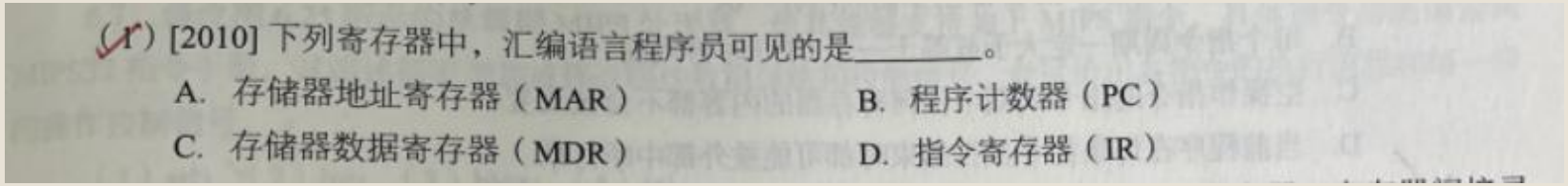
《计算机组成原理》
作业6

2023

33920212204567　任宇

6-2

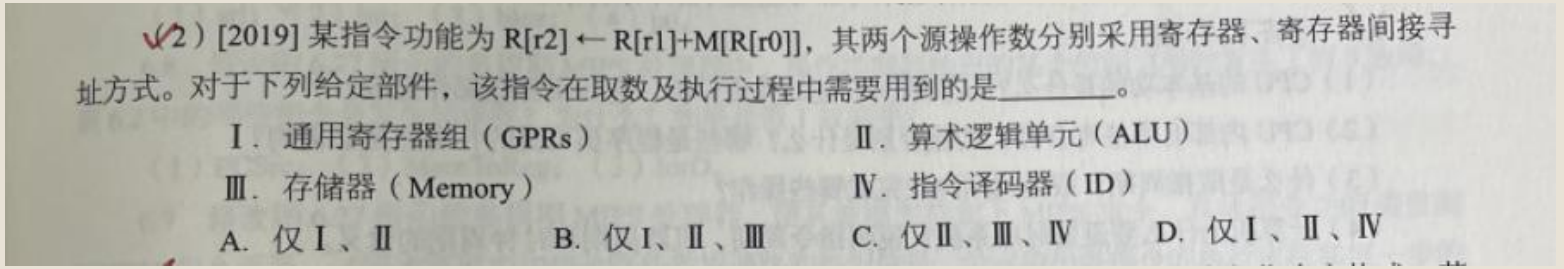

(1) [2010] 下列寄存器中，汇编语言程序员可见的是_____。

A. 存储器地址寄存器（MAR）　　B. 程序计数器（PC）

C. 存储器数据寄存器（MDR）　　D. 指令寄存器（IR）

答：B，原因如下：

可见寄存器是指该寄存器可以通过汇编指令访问。汇编程序员可以通过转移指令、子程序调用等指令来修改PC的值，所以PC是可见寄存器，通用寄存器和程序状态寄存器也都是可见寄存器。


(2) [2019] 某指令功能为 R[r2] ← R[r1]+M[R[r0]]，其两个源操作数分别采用寄存器、寄存器间接寻址方式。对于下列给定部件，该指令在取数及执行过程中需要用到的是_____。

Ⅰ. 通用寄存器组（GPRs）　　Ⅱ. 算术逻辑单元（ALU）

Ⅲ. 存储器（Memory）　　Ⅳ. 指令译码器（ID）

A. 仅Ⅰ、Ⅱ　　B. 仅Ⅰ、Ⅱ、Ⅲ　　C. 仅Ⅱ、Ⅲ、Ⅳ　　D. 仅Ⅰ、Ⅱ、Ⅳ

答：B，原因如下：

源操作数使用了寄存器、寄存器间接寻址方式，所以需要用到存储器和通用寄存器组，同时也要用到算术逻辑单元，不需要使用到指令译码器。

6-2

（3）[2016] 某计算机主存空间为 4 GB，字长为 32 位，按字节编址，采用 32 位定长指令字格式。若指令按字边界对齐存放，则程序计数器（PC）和指令寄存器（IR）的位数至少分别是_____。

A. 30、30　　B. 30、32　　C. 32、30　　D. 32、32

答：B，原因如下：

PC存放的是下一条指令的主存地址，通常位宽和主存地址总线相同，但题中指令按字边界对其存放，因此PC可以按字编址，也就是30位即可。IR用于存放当前正在执行的指令，位宽和指令字长相同，因此IR为32位。

（4）[2019] 下列有关处理器时钟脉冲信号的叙述中，错误的是_____。

A. 时钟脉冲信号由机器脉冲源发出的脉冲信号经整形和分频后形成

B. 时钟脉冲信号的宽度称为时钟周期，时钟周期的倒数为机器主频

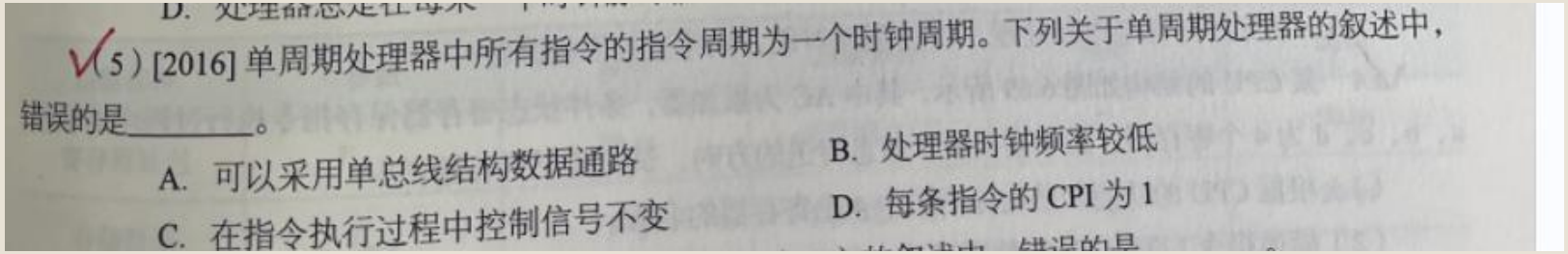
C. 时钟周期以相邻状态单元间组合逻辑电路的最大延迟为基准确定

D. 处理器总是在每来一个时钟脉冲信号时就开始执行一条新的指令

答：D，原因如下：

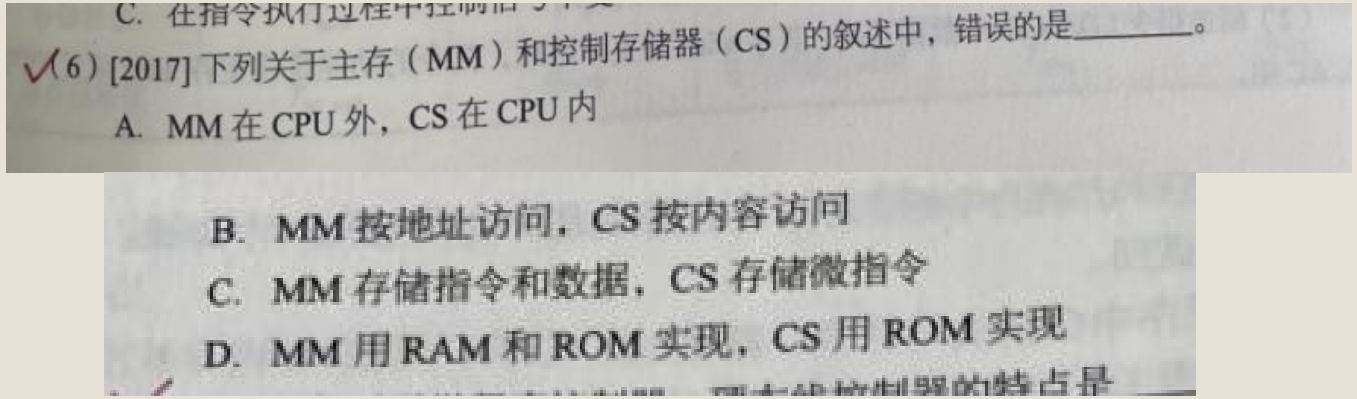
CPU从内存中取出并执行一条指令所需的全部时间称为指令周期，指令周期又用若干机器周期表示，一个机器周期又包含若干个时钟周期，所以D错误。

6-2



答：A，原因如下：

由于只能在一个时钟周期内完成取指执行过程，指令执行过程中数据通路的任何资源都不能被重复使用，所以应该是专用数据通路结构。



答：B，原因如下：

CS在CPU内部，用来存放由微指令组成的微程序，按微指令地址进行访问。

6-2

D. MM 用 RAM 和 ROM 实现，[illegible]

(7) [2009] 相对于微程序控制器，硬布线控制器的特点是_____。

A. 指令执行速度慢，指令功能的修改和扩展容易

B. 指令执行速度慢，指令功能的修改和扩展难

C. 指令执行速度快，指令功能的修改和扩展容易

D. 指令执行速度快，指令功能的修改和扩展难

答：D，原因如下：

硬布线控制器采用专门的逻辑电路实现，其速度主要取决于逻辑电路的延迟，因此速度快，但修改和扩展困难，灵活性差。

D. 指令执行速度快，指令功能的修改和扩展难

(8) [2012] 某计算机的控制器采用微程序控制方式，微指令中的操作控制字段采用字段直接编码法，共有 33 个微命令，构成 5 个互斥类，分别包含 7、3、12、5 和 6 个微命令，则操作控制字段至少有_____。

43%

A. 5 位　　B. 6 位　　C. 15 位　　D. 33 位

答：C，原因如下：

互斥性微命令应该分在同一字段，每个字段还要留出一个空状态，表示该字段不发出任何微命令。5个互斥类分别包括7、3、12、5、6个微命令，操作控制字段至少分别需要3、2、4、3、3，一共15位。

6-2

A. 5位　　B. 6位

（9）[2014] 某计算机采用微程序控制器，共有 32 条指令，公共的取指令微程序包含两条微指令，各指令对应的微程序平均由 4 条微指令组成，采用断定法（下址字段法）确定下条微指令地址，则微指令中下址字段的位数至少是_____。

A. 5　　B. 6　　C. 8　　D. 9

答：C，原因如下：

32条机器指令对应的微命令为32x4=128条，而公共取指令微程序还包括两条微指令，所以控制存储器中有130条微指令，所以下址地段至少需要8位。

（10）[2011] 假定不采用 cache 和指令预取技术，且计算机处于“开中断”状态，则在下列有关指令执行的叙述中，错误的是_____。

A. 每个指令周期中 CPU 都至少访问内存一次

B. 每个指令周期一定大于或等于一个 CPU 时钟周期

C. 空操作指令的指令周期中任何寄存器的内容都不会被改变

D. 当前程序在每条指令执行结束时都可能被外部中断打断

答：C，原因如下：

空操作指令的指令周期中PC的值会修改。

6-3

（1）CPU的基本功能是什么？从实现其功能的角度分析，它应由哪些部分组成？
答：CPU的基本功能是：1.程序控制：控制程序中指令执行的顺序。2.操作控制：产生指令执行过程中需要的操作控制信号。3.时序控制：对每个操作控制信号进行定时。4.数据加工：对数据进行算术、逻辑运算。5.中断处理：及时响应内部异常和外部中断请求。

CPU主要由控制器和运算器两个部分组成。运算器是执行部件，由算术逻辑单元和各种寄存器组成。

（2）CPU内有哪些寄存器？功能分别是什么？哪些是程序员可见的？哪些是必需的？
答：1.程序计数器PC：保存将要执行指令的字节地址。
2.存储器地址寄存器AR：通常用来保存CPU访问主存的单元地址。
3.存储器数据寄存器DR：用于存放从主存中读出的数据或准备写入主存的数据。
4.指令寄存器IR：用于保存当前正在执行的指令。
5.通用寄存器组GR：运算器内部的若干寄存器。
6.程序状态字寄存器PSW：用于保存由运算指令创建的各种条件标志。
通用寄存器组，程序状态字寄存器，程序计数器是程序员可见的。
PC，GR，PSW一般是必需的。

6-3

（3）什么是取指周期？取指周期内应完成哪些操作？
答：取指周期就是从开始取指令到取指令完成所需要的时间。取指周期要完成两个方面的操作，首先是利用PC值作为地址访问主存中的指令，其次是形成后续指令地址。

（4）计算机为什么要设置时序系统？说明指令周期、机器周期和时钟周期的含义。
答：指令执行过程中的所有操作必须遵守严格的顺序，对这些操作的开始时间、持续时间有着严格的限制，因此在计算机系统中要设置时序系统，对指令执行过程中的所有控制信号进行时间控制，以保证指令功能的正确实现。
指令周期：从开始取指令到取指令完成所需要的时间。
机器周期：一般把从内存里面读取一条指令的最短时间（与数据通路相关），规定为机器周期。
时钟周期：定义为主频的倒数，有时也称之为节拍（pulse）。

（5）简述传统三级时序和现代时序的差异。
答：传统三级时序系统采用状态周期、节拍电位和节拍脉冲来对操作信号进行定时控制。传统三级时序系统的核心部件是时序产生器，内部包含一个状态机，用于产生状态周期电位和节拍电位。传统三级时序系统的优点是设计相对简单，缺点是存在一些时间浪费。
现代时序是指每条指令的执行仅仅需要若干个时钟周期，每个时钟周期对应一个状态。现代时序的核心部件是状态寄存器和状态机组合逻辑，用于产生操作控制信号。现代时序的优点是更加灵活和高效，可以根据不同的指令和反馈信号进行状态转换，缺点是设计相对复杂，需要考虑更多的状态和逻辑。

6-3

（6）比较单周期MIPS处理器与多周期MIPS处理器的差异。

答：

单周期处理器：所有指令在一个时钟周期内完成，指令执行过程中数据通路任何资源都不能被重复使用，都应该是专用数据通路；会被多次使用的资源需要设置多个；为避免访存冲突，指令存储器和数据存储器要单独设置。

多周期处理器：指令周期包括多个时钟周期，每个时钟周期执行指令的一部分，并将操作结果暂存在相关寄存器中供下一个时钟周期处理，直至指令执行完毕；功能部件可在同一指令的不同时钟周期被多次使用，共享复用能提高硬件实现效率，指令存储器和数据存储器不需分开设置。

（7）组合逻辑控制器与微程序控制器各有什么特点？

答：组合逻辑控制器又称为硬布线控制器，控制器由各种类型的逻辑门电路和触发器等构成。与微程序控制器相比，组合逻辑控制器具有速度快的特点，但组合逻辑控制器结构复杂，指令功能的修改和扩展较为困难。

微程序控制器的设计采用了存储技术和程序设计技术，使复杂的控制逻辑得到简化。计算机通过读出存放在微程序控制器中微指令产生指令执行过程种所需要的控制信号，与硬布线控制器相比，微程序控制器的速度较慢。

6-3

（8）说明程序与微程序、指令与微指令的异同。

答：微程序是多条微指令的集合，用于实现指令的功能，属于机器指令级别，对用户透明，存放在CPU内部的控制存储器中；程序则是为了完成某一应用功能所编写的指令（包括机器语言指令或高级语言指令）集合，运行时存放在计算机的主存中。

指令是指挥计算机执行某种功能的命令，是构成程序的基本单位，由操作码和地址字段构成，而微指令则用于微程序控制器中产生指令执行过程中所需要的微命令，是构成微程序的基本单位，由操作控制字段、判断测试字段和下地址字段组成。

（9）微命令有几种编码方法？它们是如何实现的？

答：

1.直接表示法：微指令操作控制字段的每一位都直接表示一个微命令，该位为“1”，表示执行这个微命令。

2.编码表示法：将微指令格式中的互斥性微命令分成若干组，一个组对应一个字段，各组的微命令信号均是互斥的，各字段通过译码器生成微命令信号，经时间同步后再去控制相应数据通路中的部件。

3.混合表示法：将直接表示法和编码表示法混合使用，以便在微指令字长、并行性及执行速度和灵活性等方面进行折中，发挥它们的共同优点。

6-3

（10）简述微程序控制器和硬布线控制器的设计方法。

答：微程序控制器设计方法如下：

1.分析指令执行的数据通路，列出每条指令在所有寻址方式下的执行操作流程和每一个需要的控制信号。

2.对指令的操作流程进行细化，将每条指令的每个微操作分配到具体机器周期的各个时间节拍信号上。

3.以时钟周期为单位，构建指令执行的状态图。

4.设计微指令格式和微命令编码方式。

5.根据指令执行状态图编制每条指令的微程序，并存放到控制存储器中。

6.根据微程序组织方式构建微程序控制器中的地址转移逻辑、微地址寄存器、控制存储器间的通路，实现微程序控制器。

硬布线控制器设计方法如下：

1.分析指令执行的数据通路，列出每条指令在所有寻址方式下的执行操作流程和每一个需要的控制信号。

2.对指令的操作流程进行细化，将每条指令的每个微操作分配到具体机器周期的各个时间节拍信号上。

3.根据控制信号同步控制方式构建合适的时序发生器。

4.对每一个控制信号进行逻辑综合，得到每个控制信号的逻辑表达式；

5.采用逻辑门或PLA或ROM实现逻辑表达式的功能。

6-3

（11）简述CPU内部异常与外部中断的区别。
答：1.产生源不同：异常是由CPU内部事件所引起的中断，如程序出错（非法指令、地址越界）等；外部中断是由外部设备事件所引起的中断，如磁盘中断、打印机中断等。2.处理方式不同：异常通常需要修正错误或终止程序，而外部中断通常需要响应设备的请求或完成设备的操作。3.优先级不同：异常的优先级高于外部中断，因为异常涉及到程序的正确性和安全性，而外部中断可以稍后处理或屏蔽。4.返回地址不同：异常的返回地址是引起异常的指令，因为该指令可能需要重新执行或跳过；外部中断的返回地址是发生中断时的下一条指令，因为该指令没有执行过。

（12）简述异常和中断处理的一般流程。
答：异常和中断处理的一般流程如下：
中断请求：外部设备或内部事件向CPU发出中断信号，表示需要CPU的服务。
中断判优：CPU在每条指令执行完后检查是否有中断信号，如果有多个中断信号，根据优先级选择一个最高优先级的中断进行响应。
中断响应：CPU暂停当前程序的执行，保存现场（如程序计数器、状态寄存器等），并根据中断向量表找到对应的中断处理程序的入口地址。
中断处理：CPU切换到相应的处理器模式，执行中断处理程序，完成对中断源的服务。
中断返回：CPU恢复现场，返回到被中断的程序继续执行。
异常处理的流程与中断处理类似，只是异常是由CPU内部产生的，不需要判优和请求，而是直接响应。异常处理程序通常会根据异常的类型和原因进行相应的处理，如修正错误、终止程序、调用操作系统等。

6-3

（13）要支持异常与中断处理，CPU需要对硬、软件进行哪些扩展？

答：

硬件扩展：CPU需要增加一些引脚和寄存器来响应和处理异常与中断，如INTR、NMI、CPSR、SPSR等。CPU还需要增加一些逻辑电路来检测和优先级裁决异常与中断，如中断控制器、优先级解析器等。CPU内部还需要设置中断使能寄存器IE用于开/关中断，外部请求会与IE逻辑与后送CPU，关中断后CPU无法接受外部中断请求。中断响应阶段需要保存断点、硬件关中断和中断识别。

软件扩展：CPU需要提供一些异常向量表和异常处理程序来响应不同类型的异常与中断。另外需要设置保存现场的堆栈，对MIPS结构需要增加EPC寄存器，程序中需要设置好堆栈指针sp。

6-4

6.4 某 CPU 的结构如图 6.69 所示，其中 AC 为累加器，条件状态寄存器保存指令执行过程中的状态。a、b、c、d 为 4 个寄存器。图中箭头表示信息传送的方向，试完成下列各题。

（1）根据 CPU 的功能和结构标明图中 4 个寄存器的名称。

（2）简述指令 LDA addr 的数据通路，其中 addr 为主存地址，指令的功能是将主存 addr 单元的内容送入 AC 中。

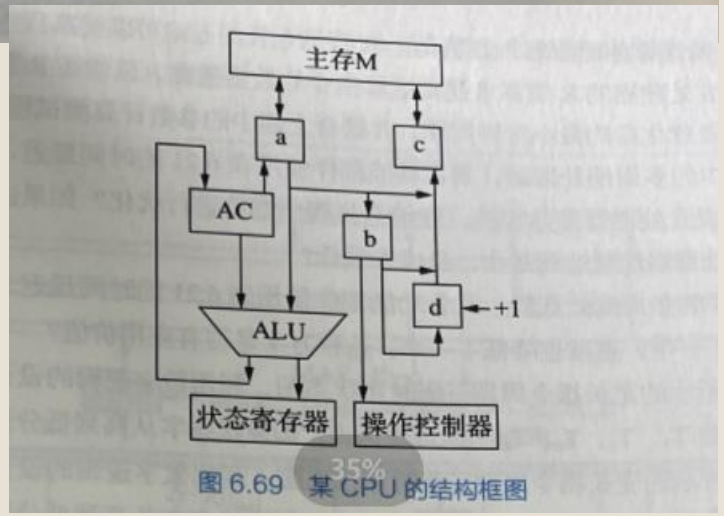


图 6.69 某 CPU 的结构框图

答：

（1）a是DR，b是IR，c是AR，d是PC

（2）取指阶段数据通路：

PC->AR->主存M->DR->IR；PC=PC+1

执行阶段数据通路：

IR（A）->AR->主存M->DR->AC

6-5

6.5 修改图 6.8 所示的单总线结构处理器，使其能够支持如下 MIPS 指令，具体指令功能请查阅 MIPS32 指令手册。试描述需要增加或修改哪些数据通路和控制信号，尝试给出各指令的执行流程和每一步的操作控制信号。

（1）sll；（2）lui；（3）bltz；（4）j。

答：

需要增加：

IR增加一个控制信号，该控制信号将IR中的shamt字段输出到总线

运算器ALU增加SLL控制信号，该控制信号使ALU进行逻辑左移操作

该指令的执行周期流程为：

T1：R[rt] -> X　　控制信号：Rs/Rt=1，R_{out}=1，X_{in}=1

T2：R[rt]<<shamt -> Z　　控制信号：$IR(shamt)_{out}$=1，SLL=1

T3：Z -> R[rd]　　控制信号：Z_{out}=1，RegDst=1，R_{in}=1

6-7

6.7 修改图 6.25 所[illegible]单周期 MIPS 处理器，使其能够支持如下 MIPS 指令，具体指令功能请查阅 MIPS32 指令手册。试描述需要增加或修改哪些数据通路和控制信号，尝试给出各指令的执行流程和每一步的操作控制信号。

（1）srl；（2）lui；（3）blez；（4）jal。

答：

需要增加：

ALU需要增加功能：ALUResult=SrcB

该指令的执行周期流程为：

取指周期：PC->AR, $PC_{out}=AR_{in}=1$;

PC->PC+4;

M[AR]->DR, Read=$DRE_{in}=1$;

DR->IR, DREout=$IR_{in}=1$

执行周期：IR(I)->R[rd], $IR(I)_{out}=X_{in}=1$

6-9

6.9 修改图 6.27 所示的多周期 MIPS 处理器，使其能够支持如下 MIPS 指令，具体指令功能请查阅 MIPS32 指令手册。试描述需要增加修改哪些数据通路和控制信号，尝试给出各指令的执行流程和每一步的操作控制信号。

（1）sra；（2）lui；（3）bgtz；（4）j。

答：需要增加：

（1）ALU增加减法运算的功能，控制信号=SUB

（2）ALU的标志位：SF（符号标志）和ZF（零标志）

（3）4选1MUX的输出接1个2选1MUX，该MUX的另一个输入为0

（4）控制器增加bgtz指令译码输出信号Bgtz

（5）ALU的标志位SF和ZF接或非门（SF=0表示>=0，ZF=0表示≠0；即>0），再与控制器的Bgtz信号进行与

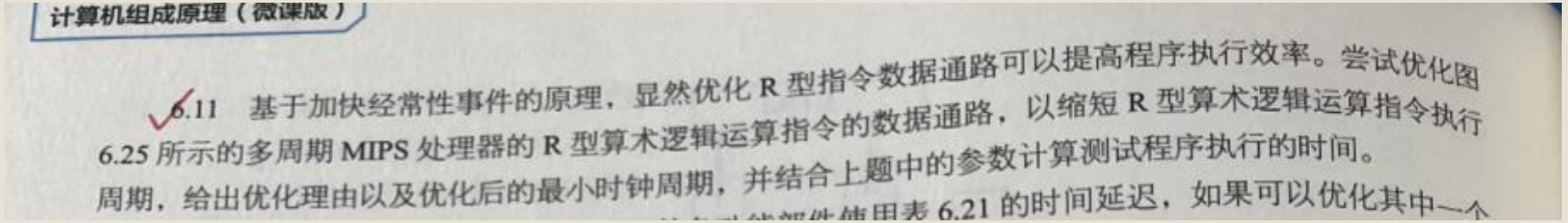
该指令的执行周期流程为：

取指阶段T1： M[PC] -> IR PC+4 -> PC 控制信号：IorD=0、MeMWrite=0、IRWrite=1、AluCtrl=0、AluSrcA=0、AluSrcB=1、AluSrc=0、PCSrc=0、PCWrite=1

译码/取数阶段T2：R[rs] -> A R[rt] -> B PC+SignImm<<2 -> PC 控制信号：AluSrcA=0、AluSrcB=3、AluCtrl=0

执行周期：A -> SrcA 0 -> SrcB ALU做减法运算 如果结果>0，则 PC+4+SignExt(imm)<<2 -> PC；控制信号：ALUSrcA=1，ALUSrcBZero=1，SUB=1，Bgtz=1

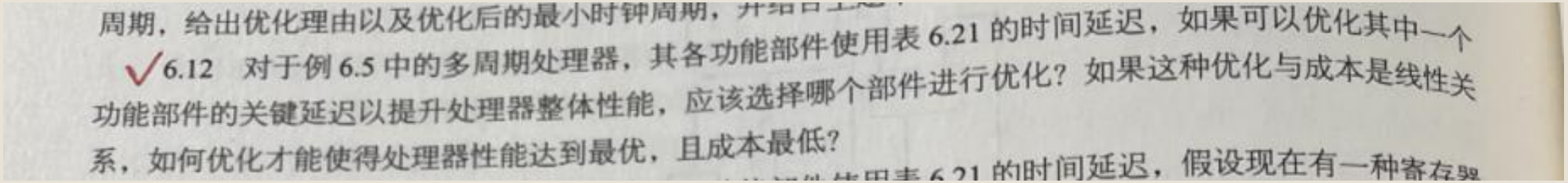
6-11



答：R型运算指令中计算结果需要先缓存在C寄存器中，再送通用寄存器组写回；直接将ALU运算结果送寄存器写回数据段，这样可以减少一个时钟。修改后lw、sw、beq、R型运算和I型运算指令的CPI分别为5、4、3、3、4，因此CPI=5*0.1+4*0.1+3*0.1+3*0.5+4*0.2=3.5

T= 1000 * 10^8 * 3.5 * 200 * 10^(-12) =70s

6-12



周期，给出优化理由以及优化后的最小时钟周期，并给出理由。

√6.12 对于例 6.5 中的多周期处理器，其各功能部件使用表 6.21 的时间延迟，如果可以优化其中一个功能部件的关键延迟以提升处理器整体性能，应该选择哪个部件进行优化？如果这种优化与成本是线性关系，如何优化才能使得处理器性能达到最优，且成本最低？

…件使用表 6.21 的时间延迟，假设现在有一种寄存器

答：Tmin=Tclk_to_q+Tmux+max(Talu+Tmux,Tmem)+Tsetup

从上式可以看出，应该选择存储器进行优化，减少存储器延迟可以提升性能，当存储器延迟为Talu+Tmux=110ps时，性能优化到达极限，此时成本最低。

6-20

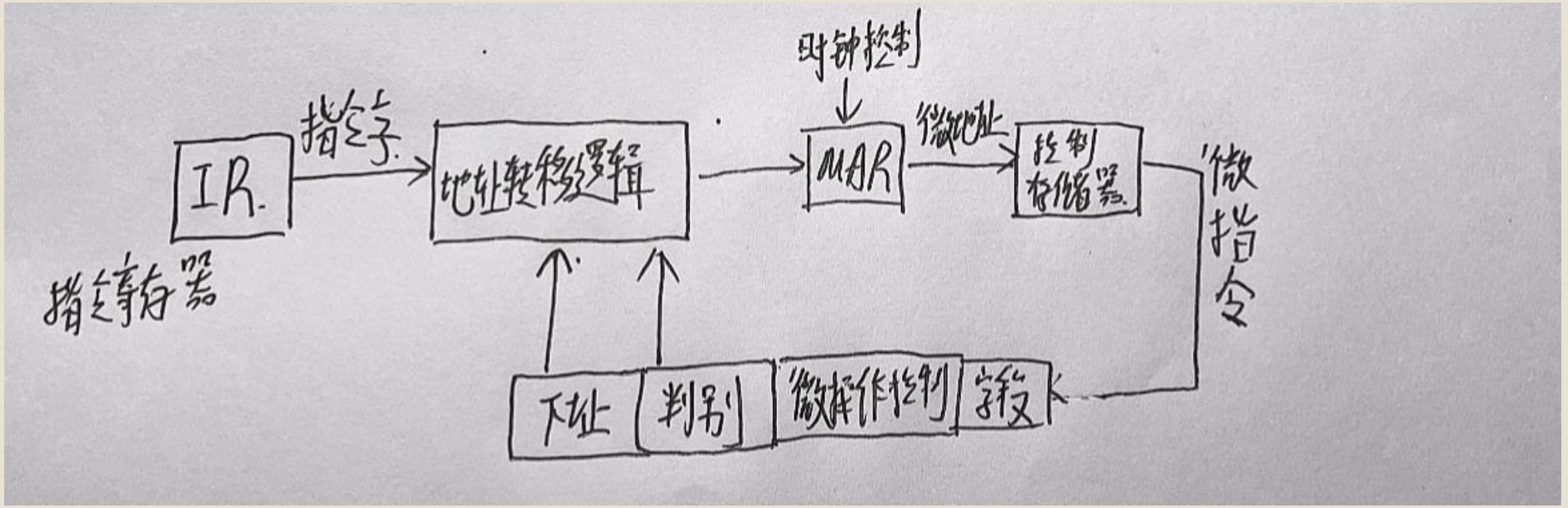
6.20 已知某计算机采用微程序控制方式，控制存储器容量为 128×32 位。微程序可在整个控制存储器中实现分支跳转，控制微程序判别测试条件共 3 个，微指令采用水平型格式，后续微指令地址采用下址字段法。回答下列问题。

（1）微指令的 3 个字段分别应为多少位？

（2）画出对应这种微指令格式的微程序控制器逻辑框图。

答：（1）控制存储器容量128*32=2^7*32位，因此下址地段为7位，判别测试条件为3位，所以操作控制字段为22位

（2）



6-21

√6.21 某微程序包含 5 条微指令，每条微指令发出的操作控制信号如表 6.22 所示，试对这些微指令进行编码，要求微指令的控制字段最短且能保持微指令应有的并行性。

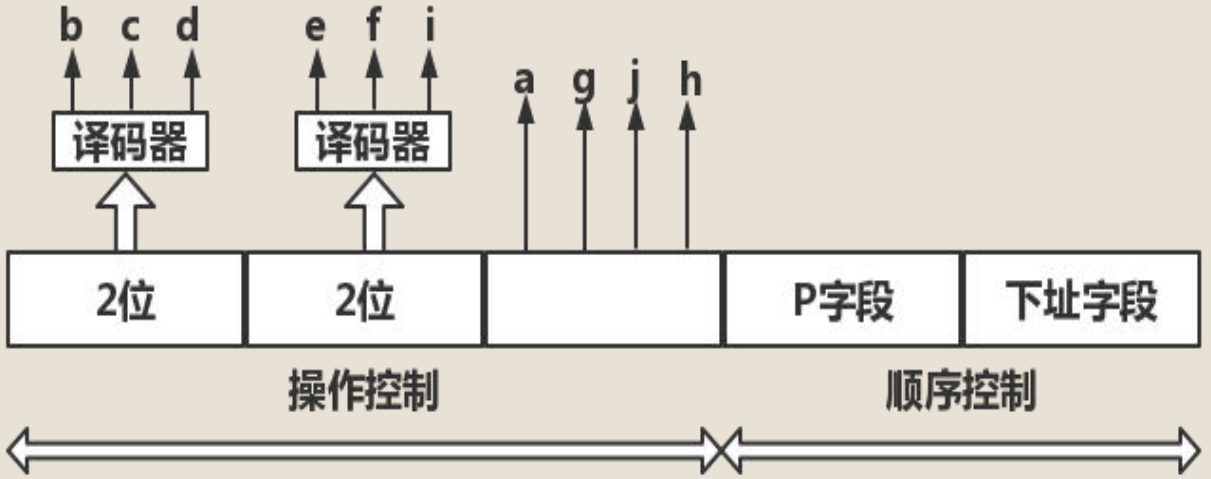
表 6.22 微指令及其对应的微操作控制信号

微指令	微操作控制信号	微指令	微操作控制信号	微指令	微操作控制信号
μI_1	a,c,e,g	μI_3	a,d,e	μI_5	a,d,f,j
μI_2	a,d,f,h,j	μI_4	a,b,i		

答：

	a	b	c	d	e	f	g	h	i	j
μI1	√		√		√		√			
μI2	√			√		√		√		√
μI3	√			√	√					
μI4	√	√							√	
μI5	√			√		√				√

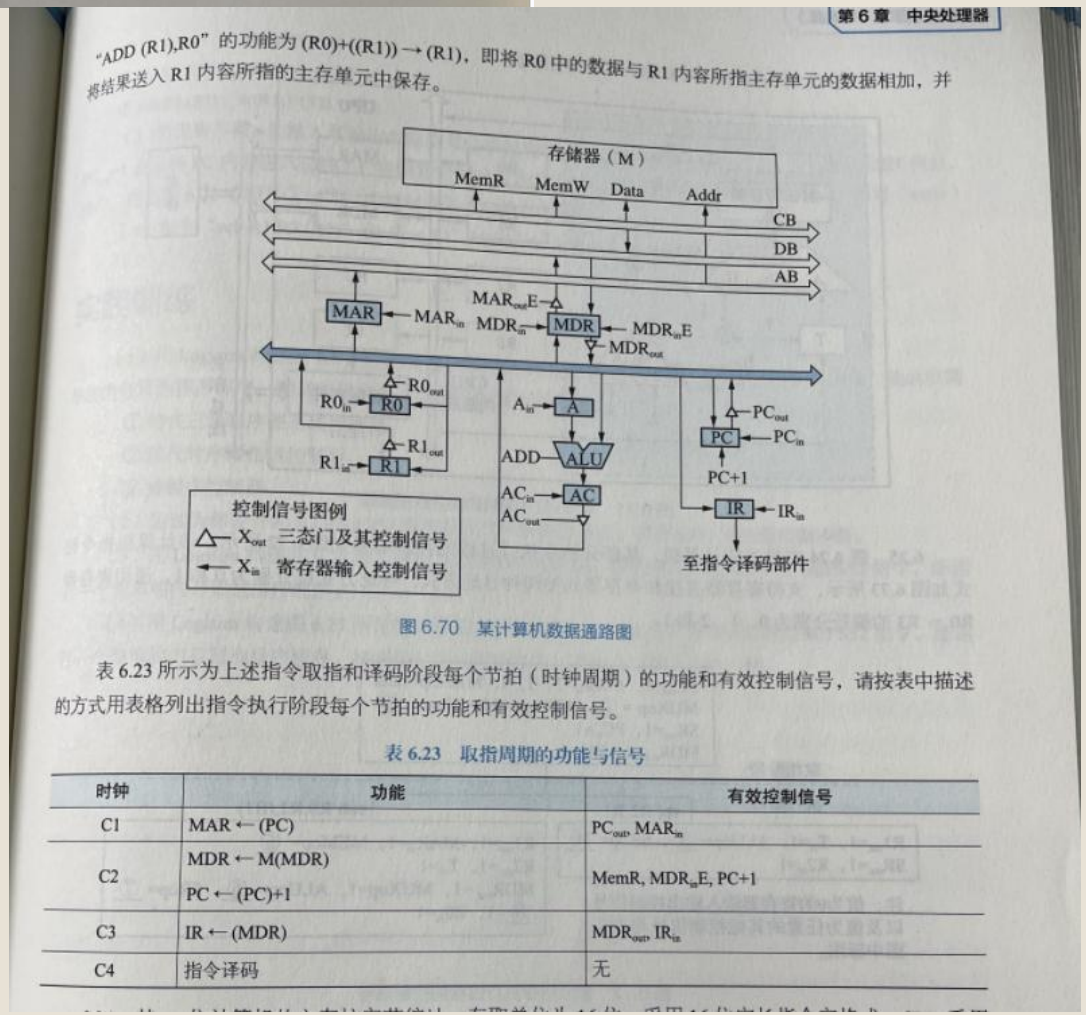
编码如图：



6-23

6.23 某计算机字长为 16 位，采用 16 位定长指令字结构，部分数据通路结构如图 6.70 所示，图中所有控制信号为 1 时表示有效、为 0 时表示无效。例如，控制信号 $MDR_{in}E$ 为 1 表示允许数据从 DB 送入 MDR 中，MDR_{in} 为 1 表示允许数据从内总线送入 MDR 中。假设 MAR 的输出一直处于使能状态。加法指令

250



第 6 章 中央处理器

"ADD (R1),R0" 的功能为 (R0)+((R1)) → (R1)，即将 R0 中的数据与 R1 内容所指主存单元的数据相加，并将结果送入 R1 内容所指的主存单元中保存。

图 6.70 某计算机数据通路图

表 6.23 所示为上述指令取指和译码阶段每个节拍（时钟周期）的功能和有效控制信号，请按表中描述的方式用表格列出指令执行阶段每个节拍的功能和有效控制信号。

表 6.23 取指周期的功能与信号

时钟	功能	有效控制信号
C1	MAR ← (PC)	PC_{out}, MAR_{in}
C2	MDR ← M(MDR) PC ← (PC)+1	MemR, $MDR_{in}E$, PC+1
C3	IR ← (MDR)	MDR_{out}, IR_{in}
C4	指令译码	无

答：

时钟	功能	有效控制信号
C5	MAR（R1）	$R1_{out}$，MAR_{in}
C6	MDR<-M(MAR) A<-(R0)	MemR，$MDR_{in}E$， $R0_{out}$，AC_{in}
C7	AC<-(MDR)+(A)	AC_{in}，MDR_{out}，Add
C8	MDR<-(AC)	AC_{out}，MDR_{in}
C9	M(MAR)<-(MDR)	$MDR_{out}E$，MemW