

- 冯诺依曼模型：以运算器为核心，必须顺序执行每一条指令，指令执行分为 取出指令，分析指令，执行指令（指令地址存储于 PC）。
- 机器数编码：

原码	-(2^(n-1)-1) ~ 2^(n-1)			
		00...0	01...1	
	11...1	10...0		
反码	-(2^(n-1)-1) ~ 2^(n-1)			
		00...0	01...1	
	10...0	11...1		
补码	-2^(n-1) ~ 2^(n-1)			
	10...0	11...1	00...0	01...1
	-2^(n-1)	-1	0	2^(n-1)
移码	-2^(n-1) ~ 2^(n-1)			
	00...0	01...1	10...0	11...1
	-2^(n-1)	-1	0	2^(n-1)

- 8421 码：以四位二进制数代表一位十进制数，并在末尾用 1100，1101 表示+和-。
- 原码乘法：乘数的每位与被乘数相乘并移位相加。
- 浮点数加减：调整至阶数相同，将尾数相加减，规格化后进行舍入（原码：0 舍 1 入，补码：正数（0 舍 1 入），负数（舍（-1，-0.5]，入（-0.5，0））。

浮点数表示：

十进制->二进制->左/右规格至 0.1...->用浮点数表示

阶符	阶值	尾数符	尾数值
左/右规格	规格次数	与原数相同	规格后数值

- 用 ALU 比较数的大小：将两数进行加减运算，并观察借进位，0 标志。
- 存储器：

顺序存取：SAM（磁带）	只读存取：ROM（光盘）
随机存取：RAM（内存）	直接存取：DAM（磁盘）

- CPU->主存（Cache 地址转化->Cache）-> 辅存 程序访问局部性原理
- 64K \* 16b -> 字扩展(串联)8 \* 8K \* 16b -> 8 \* ( 位扩展（并联）4 \* (8K \* 4b))。
- CPU 引脚：地址(log64K)引脚, 数据 (log16b) 引脚, 状态/控制引脚 CS(RD,WR,IO,MEM)
- Cache：

管理区	数据区
-----	-----

有效位	标记位	状态位	缓存块
-----	-----	-----	-----

- Cache 地址映射：直接映射，全相连，组相连
- 组相连：

Cache：

群号	群内块号	块内地址
----	------	------

主存：

组号	组内行号	块内地址
----	------	------

群块号=组号，块映射到组后组内查全表，块内地址相等，tag=群号

13. Cache 替换算法：RAND，先进先出，LRU（最近最少使用）

LRU：设置计数器，初始为 1，每次更新时若未被访问则+1，否则置 0，选取计数器值最大的行进行替换。

14. Cache 写策略

全写法：写入主存同时写入 Cache。T 命中=Tmem，T 缺失=0；

写回法：修改时写在 Cache 中并记录脏位，Cache 被替换时脏位为 1 的行写入主存。

Cache 行管理信息=有效位+标记+计数器位（ALU）+脏位 T 命中=Tcache，T 缺失>=Tmem