码求真值要"数值位取反加 1"，直接读数值位必错（课件 p.46）。
8. 补码加法丢弃符号位进位是模运算的正常操作，丢弃进位≠溢出（课件 p.45-46、p.49）。

9. CY≠OV：OV=C7⊕C6；64-9 有进位无溢出，60+100 无进位有溢出（课件 p.49）。
10. 溢出只针对带符号补码运算；正数加负数永不溢出（课件 p.48-49）。
11. ASCII 是 7 位编码 128 字符，不是 8 位 256 字符（课件 p.37）。
12. 8 KB = 64 Kb：大写 B 是字节、小写 b 是位（课件 p.56）。
13. 地址与内容不同：08F04H 是地址、10110110B 是该地址单元的内容（课件 p.57）。
14. 读操作不改变单元内容，写操作覆盖原内容（课件 p.58）。
15. AB 单向、DB 双向；AB 位数定寻址范围、DB 位数定字长（课件 p.68-69、p.72）。
16. CPU=运算器+控制器，主机=CPU+内存，不要漏"内存"或把外设算进主机（课件 p.68）。
17. 微处理器⊂微型计算机⊂微型计算机系统，三级是嵌套包含关系（课件 p.66-67）。
18. 课件 p.25 的 1AF.4H=430.25 有误（应为 431.25，或数应为 1AE.4H）；p.43 的 [Y]反=1111110 少写一位（8 位应为 11111110），考试按正确规则计算。

12. 典型例题精讲

例 A（数制转换综合）：把 $(22.625)_{10}$ 依次转换为二进制、十六进制和 8421BCD 码

解：1. 整数 22 除 2 取余：22÷2=11 余 0；11÷2=5 余 1；5÷2=2 余 1；2÷2=1 余 0；1÷2=0 余 1。余数倒读 → 10110。 2. 小数 0.625 乘 2 取整：0.625×2=1.25 取 1；0.25×2=0.5 取 0；0.5×2=1.0 取 1。整数顺读 → 101。故 $(22.625)_{10}=(10110.101)_{2}$。 3. 四位分组：整数向左补 0 → 0001 0110；小数向右补 0 → 1010。得 $(16.A)_{16}$。验算：1×16+6+10/16=22.625 ✓。 4. BCD 码按十进制每位独立编码：2→0010、2→0010、6→0110、2→0010、5→0101，即 (0010 0010.0110 0010 0101)BCD。注意 BCD 不是把 10110.101 分组，而是对十进制数字 2、2、6、2、5 编码。

例 B（补码加法与溢出）：8 位补码计算 64-9 与 60+100，给出 CY、OV 并判断

解：1. 64-9：[64]补=01000000；9=00001001 → [-9]原=10001001 → [-9]反=11110110 → [-9]补=11110111。 相加：01000000+11110111=1 00110111。丢弃进位 → 00110111=+55 ✓。 进位：C7=1、C6=1 → OV=1⊕1=0 无溢出；CY=C7=1。**有进位但无溢出**。 2. 60+100：[60]补=00111100；[100]补=01100100。 相加：00111100+01100100=10100000。符号位为 1（负），两正相加得负 → 溢出。 进位：C7=0、C6=1 → OV=0⊕1=1 溢出；CY=0。**无进位但有溢出**。结果 10100000 若误读为 -96 即错，真值 160 超出 +127。

例 C（程序执行微操作）：写出模型机执行 MOV ACC,[265] / ADD ACC,[266] / MOV [267],ACC 的全过程（(265)=10、(266)=8）

解：初始 IP=100。 1. 取指：IP(100)→M 选单元，IP+1→IP(=101)，M→IR（IR=MOV ACC,[265]）；执行：IR→ID 译码；265→M，M→ACC，ACC=10。 2. 取指：IP(101)→M，IP+1→IP(=102)，M→IR（ADD ACC,[266]）；执行：译码；266→M，M→暂存器1(=8)；ACC→暂存器2(=10)；ALU 相加；结果→ACC，ACC=18。 3. 取指：IP(102)→M，IP+1→IP(=103)，M→IR（MOV [267],ACC）；执行：译码；267→M，ACC→M（写），(267)=18。 结论：存储器 267 单元最终为 18；IP 最终 103；全程体现"存储程序、程序控制"。

附：本章速查表

2 的幂：$2^8=256$、$2^9=512$、$2^{10}=1024$、$2^{11}=2048$、$2^{12}=4096$、$2^{16}=65536$、$2^{20}=1048576$（1 M）。 **16 的权**：$16^0=1$、$16^1=16$、$16^2=256$、$16^3=4096$。

表数范围：

码制	8 位	16 位	0 的表示
原码	-127~+127	-32767~+32767	两种
反码	-127~+127	-32767~+32767	两种
补码	-128~+127	-32768~+32767	唯一

n 位补码范围：-2^{n-1} ~ $+2^{n-1}-1$。

求码口诀：正数三码相同；负数原码=符号 1+绝对值；反码=原码数值位取反；补码=反码+1；补码求真值（负）=数值位取反+1。 **运算公式**：$[X\pm Y]补=[X]补+[\pm Y]补$（mod 2^n，丢进位）；$OV=C7\oplus C6$；$CY=C7$。 **转换口诀**：二/十六→十按权展开；十→二/十六整数除基取余倒读、小数乘基取整顺读；二↔十六四位一组补 0。

常用 ASCII（课件 p.38 表查得）：'0'~'9'=30H~39H；'A'~'Z'=41H~5AH；'a'~'z'=61H~7AH；空格 SP=20H；回车 CR=0DH；换行 LF=0AH；DEL=7FH。ASCII 共 7 位、128 字符。

8421BCD：0~9 → 0000~1001；10 → 0001 0000；每位十进制数独立占 4 位。

容量单位：1 B=8 b；1 KB=2^{10} B；1 MB=2^{20} B；1 GB=1024 MB；8 KB=64 Kb。

总线：AB 单向、位数→寻址范围（2^n）；DB 双向、位数→字长；CB 传控制信号。

组成链：运算器+控制器=CPU；CPU+内存=主机；主机+总线+接口+外设=微机硬件；微处理器+内存+I/O 接口+系统总线=微型计算机；再加外设+软件=微型计算机系统。

冯•诺依曼三思想：二进制表示指令数据；存储程序、自动连续执行；五大部件。核心：存储程序+程序控制。

发展史数字：ENIAC 1946 年、17468 电子管、5000 次加法/秒；四代器件：电子管→晶体管→中小规模 IC→大规模 IC；摩尔定律 18 个月翻番。