

图 A-14-1 4 个 4 位数加法的传统行波进位和进位保留加法器。加法器细节参见图的左侧，单独信号用小写表示，相应的高层模块在图右，组合信号用大写表示。注意，4 个 n 位数的和可以取 $n+2$ 位

- A.33** [10] <A.6> 有时我们想要对一组数进行相加。假设用户想使用 1 位全加法器对 4 个 4 位数 (A , B , E , F) 进行相加。现在先暂时忽略超前进位。将 1 位加法器按图 A-14-1 上方的组织形式连接起来。传统组织形式下面是一个完全加法器的全新组织形式。尝试使用这两种组织形式对 4 个数做加法，并确保能得到相同结果。
- A.34** [5] <A.6> 首先，如图 A-14-1 所示，画出 16 位进位保留加法器的组织结构，用来实现 16 个部分积相加。假设通过一个 1 位加法器的时间延迟是 $2T$ 。计算上下两个组织结构对 4 个 4 位数做加法的时间。
- A.35** [5] <A.8> 通常，你会期望得到一个时序图，其中包含对数据输入 D 和时钟输入 C 发生的变化描述（分别如图 A-8-3 和图 A-8-6 所示）， D 锁存器和 D 触发器的输出波形 (Q) 之间会有差异。用一两句话描述二者输出波形之间不存在任何差异的情况（例如，输入需要满足的性质）。
- A.36** [5] <A.8> 图 A-8-8 说明了 RISC-V 数据通路的寄存器堆的实现。假设要构建一个新的寄存器堆，但只有两个寄存器和一个读端口，并且每个寄存器只能存储 2 位数据。重画图 A-8-8，使图中的每条线仅对应 1 位数据（与图 A-8-8 中的图不同，其中一些线为 5 位，一些线为 32 位）。使用 D 触发器重画寄存器。无须描绘如何实现 D 触发器或多选器。
- A.37** [10] <A.10> 朋友希望你建立一个“电子眼”用作仿安全设备。该装置由连续排列的三个灯

组成，由输出 Left、Middle 和 Right 控制，如果某个信号有效，则相应的指示灯应该打开。一次只打开一盏灯，且灯光从左向右“移动”，然后再从右向左，从而吓跑相信设备正在监视其活动的小偷。绘制用于控制电子眼的有限状态自动机的图形表示。请注意，“电子眼”的移动速度将由时钟速度（不应太大）控制，并且没有输入信号。

- A.38 [10] < A.10 > 给上题构造的有限状态自动机分配状态编码，并为每个输出写出一组逻辑表达式，包含下一状态位。
- A.39 [15] < A.2, A.8, A.10 > 使用 3 个 D 触发器和若干逻辑门构建一个 3 位计数器。输入应包括一个将计数器复位为 0 的信号 reset，以及一个增加计数器的信号 inc。输出是计数器的值。当计数器的值为 7 并且继续增加时，重新归零。
- A.40 [20] < A.10 > 格雷码是一系列二进制数，其特性是序列中相邻的编码最多只有一位不同。例如，这是一个 3 位二进制格雷码序列：000，001，011，010，110，111，101，100。使用 3 个 D 触发器和 1 个 PLA，构造一个 3 位格雷码计数器，它有两个输入：reset，将计数器设置为 000；inc，使计数器转到序列中的下一个值。请注意，该编码序列是循环的，因此序列中 100 的下一个值为 000。
- A.41 [25] < A.10 > 我们希望在 A.10 节交通灯的示例中添加黄灯。通过将时钟更改为以 0.25Hz（4 秒的时钟周期时间，这是黄灯的持续时间）运行来完成此操作。为了防止绿灯和红灯循环太快，我们添加了一个 30 秒的计时器。计时器有一个输入，称为 TimerReset，该信号用于重新启动计时器；还有一个输出，称为 TimerSignal，表示已经过去 30 秒。此外，为了将黄灯包含进去，必须重新定义交通信号。我们通过为每个灯定义两个输出信号来实现，分别为 green 和 yellow。如果输出 NSgreen 有效，则绿灯亮；如果输出 NSyellow 有效，黄灯亮。如果两个信号均关闭，则红灯亮。不要同时将 green 和 yellow 信号置为有效，否则美国司机肯定会感到困惑，即使欧洲司机明白这其中的含义！为这个改进的控制器绘制有限状态自动机的图形表示。状态的名称不要和输出信号相同。
- A.42 [15] < A.10 > 写出上题中描述的交通灯控制器的下一状态和输出函数表。
- A.43 [15] < A.2, A.10 > 为交通灯示例中的状态分配状态编号，并使用上题的表格为每个输出写入一组逻辑表达式，包括下一状态输出。
- A.44 [15] < A.3, A.10 > 使用 PLA 实现上题的逻辑表达式。

自我检测答案

- A.2 节 否，如果 $A=1$ ， $C=1$ ， $B=0$ ，则第一个为真，第二个为假。
- A.3 节 C 。
- A.4 节 全部相同。
- A.4 节 $A = 0$ ， $B = 1$ 。
- A.5 节 2。
- A.6 节 1。
- A.8 节 c 。
- A.10 节 b 。
- A.11 节 b 。

术 语 表

absolute address 绝对地址 访问单个变量或函数时使用的内存地址。	进制接口 指令系统中的用户部分，加上操作系统与应用程序接口部分。ABI 标准用来保证不同机器间的二进制可移植性。
abstraction 抽象 为实现复杂设计，对计算机系统的底层细节进行屏蔽的方法。	architectural registers 体系结构寄存器 处理器中可见的寄存器的指令系统，例如，在 RISC-V 中，有 32 个整型寄存器和 32 个浮点寄存器。
access bit 被访问位（也称为被使用位或引用位）页表属性的一个字段，用来实现 LRU（最近最少使用）算法或其他替换算法。	arithmetic intensity 算术强度 程序中浮点运算数量与程序从主存中访问的字节数的比值。
acronym 缩写 单词首字母组成的字符串。比如，RAM 是 Random Access Memory（随机访问存储器）的缩写，CPU 是 Central Processing Unit（中央处理单元）的缩写。	arithmetic logic unit (ALU) 算术逻辑单元 执行加法、减法和通常的逻辑运算（如与和或）的硬件。
active matrix display 有源矩阵显示器（也称为液晶显示屏）使用晶体管来控制光在每个像素点的传输。	assembler 汇编器 将符号语言的指令翻译为二进制语言的程序。
address 地址 用来描述在内存中的位置的数值。	assembler directive 汇编制导 一种告诉汇编程序如何翻译程序但不产生机器指令的操作，总是以句号开始。
address translation 地址转换（也称为地址映射）将虚地址映射到内存访问的真实物理地址上。	assembly language 汇编语言 能被翻译为二进制机器语言的符号语言。
address mode 寻址模式 通过对操作数和地址的不同处理来确定存储访问方式。	asserted 有效 信号为逻辑高或为真。
aliasing 别名 使用两个不同的地址访问同一个对象的情况，一般出现在使用虚拟存储的系统中。例如，在不同的进程中使用不同的虚拟地址访问相同物理页面。	asserted signal 有效信号 一个是逻辑高或 1 的信号。
alignment restriction 对齐限制 数据在内存中的地址需与自然边界对齐。	backpatching 回填 在将汇编语言翻译为机器指令过程中的一种技术。汇编程序通过扫描一遍程序为每条指令建立一种（可能不完整的）二进制表示，再返回填写之前不确定的标签。
Amdahl's Law Amdahl 定律 对于一种给定的改进，其带来的性能提升受限于这种改进的改进量。这是收益递减规律的量化版。	basic block 基本块 不包含分支指令（除非可能在结尾）、不包含分支指令目标指令或分支标签（除非可能在开头）的一段指令序列。
AND 与 一种逻辑位操作。两数相与，都为 1 时结果为 1。	behavioral specification 行为规范 描述一个数字系统如何正常运作。
antidependence 反相关（也称为名字相关）由于使用相同寄存器名而带来的指令之间的相关，与真相关不同，可以采取重命名技术消除。	benchmark 基准程序 一个用于比较计算机性能的程序。
antifuse 反熔断 可编程集成电路中的技术，可使得两条线永久相连。	biased notation 偏移表示法 最大的负数用 $00\cdots 000_2$ 表示，最大的正数用 $11\cdots 111_2$ 表示，0 一般用 $10\cdots 000_2$ 表示，即通过将数加一个偏移使其具有非负的表示形式。
application binary interface (ABI) 应用程序二	

- binary digit** 二进制位 二进制数字之一，为 0 或 1，信息的基本组成。
- bisection bandwidth** 切分带宽 多处理器中两个相等部分之间的带宽。这种测量表示多处理器的最差拆分情况。
- block (or line)** 块 (或行) 可出现或不出现在 cache 中的信息的最小单位。
- blocking assignment** 阻塞赋值语句 在 Verilog 语言中，在执行下一条语句前完成的赋值语句。
- branch address table** 分支地址表 (也称为分支表) 由二选一的指令序列的地址构成的表。
- branch-and-link instruction** 带链接的分支指令 分支到一个地址，同时将下一条指令的地址保存在寄存器 (在 RISC-V 中通常是 x1 寄存器) 的指令。
- branch not taken or (untaken branch)** 分支未发生跳转 (或未发生的分支) 一种分支指令，其分支条件不成立，程序计数器 (PC) 变为分支指令的下一条指令的地址。
- branch prediction** 分支预测 一种解决分支冒险的方法。它预测分支的结果并沿预测方向执行，而不是等分支结果确定后才开始执行。
- branch prediction buffer** 分支预测缓冲 (也称为分支历史表) 由分支指令的低位部分索引的一小块存储，包含一至多位来表明最近分支是否发生。
- branch taken** 分支发生跳转 一条分支指令，其分支条件满足，PC 变为分支目标地址。所有无条件分支指令都是发生的分支。
- branch target address** 分支目标地址 在一个分支指令中指定的地址，如果分支发生，该地址成为新的 PC 的值。在 RISC-V 体系结构中，分支目标地址为该指令的立即数字段，与分支指令的地址的和。
- branch target buffer** 分支目标缓冲 一种用来缓存分支目标 PC 或分支目标指令的结构。通常是一个带标签的 cache，比简单的预测缓冲消耗更多硬件。
- bus** 总线 在逻辑设计中，被共同看作一个逻辑信号的一组数据线。或者，一组有多个源和用途的共享线。
- cache memory** 缓存 一种小而快的存储器，作为大而慢存储器的缓冲。
- cache miss** 缓存失效 由于数据不在缓存中，而无法填充缓存的数据请求。
- callee** 被调用者 根据调用者提供的参数，执行一系列已经存储的指令。然后控制返回到调用者的过程。
- callee-saved register** 被调用者保存寄存器 一种由程序保存的寄存器，进行过程调用。
- caller** 调用者 调用一个过程并提供必要参数值的程序。
- caller-saved register** 调用者保存寄存器 调用者程序保存的寄存器。由被调用的程序保存的寄存器。
- capacity miss** 容量失效 一种缓存失效，因为即使是全相联也不能包含为满足请求所需要的所有块。
- central processing unit (CPU)** 中央处理单元 (也称为中央处理器或处理器) 计算机中包含数据通路和控制的活跃部分，做加法、比较、向 I/O 设备发信号使其激活等。
- clock cycle** 时钟周期数 (也称为滴答数、时钟滴答数、时钟数、周期数) 一个时钟周期的时间，通常是处理器时钟。
- clock cycles per instruction (CPI)** 每条指令时钟周期数 执行一个程序或程序片段，平均每条指令的时钟周期数。
- clock period** 时钟周期长度 一个时钟周期的长度。
- clock skew** 时钟扭斜 两个状态元素看到时钟沿的时间之间的绝对时间差。
- clocking methodology** 时钟同步方法 用来确定数据相对于时钟何时稳定和有效的方法。
- cloud computing** 云计算 指通过互联网提供服务的大量服务器。一些提供商出租数量动态变化的服务器作为实用程序。
- cluster** 集群 通过局域网连接的一组计算机，其作用等同于一个大型的多处理器。
- clusters** 集群 一组通过 I/O 接口与标准网络交换机连接而形成的消息传递多处理机。
- coarse-grained multithreading** 粗粒度多线程 硬件多线程的一种形式，暗示仅在一些重要事件 (如缓存缺失) 之后进行线程切换。
- combinational element** 组合单元 一个操作单元，如 AND 门或 ALU。
- combinational logic** 组合逻辑 一个逻辑系统，

其模块不包含存储器，因此在给定相同输入的情况下计算出相同的输出。

commit unit 提交单元 位于动态流水线或乱序流水线中的一个单元，用以决定何时可以安全地将操作结果发至程序员可见的寄存器和存储器。

compiler 编译器 将高级语言语句翻译为汇编语言语句的程序。

compulsory miss 强制失效（也称为冷启动失效） 第一次访问从未出现在 cache 中的块所引起的缓存失效。

conditional branch 条件分支指令 该指令测试一个值，并且允许根据测试的结果将控制转移到程序中的新地址。

conflict miss 冲突失效（也称为碰撞失效） 在组相联或者直接映射 cache 中，很多块竞争同一个组导致的失效。这种失效在使用相同大小的全相联 cache 中可被消除。

context switch 上下文切换 为允许另一个不同的进程使用处理器，改变处理器内部的状态，并保存返回正在执行的进程所需要的状态。

control 控制器 处理器中根据程序的指令，控制数据通路、存储器和 I/O 设备的部件。

control hazard 控制冒险（也称为分支冒险） 由于取到的指令并不是所需要的，或者指令地址的流向不是流水线所预期的，导致正确的指令无法在正确的时钟周期内执行。

control signal 控制信号 决定多选器的选择或指示功能单元操作的信号；与数据信号相比，数据信号包含功能单元所操作的信息。

correlating predictor 相关预测器 结合特定分支的局部行为，以及一些最近执行的分支的行为的全局信息的分支预测器。

CPU execution time CPU 执行时间（也称为 CPU 时间） CPU 为特定任务做计算的 actual 时间。

crossbar network 交叉开关网络 任何一个节点仅需一次即可与其他任意一个节点通信的网络。

D flip-flop D 触发器 有一个数据输入的触发器，它在时钟沿将输入信号的值存入内部存储器。

data hazard 数据冒险（也称为流水线数据冒

险）因无法提供指令执行所需数据而导致指令不能在预期的时钟周期内执行。

data race 数据竞争 如果来自不同线程的两个内存访问指向同一个地址，它们连续出现，并且其中至少一个是写操作，那么这两个存储访问形成数据竞争。

data segment 数据段 UNIX 目标文件或可执行文件中的一段，包含程序使用的初始化数据的二进制表示。

data transfer instruction 数据传送指令 在存储器和寄存器之间移动数据的命令。

data-level parallelism 数据级并行 对相互独立的数据执行相同操作所获得的并行。

datapath 数据通路 处理器中执行算术操作的部件。

datapath element 数据通路单元 一个用来操作或保存处理器中数据的单元。在 RISC-V 的实现中，数据通路单元包括指令存储器、数据存储器、寄存器堆、ALU 和加法器。

deasserted 无效 信号为逻辑低或假。

deasserted signal 无效信号 一个为逻辑低或 0 的信号。

decoder 译码器 一个为有 n 位输入和 2^n 个输出的逻辑块，每种输入的组合只对应一种输出。

defect 瑕疵 晶圆上或者图样化过程中的一个微小缺陷，包含这个缺陷会导致芯片失效。

delayed branch 延迟转移 不管分支条件是否成立，分支指令之后的那条指令总被执行的一种分支。

die 晶片 从晶圆中切割出来的一个独立的矩形区域，更正式的叫法是芯片（chip）。

direct-mapped cache 直接映射缓存 一种 cache 结构，每个内存地址正好只映射到 cache 中的一个位置。

dividend 被除数 被除的数。

divisor 除数 去除被除数的数。

don't-care term 无关项 逻辑函数的一个元素，输出与所有输入的取值无关。无关项可以用不同的方式指定。

double precision 双精度 以 64 位双字表示的浮点值。

doubleword 双字 计算机中另一种自然的访

- 问单元，通常是一组 64 位；对应 RISC-V 体系结构的一种寄存器的大小。
- dynamic branch prediction 动态分支预测** 根据运行信息在运行时进行分支预测。
- dynamic multiple issue 动态多发射** 实现多发射处理器的一种方式，其中很多决策是由处理器在执行阶段做出的。
- dynamic pipeline scheduling 动态流水线调度** 为避免阻塞对指令进行重排序的硬件支持。
- dynamic random access memory (DRAM) 动态随机访问存储** 构建为集成电路的存储器，它提供对任何位置的随机访问。访问时间为 50 纳秒，2012 年每千兆字节的成本为 5 到 10 美元。
- dynamically linked libraries (DLL) 动态链接库** 在程序执行过程中被链接的库例程。
- edge-triggered clocking 边沿触发的时钟同步** 所有状态的改变发生于时钟沿的时钟机制。
- embedded computer 嵌入式计算机** 用于运行一个预定应用程序或软件集合的另一个设备内的计算机。
- EOR 异或** 二元操作数的逻辑按位运算，计算两个操作数的异或。也就是说，只有两个操作数的值不同时，它才会计算 1。
- error detection code 检错码** 能够检测数据中的错误并纠正错误的代码，不能确定错误的准确位置。
- exception 例外（也称为中断）** 一种打断程序执行的非预期的事件，用于溢出检测。
- exception enable 例外使能（也称为中断使能）** 用于控制处理器是否响应异常的信号或动作；在处理器安全地保存重启所需信息之前，必须阻止异常的发生。
- executable file 可执行文件** 一个具有目标文件格式的功能程序，不包含未确定的引用。它可以包含符号表和调试信息。“被剥离的可执行程序”不包含这些信息。可能包含加载器所需的重定位信息。
- exponent 指数（也称为阶码）** 在浮点运算的数值表示系统中，放置在指数字段中的值。
- external label 外部标签（也称为全局标签）** 指向一个对象的标签，该标签可被除了定义该标签的文件之外的文件引用。
- false sharing 伪共享** 当两个不相关的共享变量放在相同的 cache 块中时，尽管每个处理器访问的是不同的变量，但是在处理器之间还是将整个块进行交换。
- field programmable devices (FPD) 现场可编程设备** 包含组合逻辑，以及可能的存储器设备，且可由最终用户配置的集成电路。
- field programmable gate array (FPGA) 现场可编程门阵列** 一个包含组合逻辑块和触发器的可配置集成电路。
- fine-grained multithreading 细粒度多线程** 硬件多线程的一种形式，暗示每条指令执行之后都进行线程切换。
- finite-state machine 有限状态自动机** 由一组输入和输出，以及下一状态函数和输出函数组成的时序逻辑函数。下一状态函数将当前状态和当前输入映射为一个新的状态，输出函数将当前状态和当前输入映射为一组确定的输出。
- flash memory 闪存** 一种非易失性半导体内存，价格和速度均低于 DRAM，但每一位比磁盘昂贵，比磁盘快。访问时间大约为 5 到 50 微秒，2012 年每千兆字节的成本为 0.75 到 1.00 美元。
- flip-flop 触发器** 一个存储元件，其输出等于元件内部存储状态的值，并且内部状态仅在时钟沿上改变。
- floating point 浮点** 计算机算术，表示二进制点不固定的数字。
- flush 清除** 因发生了意外而丢弃流水线中的指令。
- formal parameter 形式参数** 过程或者宏的参数，一旦宏被展开，这个参数将被变量替换。
- forward reference 前向引用** 一个标签在被定义之前就被使用。
- forwarding 前递（也称为旁路）** 一种解决数据冒险的方法，从内部缓冲中取回数据，而不是等到数据从程序员可见的寄存器或存储器中到达。
- fraction 分数** 值通常在 0 和 1 之间，放置在分数字段中。
- frame pointer 帧指针** 指向给定过程中保存的寄存器和局部变量的值。
- fully associated cache 全相联缓存** 一种缓存结构，一个块可以放在缓存中的任意位置。
- fully connected network 全连接网络** 通过在

每个节点之间提供专用通信链路，来连接处理器 – 内存节点的网络。

fused multiply add 混合乘加指令 一个浮点指令，既执行乘法又执行加法，但在加法后只舍入一次。

gate 逻辑门 实现基本逻辑功能的设备，如与、或。

global miss rate 全局失效率 在多级 cache 中所有级都缺失的那部分访问。

global pointer 全局指针 指向静态数据区的保留寄存器。

guard 保护位 在浮点数的中间计算期间，两个额外位的第一位保持在右侧；用于提高舍入精度。

handler 处理程序 用于“处理”例外或中断的软件程序的名称。

hardware description language 硬件描述语言 一种用于描述硬件的编程语言，用于生成硬件设计的模拟，也可作为生成实际硬件的综合工具的输入。

hardware multithreading 硬件多线程 当一个线程被阻塞时，通过切换到另一个线程来提高处理器的利用率。

hardware synthesis tools 硬件综合工具 计算机辅助设计软件，根据对数字系统的行为描述，可生成门级设计。

hexadecimal 十六进制 基数为 16 的数。

high-level programming language 高级编程语言 一种轻便的语言，如 C、C++、Java 或 Visual Basic，由单词和代数符号组成，可由编译器翻译成汇编语言。

hit rate 命中率 在一层存储层次结构中找到目标数据的存储访问比例。

hit time 命中时间 访问某存储器层次结构所需要的时间，包括判断这个访问是命中还是缺失所需的时间。

hold time 保持时间 在时钟沿之后，输入必须保持有效的最短时间。

implementation 实现 遵循体系结构抽象的硬件。

imprecise interrupt 非精确中断（也称为非精确异常） 流水线处理器中的中断或异常，其不与导致中断或异常的指令精确地关联。

in-order commit 按序提交 流水线执行的结果以取出指令的顺序写回程序员可见寄存器

的一种提交方式。

input device 输入设备 为计算机提供信息的装置，如键盘和鼠标。

instruction 指令 计算机硬件能够理解并且遵循的命令。

instruction count 指令数 程序执行的指令数量。

instruction format 指令格式 由二进制数组成的指令表示形式。

instruction latency 指令延迟 指令的固有执行时间。

instruction-level parallelism 指令级并行 指令间的并行性。

instruction mix 指令混合比例 一个或多个程序中指令的动态频率的度量。

instruction set architecture 指令系统体系结构（也称为体系结构） 低层次软件和硬件之间的抽象接口，包含了写一段能正确运行的机器语言程序需要的所有信息，包括指令、寄存器、存储访问和 I/O 等。

integrated circuit 集成电路（也称为芯片） 一种结合数十至数百万个晶体管的器件。

interrupt 中断 来自处理器之外的异常（一些体系结构对所有的异常都使用术语“中断”）

interrupt handler 中断处理程序 一段由于异常或中断而运行的代码。

issue packet 发射包 在一个时钟周期内发射的多条指令的集合。这个包可以由编译器静态生成，也可以由处理器动态生成。

issue slots 发射槽 指令在一个给定的时钟周期内可以发射的位置；做一个类比，这些位置对应着冲刺起跑的位置。

Java bytecode Java 字节码 一个指令集中，为了解释 Java 程序而设计的指令。

just in time compiler (JIT) 即时编译器 一类通用编译器的名称，编译器能够在运行时将解释的代码段翻译成宿主计算机上的机器语言。

latch 锁存器 一个存储元件，其中输出等于元件内部存储状态的值，只要适当的输入改变并且时钟有效就改变状态。

latency (pipeline) 延迟（流水线） 流水线的级数，或执行过程中两条指令间的级数。

least recently used (LRU) 最近最少使用 一种替换策略，总是替换最长时间没有被使

用的块。

least significant bit 最低有效位 RISC-V 双字中最右的一位。

level-sensitive clocking 电平敏感的时钟同步 一种时序方法，其中状态变化发生在高或低时钟电平，但不是瞬时的，因为这种变化发生在边沿触发设计中。

linker 链接器（也称为链接编辑器） 一个系统程序，把各个独立的汇编机器语言结合起来，并且确定所有未定义的标记，最后生成可执行文件。

liquid crystal display 液晶显示器 一种使用液态聚合物薄层的显示技术，可以根据是否施加电荷来传输或阻挡光线。

load-use data hazard 载入－使用的数据冒险 一种特定的数据冒险，指当载入指令要取的数据还没取回来时，其他指令就需要该数据的情况。

loader 加载器 把目标程序装载到内存中以准备运行的系统程序。

local area network (LAN) 局域网 一种在一定地理区域使用的传输数据的网络，通常在一个建筑物内。

local label 局部标签 指向一个对象的标签，只能在定义这个标签的文件中使用。

local miss rate 局部失效率 在多级 cache 中，某一级 cache 的缺失率。

lock 锁 一个时刻仅允许一个处理器访问数据的同步装置。

lookup tables (LUT) 查找表 在现场可编程设备中，由少量的逻辑和 RAM 组成的单元。

loop unrolling 循环展开 一种从存取数组的循环中获取更多性能的技术，其中循环体会被复制多份并且不同循环体中的指令可能被调度到一起。

machine language 机器语言 在计算机系统中，用以交流的二进制表示形式。

macro 宏 一种模式匹配和替换技术，为常用的指令序列提供简单的命名机制。

magnetic disk 磁盘（也称为硬盘） 由磁性记录材料涂覆的旋转盘片组成的非易失性二级存储器形式。它们是旋转的机械设备，访问时间大约为 5 到 20 毫秒，2012 年的每千兆字节成本为 0.05 到 0.10 美元。

main memory 主存 用于在程序运行时保存程序。

现代计算机中，通常由 DRAM 构成内存。

memory 内存 程序运行时的存储空间，也包含程序运行时所需的数据。

memory hierarchy 存储层次 一种由多存储层次组成的结构，存储器的容量和访问时间随着与处理器距离的增加而增加。

message passing 消息传递 通过显式发送和接收信息的方式在多个处理器之间通信。

metastability 亚稳态 如果信号在设置和保持时间不稳定时进行采样，可能导致信号采样值落入高值和低值之间的不确定区域，这种情况即为亚稳态。

microarchitecture 微体系结构 处理器的组织架构，包括主要的功能单元及它们的互连关系与流水线控制。

million instructions per second (MIPS) 每秒百万条指令 基于数百万条指令的程序执行速度的度量。MIPS 被计算为指令总数除以执行时间与 10^6 的乘积。

MIMD 多指令流多数据流 一种多处理器。

minterms 小项（也称为乘积项） 通过 AND 操作连接的一组逻辑输入；乘积项形成可编程逻辑阵列（PLA）的第一个逻辑阶段。

miss penalty 失效损失或者失效代价 将相应的块从低层存储器替换到高层存储器所需的时间，包括访问块、将数据逐层传输、将数据插入发生缺失的层和将信息块传送给请求者的时间。

miss rate 失效率 在高层存储器中没有找到目标数据的存储访问比例。

most significant bit 最高有效位 在 RISC-V 双字节字中最左边的一位。

multicore microprocessor 多核微处理器 在单一集成电路上包含多个处理器（核）的微处理器。当今几乎所有台式机和服务器中的微处理器都是多核的。

multilevel cache 多级 cache 存储系统由多级 cache 组成，而不仅仅只有主存和一个缓存。

multiple issue 多发射 一种单时钟周期内发射多条指令的机制。

multiprocessor 多处理器 一种至少有两个处理器的计算机系统。与之对应的概念是单处理器。单处理器计算机只有一个处理器，

现在这种计算机已经使用得越来越少了。

multistage network 多级网络 在每个节点上提供小型交换机的网络。

NAND gate 与非门 一个倒置的与门。

network bandwidth 网络带宽 非正式用语，用于表示网络传输速度的峰值；既可以指单一链路的速度，也可以指网络中全部链路的共同传输速度。

next-state function 下一状态函数 根据当前状态及当前输入来确定有限状态自动机下一状态的组合函数。

nonblocking assignment 非阻塞赋值 一种仅在求值右侧后才继续执行的任务，只有在右侧全部被求值后才能对左侧进行赋值。

nonblocking cache 非阻塞 cache 在处理器处理前面的 cache 缺失时仍可正常访问的 cache。

nonuniform memory access (NUMA) 非统一存储访问 使用单一地址空间多处理器的一种类型，某些存储访问速度高于其他访问，访问速度与访问哪个处理器及访问哪个字相关。

nonvolatile memory 非易失性存储 即使在没电源的情况下也可保留数据的存储器形式，用于在运行之间存储程序。DVD 就是非易失性存储器。

nop 空指令 一种不进行任何操作或不改变任何状态的指令。

NOR 或非 具有两个操作数的逻辑逐位操作，用于计算两个操作数的或的非。也就是说，它只在两个操作数中都有 0 时才计算为 1。

NOR gate 或非门 一个倒置的或门。

normalized 规格化 没有前导 0 的浮点数。

NOT 非 一个逻辑逐位操作，该操作将操作数反转；也就是说，用 0 替换 1，用 1 替换 0。

object oriented language 面向对象语言 一种面向对象而非动作或数据的编程语言。

one's complement 反码 使用 $10\dots000_2$ 表示最大负数， $01\dots11_2$ 表示最大正数，正数和负数的数量相同，但保留两个零，即正零 ($00\dots00_2$) 与负零 ($11\dots11_2$)。该术语也用于表示模式中每个位的反转，即由 0 变 1 或由 1 变 0。

opcode 操作码 表示指令操作和格式的字段。

OpenMP OpenMP 语言 在 C、C++ 或 Fortran

中用于共享内存多处理编程的 API，可以运行于 UNIX 和 Microsoft 平台。它包括编译器指示、库和运行时指令。

OR 或 使用两个操作数的逻辑逐位操作，只要两个操作数中有一个为 1，则计算结果为 1。

out-of-order execution 乱序执行 流水线执行的一种情况，即执行的指令被阻塞时不会导致后面的指令等待。

output device 输出设备 将计算结果传送给用户或另一台计算机的装置。

page fault 缺页异常 当访问页面不在主存中时发生的事件。

page table 页表 该表包含虚拟内存系统中的虚实地址转换。页表存储在内存中，通常由虚拟页面编号索引；如果页面当前在内存中，页表中的每个条目都包含该虚拟页面的物理页号。

parallel processing program 并行处理程序 可同时运行在多个处理器上的单一程序。

PC-relative addressing PC 相对寻址 一种寻址机制，它将 PC 和指令中的常数相加作为寻址结果。

personal computer (PC) 个人计算机 专为个人设计的计算机，通常包含图形显示器、键盘和鼠标。

personal mobile devices (PMD) 个人移动设备 连接到互联网的小型无线设备，依靠电池供电，并通过下载应用程序来安装软件。典型的例子是智能手机和平板电脑。

physical address 物理地址 主存中的地址。

physically addressed cache 物理地址 cache 使用物理地址寻址的 cache。

pipeline stall 流水线停顿 (也称为气泡) 为了解决冒险而实施的一种阻塞。

pipelining 流水线 一种实现多条指令重叠执行的技术，与生产流水线类似。

pixel 像素 最小的单个图片元素。屏幕由数十万至数百万像素组成，以矩阵形式组织。

pop 出栈 从堆栈中移除元素。

precise interrupt 精确中断 (也称为精确例外) 流水线处理器中的中断或异常与导致中断或异常的指令精确地关联。

prefetching 预取 使用特殊指令将未来可能用到的指定地址的 cache 块提前搬到 cache

- 中的一种技术。
- procedure 过程** 根据提供的参数执行特定任务的存储子程序。
- procedure call frame 过程调用帧** 用来保存被调用过程的参数, 保存可能会被过程修改的寄存器的值, 但是这些寄存器的值不会被调用者所修改, 并为被调用程序的局部变量提供空间。
- procedure frame 过程帧 (也称为活动记录)** 栈中包含过程所保存的寄存器以及局部变量的片段。
- process 进程** 包括一个或多个线程、地址空间和操作系统状态。因此, 进程切换通常调用操作系统, 而不是切换线程。
- program counter (PC) 程序计数器** 包含当前程序正在执行的指令地址的寄存器。
- programmable array logic (PAL) 可编程阵列逻辑** PAL 由一个可编程的“与”平面和一个固定的“或”平面构成。
- programmable logic array (PLA) 可编程逻辑阵列** 可编程逻辑器件的一种, 它是与/或阵列均可编程的、包含有记忆元件的大规模集成电路, 能实现任意逻辑函数的组合电路以及时序电路。
- programmable logic device (PLD) 可编程逻辑器件** 包含组合逻辑的集成电路, 其功能由最终用户配置。
- programmable ROM (PROM) 可编程 ROM** 一种只读存储器, 可在设计人员知道其内容时进行编程。
- propagation time 传播时间** 从输入到触发器传播到触发器的输出所需的时间。
- protection 保护** 用于确保共享处理器、存储器或 I/O 设备的多个进程不会有意或无意地通过读取或写入彼此的数据来相互干扰的一组机制。这些机制还将操作系统与用户进程隔离开来。
- pseudoinstruction 伪指令** 汇编语言指令的一个变种, 常被看作一条汇编指令。
- Pthreads 并行线程** 用于创建和操作线程的 UNIX API。它的结构是一个库。
- push 压栈** 向栈中增加元素。
- quotient 商** 除法的主要结果。该数乘以除数再加上余数得到被除数。
- read-only memory (ROM) 只读存储器** 一种存储器, 其内容在创建时指定, 之后只能读取内容。ROM 用作结构化逻辑, 通过将逻辑功能中的术语用作地址输入并将输出用作存储器的每个字中的位来实现一组逻辑功能。
- receive message routine 接收消息例程** 具有私有存储器的机器中一个处理接收来自其他处理器消息的例程。
- recursive procedures 递归程序** 通过一连串的调用直接或间接调用自己的程序。
- reduction 约简** 处理一个数据结构并返回单一值的函数。
- reference bit 引用位 (也称为使用位或访问位)** 每当访问一个页面时该位被置位, 通常用来实现 LRU 或其他替换策略。
- reg 寄存器** 在 Verilog 中是一个寄存器。
- register file 寄存器组 (或寄存器堆)** 状态元素, 由一组寄存器组成, 可通过提供要访问的寄存器编号进行读写。
- register renaming 寄存器重命名** 由编译器或硬件对寄存器进行重命名以消除反相关。
- register use convention 寄存器使用惯例 (也称为过程调用惯例)** 管理过程 (调用) 使用寄存器的软件协议。
- relocation information 重定位信息** UNIX 目标文件中的一段, 根据绝对地址来区别数据字和指令。
- remainder 余数** 除法的次要结果。该数加在商和除数的乘积上产生被除数。
- reorder buffer 重排序缓冲** 动态调度处理器中用于暂时保存执行结果的缓冲区, 等到安全时才将其中的结果写回寄存器或存储器。
- reservation station 保留站** 功能单元的缓冲区, 用来保存操作数和操作。
- response time Also called execution time 响应时间 (也称为执行时间)** 计算机完成某任务所需的总时间, 包括硬盘访问、内存访问、I/O 活动、操作系统开销和 CPU 执行时间。
- restartable instruction 可重启指令** 一种在异常被处理之后能从异常中恢复而不会影响指令的执行结果的指令。
- return address 返回地址** 指向调用点的链接,

- 使过程可以返回到合适的地址。
- rotational latency** 旋转时间 (也称为旋转延迟) 使得合适的扇区旋转到读 / 写头下的时间。
- round** 舍入 使中间浮点结果符合浮点格式的方法, 目标通常是找到可以形式化表示的最近数字。它也是在中间浮点计算期间保留在右侧的两个额外位的第二个的名称, 这提高了舍入精度。
- scientific notation** 科学记数法 使用小数点左边的单个数字呈现数字的表示法。
- secondary memory** 辅助存储 非易失性存储器, 用来保存两次运行之间的程序和数据; 在现代计算机中, 一般由磁盘组成。
- sector** 扇区 磁道上的一段弧称为扇区, 是磁盘中被读或者写的最小信息块。
- seek** 寻道 在一个读或者写操作中, 把磁头定位到合适的磁道的过程。
- segmentation** 分段 一种可变大小的地址映射方案, 其中地址由两部分组成: 映射到物理地址的段号和段偏移量。
- selector value** 选择器值 (也称为控制值) 用于选择多路复用器的输入值之一作为多路复用器的输出的控制信号。
- semiconductor** 半导体 导电能力介于导体和绝缘体之间。
- send message routine** 发送消息例程 具有私有存储器的机器中一个处理器将消息发送给另一个处理器的例程。
- sensitivity list** 敏感变量列表 指定何时应该重新评估 always 块的信号列表。
- separate compilation** 单独编译 将程序划分成多个文件, 每个文件被编译时, 并不知道其他文件的信息。
- sequential logic** 时序逻辑 一组包含内存的逻辑单元, 因此其值取决于输入以及内存的当前内容。
- server** 服务器 用于为多个用户运行较大程序的计算机, 通常是并行的, 并且通常只能通过网络访问。
- set-associative cache** 组相联 cache cache 的另一种组织方式, 块可以放置到 cache 中的部分位置 (至少两个)。
- setup time** 建立时间 在时钟沿之前, 存储器设备的输入必须有效的最短时间。
- shared memory multiprocessor (SMP)** 共享存储多处理器 具有单一地址空间的并行处理器, 存取时采用隐式通信的方式。
- sign-extend** 符号扩展 为增加数据项的长度, 将原数据项的最高位复制到新数据多出来的高位。
- silicon** 硅 一种自然元素, 是一种半导体。
- silicon crystal ingot** 硅锭 由硅晶体组成的棒, 直径 8 ~ 12 英寸, 长约 12 ~ 24 英寸。
- SIMD** 单指令流多数据流 同样的指令在多个数据流上操作, 和向量处理器或阵列处理器一样。
- simple programmable logic device (SPLD)** 简单可编程逻辑器件 可编程逻辑器件, 通常包含一个 PAL 或 PLA。
- simultaneous multithreading (SMT)** 同时多线程 多线程一个版本, 通过利用多个问题所需的资源、动态调度的微体系结构, 降低多线程成本。
- single precision** 单精度 以 32 位字表示的浮点值。
- single-cycle implementation** 单周期实现 (也称为单时钟周期实现) 一个时钟周期执行一条指令的实现机制。
- SISD** 单指令流单数据流 一个单处理器。
- Software as a Service (SaaS)** 软件即服务 软件不再是安装和运行在客户自己的计算机上, 而是运行在远程计算机上, 通过网络来使用。典型情况是通过网络接口为客户服务, 然后根据使用情况向客户收费。
- source language** 源语言 程序最初编写的高级语言。
- spatial locality** 空间局部性 本地原则指出, 如果数据位置被引用, 那么附近地址的数据位置也将很快被引用。
- speculation** 推测 一种编译器或处理器推测指令结果以消除执行其他指令对该结果依赖的技术。
- split cache** 分离 cache 一级 cache 由两个独立的 cache 组成, 两者可以并行工作, 一个处理指令, 另一个处理数据。
- SPMD** 单程序多数据流 传统的 MIMD 编程模型, 其中一个程序在所有处理器上运行。
- stack** 栈 被组织成后进先出队列形式并用于

寄存器换出的数据结构。

stack pointer 栈指针 指示栈中最近分配的地址的值。它指示寄存器被换出的位置，或寄存器旧值的存放位置。

stack segment 堆栈段 程序用来保存过程调用帧的那段内存。

state element 状态单元 一个存储单元，如寄存器或存储器。

static data 静态数据 包含数据的那部分内存，其大小为编译器所知，生命周期为整个程序的运行时间。

static multiple issue 静态多发射 实现多发射处理器的一种方式，其中决策是在执行前的编译阶段做出的。

static random access memory (SRAM) 静态随机访问存储器 一种存储器的集成电路，但是更快，比 DRAM 集成度低。

sticky bit 粘滞位 除了保护位和舍入位之外，还有一位也用于舍入，只要舍入位右侧有非零位，就设置该位。

stored-program concept 存储程序概念 多种类型的指令和数据均以数字形式存储于存储器中的概念，存储程序型计算机即源于此。

strong scaling 强比例缩放 在多处理器上不需增加问题规模即可获得加速比。

structural hazard 结构冒险 因缺乏硬件支持而导致指令不能在预定的时钟周期内执行的情况。

structural specification 结构规范 描述数字系统如何按照单元的层次连接进行组织。

sum of products 积之和 逻辑表示形式，将使用 AND 运算符连接的项做逻辑和 (OR) 的结果。

supercomputer 超级计算机 一类性能和成本最高的计算机；它们被配置为服务器，通常花费数十亿美元到数亿美元。

superscalar 超标量 一种先进的流水线技术，通过在执行期间选择它们，处理器可以在每个时钟周期执行多条指令。

supervisor mode 超级用户模式 (也称作管理态、核心模式) 运行操作系统进程的模式。

swap space 交换区 为进程的全部虚拟地址空间所预留的磁盘空间。

symbol table 符号表 用来将标签的名字和指

令占用的内存字的地址相匹配的表。

synchronization 同步 对可能运行于不同处理器上的两个或者更多进程的行为进行协调的过程。

synchronizer failure 同步故障 触发器进入亚稳态，其中一些逻辑块读取该触发器的输出为 0，但是其他逻辑块读取输出为 1。

synchronous system 同步系统 一种采用时钟的存储系统，只有当时钟指示信号值稳定时才读取数据信号。

system call 系统调用 将控制权从用户模式转换到管理员模式的特殊指令，触发进程中的一个异常机制。

system CPU time 系统 CPU 时间 在代表程序执行任务的操作系统中花费的 CPU 时间。

systems software 系统软件 提供常用服务的软件，包括操作系统、编译程序、加载程序和汇编程序。

tag 标签 表中的一个字段，包含了地址信息，这些地址信息可以用来判断缓存中的字是否就是所请求的字。

task-level parallelism or process-level parallelism 任务级并行或进程级并行 通过同时运行独立程序来利用多个处理器。

temporal locality 时间局部性 该原则规定如果一个数据位置被引用，那么它将很快被重新引用。

terabyte (TB) 太 (即兆兆字节) 最初表示 1 099 511 627 776 (2^{40}) 字节，尽管通信和辅助存储系统开发人员开始使用该术语表示 1 000 000 000 000 (10^{12}) 字节。为了减少混淆，我们现在使用 tebibyte (TiB) 这个术语来表示 2^{40} 字节，定义 TB 表示 10^{12} 字节。(图 1.1 显示了十进制和二进制值的全部范围和名称。)

text segment 代码段 UNIX 目标文件中的段，包含源文件中例程对应的机器语言代码。

thread 线程 线程包括程序计数器、寄存器状态和堆栈。这是一个轻量级的过程；线程之间通常共享一个地址空间，进程则不会。

three Cs model 3C 模型 将所有的 cache 缺失都归为三种类型的 cache 模型，三类分别为强制缺失、容量缺失和冲突缺失。因其三类名称的英文单词首字母均为 C 而得名。

throughput 吞吐率 (也叫带宽) 性能的另一度
量参数, 表示单位时间内完成的任务数量。

tournament branch predictor 锦标赛分支预测
器 具有多种预测机制的分支预测器。其
带有一个选择器, 对给定分支可选择其中
一个作为预测结果。

track 磁道 磁盘面上的一个同心圆为一个磁道。

transistor 晶体管 一种由电信号控制的简单
开关。

translation-lookaside buffer (TLB) 快表 (也称
为旁视缓冲器) 用于记录最近使用的映
射信息的 cache, 从而可以避免每次都要
访问页表。

truth table 真值表 逻辑操作的一种表示方法,
即列出输入的所有情况和每种情况下的
输出。

underflow (floating-point) 浮点下溢 负指数
变得太大而不适合指数字段的情况。

uniform memory access (UMA) 统一存储访问
无论访存的是哪个处理器, 也无论访存的是
哪个字, 访存时间都大致相同的多处理器。

units in the last place (ulp) 最后位置单位 实
际数字与可表示数字之间最低有效位中的
错误位数。

unmapped 未映射 地址空间的一部分, 在这
个区域不会导致缺页异常。

unresolved reference 未解析引用 需要从外
部获得更多信息以进行完善的引用。

use bit 引用位 (也称为使用位、访问位) 每当
访问一个页面时该位被置位, 通常用来实
现 LRU 或其他替换策略。

use latency 使用延迟 在装载指令与可以无阻
塞使用其结果的指令间相隔的时钟周期数。

user CPU time 用户 CPU 时间 在程序本身
花费的 CPU 时间。

valid bit 有效位 表中的一个字段, 用来标识
一个块是否含有有效数据。

vector lane 向量通道 一或多个向量功能单元
和向量寄存器组的一部分。受高速公路上
提高交通速度的车道启发, 多车道同时
执行向量操作。

vectored interrupt 向量式中断 由异常原因决
定中断控制转移地址的中断。

Verilog 两种最常见的硬件描述语言之一。

very-large-scale integrated (VLSI) circuit 超
大规模集成电路 一种含有数十万至数
百万晶体管的器件。

very long instruction word (VLIW) 超长指令字
一类可以同时启动多个操作的指令集, 其
中操作在单个指令中相互独立, 并且一般
都有独立的操作码域。

VHDL 两种最常见的硬件描述语言之一。

virtual address 虚拟地址 虚拟空间的地址,
当需要访问主存时需要通过地址映射转换
为物理地址。

virtual machine 虚拟机 一种虚拟计算机, 它
的分支和取数指令没有延迟, 且指令集比
实际硬件更丰富。

virtual memory 虚拟存储 一种将主存用作辅
助存储器 cache 的技术。

virtually addressed cache 虚拟地址 cache 一
种使用虚拟地址而不是物理地址访问的
cache。

volatile memory storage 易失性存储 如 DRAM,
只有在上电时才保留数据。

wafer 晶圆 厚度不超过 0.1 英寸的硅锭片,
用来制造芯片。

weak scaling 弱比例缩放 在多处理器上增加
处理器数量的同时按比例增加问题规模所
能获得的加速比。

wide area network (WAN) 广域网 一个延伸
数百公里的网络, 可以跨越一块大陆。

wire 线 在 Verilog 中, 指定一个组合信号。

word 字 计算机中的基本访问单位, 通常是
32 位为一组。

workload 工作负载 运作在计算机上的一组程
序, 可以直接使用用户的一组实际应用程
序, 也可以从实际程序中构建。

write buffer 写缓冲 一个保存等待写入主存数
据的缓冲队列。

write-back 写返回 当发生写操作时, 新值仅
仅被写入 cache 块中, 只有当修改过的块
被替换时才写到较低层存储结构中。

write-through 写直达 (也称为写穿透) 写操作
总是同时更新主存和 cache, 以保持二者
一致性的一种方法。

yield 成品率 (或良率) 合格芯片数占总芯片
数的百分比。

计算机组成与设计 硬件/软件接口 原书第5版·RISC-V版

Computer Organization and Design The Hardware/Software Interface, RISC-V Edition

在广大计算机程序员和工程师中，几乎没有人不知道Patterson和Hennessy的大作，而今RISC-V版的推出，再次点燃了大家的热情。RISC-V作为一种开源体系结构，从最初用于支持科研和教学，到现在已发展为产业标准的指令集。正在和即将阅读本书的年轻人，你们不仅能够从先行者的智慧中理解RISC-V的精髓，而且有望创建自己的RISC-V内核，为广阔的开源硬件和软件生态系统贡献力量。

—— Krste Asanovi, RISC-V基金会主席

教材的选择往往是一个令人沮丧的妥协过程——教学方法的适用度、知识点的覆盖范围、文辞的流畅性、内容的严谨度、成本的高低等都需要考虑。本书之所以是难得一见的好书，正是因为它能满足各个方面的要求，不再需要任何妥协。这不仅是一部关于计算机组成的教科书，也是所有计算机科学教科书的典范。

—— Michael Goldweber, Xavier University

无论是对于80后、90后还是00后，这都是一本应该珍藏在书架上（或iPad中）的计算机体系结构教材。这本书既古老又新颖，不仅介绍了那些伟大的原理——摩尔定律、抽象、加速经常性事件、冗余、存储层次、并行和流水线，而且使用现代设计对这些伟大原理进行了说明。

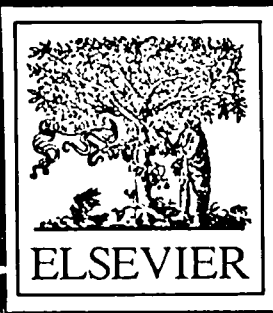
—— Mark D. Hill, University of Wisconsin-Madison

本书不仅讲解计算机体系结构，而且为读者准备了迎接新的变化与挑战的“锦囊”。目前，半导体工艺技术按比例缩小的困难使得所有系统功率受限，而移动系统和大数据处理的性能需求却仍在不断增长。在计算技术的新时代，必须进行软硬件协同设计，并且系统级体系结构优化与部件级优化一样重要。

—— Christos Kozyrakis, Stanford University

Patterson和Hennessy讨论了不断变化的计算机硬件体系结构中的重要议题，强调硬件和软件模块在不同抽象层次上的交互。书中涵盖各种硬件和软件机制，I/O和并行的概念贯穿其中，全景式呈现了后PC时代的计算机体系结构。无论是平板电脑硬件工程师还是云计算软件架构师，如果你正对能源效率和并行化问题一筹莫展，那么本书必将成为不二之选。

—— Jae C. Oh, Syracuse University



华章教育服务微信号



本书译自原版 *Computer Organization and Design: The Hardware/Software Interface, RISC-V Edition*
并由Elsevier授权出版



上架指导：计算机体系结构

ISBN 978-7-111-65214-4



9 787111 652144 >

定价：169.00元

投稿热线：(010) 88379604
读者信箱：hzsj@hzbook.com
客服电话：(010) 88361066 88379833 68326294

华章网站：www.hzbook.com
网上购书：www.china-pub.com
数字阅读：www.hzmedia.com.cn