6.2

1. B

可见寄存器是指该寄存器可以通过汇编指令进行访问。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，程序状态寄存器也是可见寄存器。而IR、MAR、MDR则不可被机器指令控制，是不可见寄存器。

1. B

源操作数采用了寄存器、寄存器简介寻址方式，因此在取数阶段需要用到通用寄存器组和存储器，在执行阶段两个源操作数相加需要用到算术逻辑单元。而指令译码器用于对操作码进行译码，产生指令译码信号，在取数及执行阶段用不到。

1. B

PC存放的是下一条指令字的主存地址，通常位宽与主存地址总线相同。主存4GB=2^32Byte，如果按字节编址，PC长度为32位，但题中指令按字边界（4Byte）对齐，所以PC可以按字编址，也就是只需要30位即可。注意在实际取指令时需要将PC中字地址左移两位变成字节地址才能访存。IR（指令寄存器）用于存放当前正在执行的指令，位宽和指令字长相同，这里指令字长为定长32位，所以IR应该是32位。

1. D

时钟脉冲信号的宽度称为时钟周期，时钟周期是CPU工作的最小时间单位，时钟周期的倒数为机器主频。时钟脉冲信号是由机器脉冲源发出的脉冲信号经整形和分频后形成的。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来确定。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，指令周期又由若干机器周期来表示，一个机器周期又包含若干时钟周期，故D错误。

1. A

单周期处理器CPI=1，时钟周期取决于执行速度最慢的指令，相对多周期处理器，其时钟频率较低。单周期处理器控制器为组合逻辑电路，控制信号在一个时钟周期保持不变。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，所以是专用数据通路结构，而不能采用单总线结构数据通路。

1. B

主存在CPU外部，用于存储指令和数据，它由RAM（随机存取存储器）和ROM（只读存储器）构成。控制存储器在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，所以B是错的。在CPU运行时是只读存储器，现代CPU为了维护方便，采用了可改写控制存储器，方便对处理器升级打补丁。D不严谨。

1. D

微程序控制器采用了“存储技术”和“程序设计技术”，其是一种软件时序，可以使复杂的控制逻辑得到简化。一条机器指令会对应一段微程序的执行，而一段微程序包括多条微指令，需要反复访问控制存储器，所以微程序速度相对较慢。硬件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。

1. C

字段字节编码法中相容性微命令分在不同字段中，而互斥性微命令应分在同一字段，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个互斥类分别包含7、3、12、5和6个微命令，操作控制字段分别需要3、2、4、3、3位，共15位。

1. C

32条机器指令对应的微指令为32\*4=128条，而公共取指令微程序还包括两条微指令，控制存储器中微指令的条数共128+2=130条，所以下址字段至少需要8（2^8=256）位才能寻址到130条微命令。

1. C

不采用cache和指令预取技术，因此每次取指令都至少要访问内存一次（多字长指令可能需要多个存储周期）。指令周期至少包括取指周期和执行周期，所以指令周期一定大于或等于一个CPU时钟周期。空操作指令的执行会引起PC寄存器的修改。“开中断”模式下，CPU能接收到所有可屏蔽中断请求，所以每条指令执行结束都可能被外部中断打断。

6.3

1.答：CPU的五大基本功能如下：

（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型号

（3）时序控制：指对每个操作控制型号进行定时。

（4）数据加工：对数据进行算数、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构成。控制器的主要功能包括取指令、计算下一条指令的地址、对指令译码、产生相应的操作控制信号、控制指令执行所需的数据通路。运算器是执行部件，由算数逻辑单元和各种寄存器组成。

1. 答：CPU主要有以下寄存器：
2. 程序计数器PC：保存将要执行指令的字节地址。
3. 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。
4. 存储器数据寄存器DR：用于存放从从主存中读出的数据或准备写入主存的数据。
5. 指令寄存器IR：用于保存当前正在执行的指令。
6. 通用寄存器组GR：运算器内部的若干寄存器，又称寄存器堆。
7. 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。

寄存器的设置与指令集及具体实现方式有较大关系，其中AR、DR、IR寄存器并不是必需的。另外运算器内部的通用寄存器组GR和程序状态字寄存器PSW属于用户可见存储器，在汇编编程时可以直接使用。

1. 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。
2. 顺序寻址时，将PC内容加当前指令所占用的主存字节数。（2）跳跃寻址时，根据寻址方式、转移条件、转移目标地址等内容计算得到。
3. 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、持续时间有严格的限制，因此在计算机系统中需要设置时序系统，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。

通常将一条指令从取出到执行完成所需要的时间称为指令周期，该**指令周期包括取指 周期和执行周期**；机器周期是指从主存取出一条指令的最短时间(指令周期通常由若干机器周期组成，指令周期具体包含的机器周期数量随指令功能、寻址方式、数据通路的不同而不同；一个机器周期分成若干个时钟节拍，通常以CPU完成一次微操作所需要的时间来定义节拍电位的长度)；时钟周期是CPU工作的最小时间单位，也称节拍脉冲。

1. 答：传统三级时序采用状态周期、节拍电位和节拍脉冲来对操作控制信号进行定时控制。其中状态周期用电位来表示当前处于指令执行的哪个机器周期，节拍电位用电位表示当前处于机器周期的第几个节拍。采用三级时序的好处是可以简化控制器的设计。完成了时序发生器的设计后，所有控制信号都是状态周期电位、节拍电位、指令译码信号、状态反馈信号的组合逻辑。

现代时序系统的定时信号就是基本时钟，一个时钟周期就是一个节拍，指令需要多少时钟周期就分院多少个时钟周期。其采用有限状态机来描述指令的执行过程，将不同指执行的每个时钟周期均对应一个状态，每一个状态会对应特定的微操作控制信号。控制器的核心模块是有限状态机，由一个状态寄存器和有限状态机组合逻辑控制单元构成。有限状态机组合逻辑控制单元的输入包括现态(来自状态寄存器输出)、指令的译码信号和反馈信号，输出为次态，送入状态寄存器输人端，在时钟信号的作用下输入状态寄存器中，为下一时刻的现态；所有操作控制信号的输出只与现态有关。

1. 答：二者的差异主要是指令周期长度、数据通路的区别。单周期处理器中所有指令在一个时钟周期内完成，如指令的取出和执行操作，指令执行过程中数据通路的任何资源都不能被重复使用，都应该是专用数据通路，而需要被多次使用的资源都需要设置多个，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，一条指令的执行过程细分为若干个更小的步骤，每个时钟周期执行其中一部分操作，并将操作结果暂存在相关寄存器中供下一个时钟周期进行处理，直至指令执行完毕。多周期数据通路中的功能部件可在一条指令执行过程的不同时钟周期中被多次使用，这种共享复用方式能提高硬件实现效率，所以多周期指令存储器和数据存储器不需要分开设置。

1. 答：硬布线控制器又称组合逻辑控制器，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结构复杂但速度快的特点，但其指令功能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中所需要的控制信号，与硬布线控制器相比，微程序控制器的速度较慢。

1. 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、判别测试字段和下地址字段等组成。

1. 答： 微命令编码方法有直接表示法、编码表示法及混合表示法3种。

（1）直接表示法将**微指令**操作控制字段的每个二进制位定义为一条微命令，一条微指令从 控制存储器中取出时，它所包含的微命令可直接用于控制数据通路中的执行部件。这种方法的优点是简单、微操作的并行能力强、操作速度快；缺点是微指令过长。

（2）编码表示法又称字段译码法，其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。编码表示法的优点是能有效缩短微指令的字长。缺点是译码器略微降低了微指令的执行速度。

（3）混合表示法将直接表示法与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。

1. 答：（1）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步需要的控制信号。

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，构建指令执行状态图。

④设计微指令格式、微命令编码方法。

⑤ 根据指令执行状态图编制每条指令的微程序，按照状态机组织微程序开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、控制存储器之间的通路，实现微程序控制器。

（2）(三级时序)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步需要的控制信号。

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具体 时间节拍信号上，即对操作控制信号进行同步控制。

③根据控制信号同步控制方式构造合适的时序发生器。

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、PLA或ROM实现逻辑表达式的功能。

1. 答：（1）内部异常是指CPU内部引起的异常事件，也称为内部中断或软件中断，它可进一步分为故障、自陷和终止3种。异常与指令或硬件有关，产生异常的指令可能需要重新执行，所以异常的断点是当前指令，而不是下一条指令。（2）外部中断是指由外部设备向CPU发出的中断请求(如鼠标单击、按键动作等)，要求CPU 暂停当前正在执行的程序，转去执行为某个外部设备事件服务的中断服务程序，处理完后再返回断点(下一条指令)继续执行。注意外部设备中断的时机是一条指令结束后，指令结束时需要查询是否有外部中断请求。外部中断来自CPU外部，与具体指令无关，是随机事件。
2. 答：异常与中断的处理方式基本一致，区别主要是断点的不同。以外部中断为例，其主要处理流程如下。CPU接收到中断请求，在指令执行结束时CPU要进入中断响应周期进行响应处理。中断响应周期的主要任务是关中断、保存断点和中断识别，中断响应周期内的操作都是由硬件实现的，整个响应周期是不可被打断的，通常这部分的功能称为中断隐指令。中断响应周期结束后CPU就开始从当前PC中取出中断服务程序的第一条指令执行中 断服务程序，直至中断返回;中断服务程序主要包括4个步骤:保护现场、中断服务、恢 复现场和中断返回。

整个中断处理过程是由较、硬件协同实现的，其中中断响应是由硬件实现，中断服务是由软件实现。

1. 答：硬件方而需要能够接收并缓存中断请求，可能需要中断请求寄存器缓存不同的中断请求信号，还需要中断优先编码器进行中断优先级的仲裁以及中断识别，CPU内部还需要设置中断使能寄存器IE用于开/关中断，外部中断请求会和IE逻辑与后送CPU，关中断 CPU无法接收外部中断请求。中断响应阶段需要保存断点、硬件关中断和中断识别。

软件方面需要增加开关中断相关、中断返回指令，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

答：

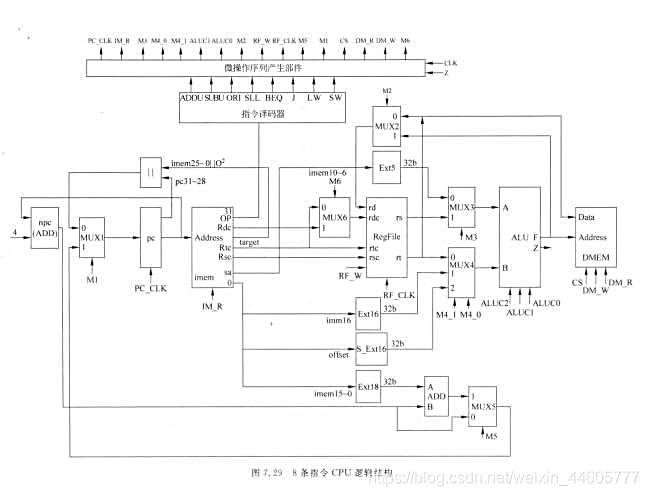
（1）a: DR；b: IR；c: AR；d: PC

（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。

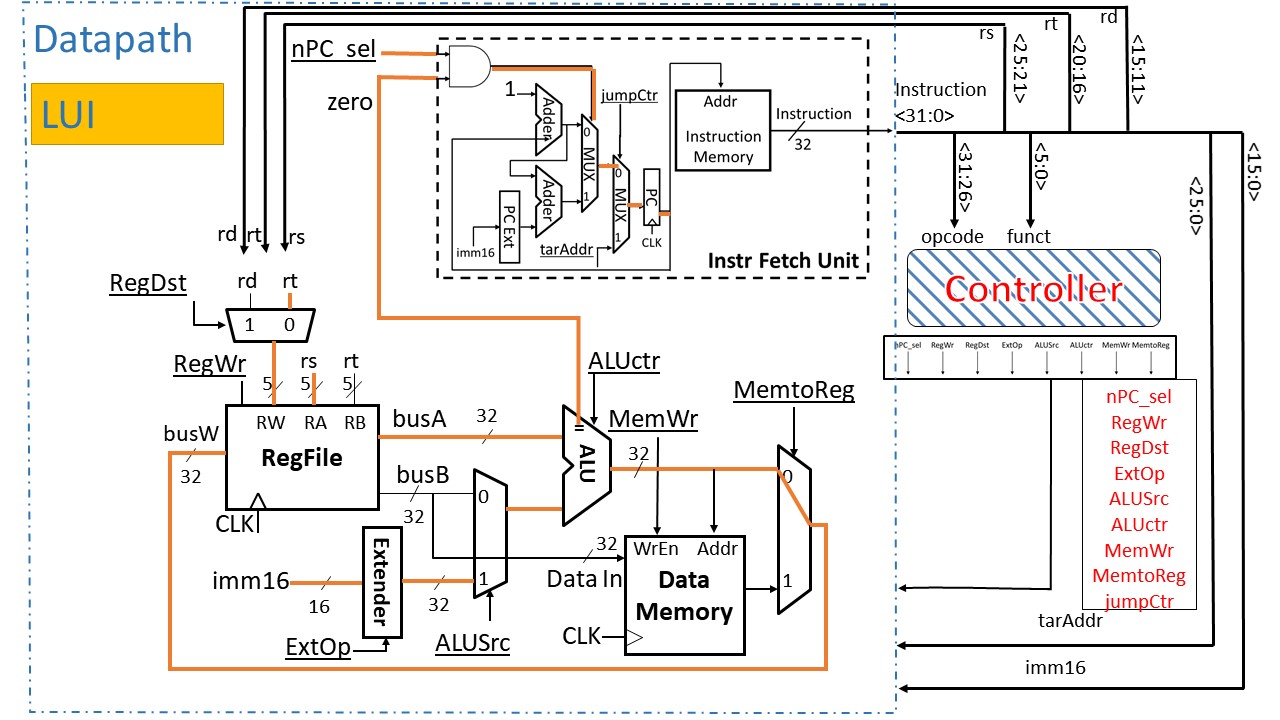
6.5

1. 答：sll（shift left logical）即左移指令，



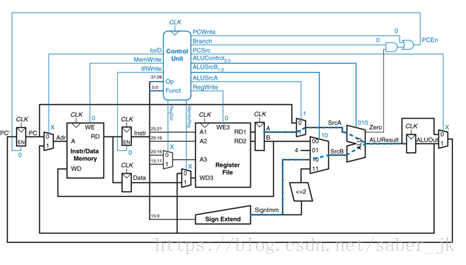
6.7

1. 答：lui指令. 指令用法为：lui rt，immediate。 LUI指令直接就是将立即数付给RT寄存器。



6.9

（3）答：bgtz:大于0转移



6.11

答：R 型运算指令中计算的结果需要先缓存在 C 寄存器中，再送寄存器堆写回，可以直接

将 ALU 运算结果送寄存器写回数据端，这样可以减少一个时钟。修改后lw、sw、beq、R型运算、I型运算指令的CPI分别为5、4、3、4、4，因此CPI=5\*0.1+4\*0.1+3\*0.1+3\*0.5+4\*0.2=3.5，Tcpu=1000\*10^8\*3.5\*200\*10^-12=70s。

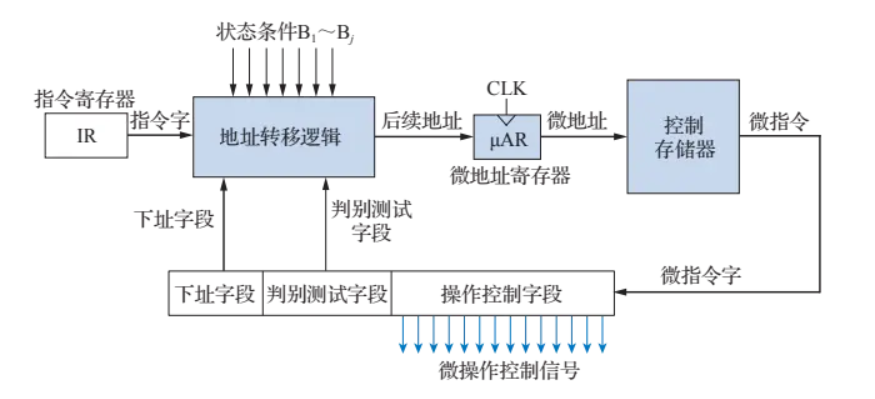
6.12

答：Tmin=Tclk\_to\_q+Tmux+max(Talu+Tmux，Tmem)+Tsetup=20+20+max(90+20，150)+10=200ps。存储器是瓶颈，减少存储器延迟可以提升性能，但当存储器延迟为 110ps 时，性能优化到达极限，此时成本最低。

6.20

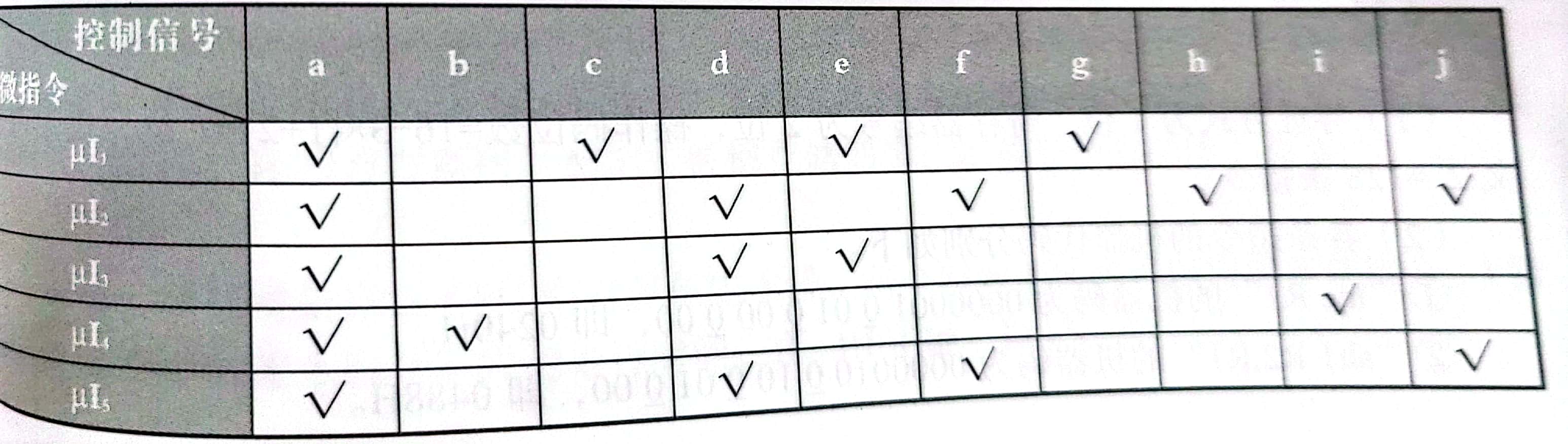
答：（1）控制器存储器容量为128\*32=2^7\*32位，因此下址字段为 7 位，判别测试条件3位，所以操作控制字段=32-7=22位。

（2）

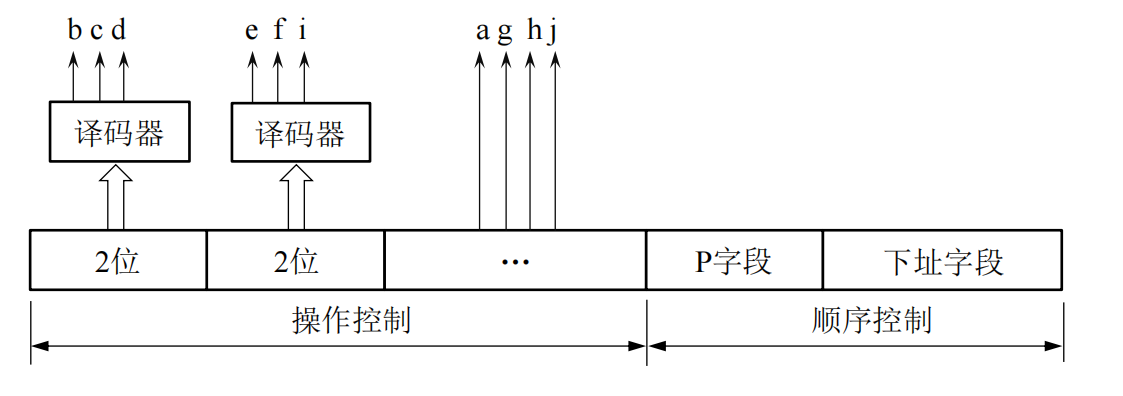


6.21

答：



可以发现两个互斥组（b，c，d），（e，f，i），可以将这两个互斥组采用码，其余的 a，g，h，j 等四个微命令采用直接表示法，具体如下图所示。



6.23

答：

