第 3 章 系统总线

1. 什么是总线？总线传输有何特点？为了减轻总线负载，总线上的部件应具备什么特点？

答： P41.总线是一种能由多个部件分时共享的公共信息传送线路。

总线传输的特点是：某一时刻只允许有一个部件向总线发送信息，但多个部件可以同时从总线上接收相同的信息。

为了减轻总线负载，总线上的部件应通过三态驱动缓冲电路与总线连通。

2. 总线如何分类？什么是系统总线？系统总线又分为几类，它们各有何作用，是单向的，还是双向的，它们与机器字长、存储字长、存储单元有何关系？

答：按照连接部件的不同，总线可以分为片内总线、系统总线和通信总线。

系统总线是连接 CPU、主存、 I/O 各部件之间的信息传输线。

系统总线按照传输信息不同又分为地址线、数据线和控制线。地址线是单向的，其根数越多，寻址空间越大，即 CPU 能访问的存储单元的个数越多；数据线是双向的，其根数与存储字长相同，是机器字长的整数倍。

4. 为什么要设置总线判优控制？常见的集中式总线控制有几种？各有何特点？哪种方式响应时间最快？哪种方式对电路故障最敏感？

答：总线判优控制解决多个部件同时申请总线时的使用权分配问题；

常见的集中式总线控制有三种：链式查询、计数器定时查询、独立请求；

特点：链式查询方式连线简单，易于扩充，对电路故障最敏感；计数器定时查询方式优先级设置较灵活，对故障不敏感， 连线及控制过程较复杂； 独立请求方式速度最快， 但硬件器件用量大， 连线多， 成本较高。

7. 画图说明异步通信中请求与回答有哪几种互锁关系？

答：见 P61-62，图 3.86。

14. 设总线的时钟频率为 **8MHZ** ，一个总线周期等于一个时钟周期。如果一个总线周期中并行传送 **16** 位数据，试问总线的带宽是多少？

解：由于： f=8MHz,T=1/f=1/8M 秒，一个总线周期等于一个时钟周期

所以：总线带宽 =16/（1/8M） = 128Mbps

15. 在一个 **32** 位的总线系统中，总线的时钟频率为 **66MHZ** ，假设总线最短传输周期为 **4** 个时钟周期，试计算总线的最大数据传输率。若想提高数据传输率，可采取什么措施？

解：总线传输周期 =4*1/66M 秒

总线的最大数据传输率 =32/(4/66M)=528Mbps

若想提高数据传输率， 可以提高总线时钟频率、 增大总线宽度或者减少总线传输周期包含的时钟周期个数。

16. 在异步串行传送系统中，字符格式为： **1** 个起始位、 **8** 个数据位、 **1** 个校验位、 **2** 个终止位。若要求每秒传送 **120** 个字符，试求传送的波特率和比特率。

解：一帧包含： 1+8+1+2=12 位；故波特率为： （1+8+1+2）*120=1440bps ；比特率为： 8*120=960bps

第 4 章 存储器

5. 什么是存储器的带宽？若存储器的数据总线宽度为 **32** 位，存取周期为 **200ns**，则存储器的带宽是多少？

解：存储器的带宽指单位时间内从存储器进出信息的最大数量。

存储器带宽 = 1/200ns × 32 位 = 160M 位/秒 = 20MB/秒 = 5M 字/秒

注意：字长 32 位，不是 16 位。（注：$1ns=10^{-9}s$）

7. 一个容量为 **16K×32** 位的存储器，其地址线和数据线的总和是多少？当选用下列不同规格的存储芯片时，各需要多少片？

1K×4 位，2K×8 位，4K×4 位，16K×1 位，4K×8 位，8K×8 位

解：地址线和数据线的总和 = 14 + 32 = 46 根；选择不同的芯片时，各需要的片数为：

1K×4：（16K×32）/（1K×4）= 16 × 8 = 128 片

2K×8：（16K×32）/（2K×8）= 8 × 4 = 32 片

4K×4：（16K×32）/（4K×4）= 4 × 8 = 32 片

16K×1：（16K×32）/（16K×1）= 1 × 32 = 32 片

4K×8：（16K×32）/（4K×8）= 4 × 4 = 16 片

8K×8：（16K×32）/（8K×8）= 2 × 4 = 8 片

15. 设 **CPU** 共有 **16** 根地址线，**8** 根数据线，并用 $\overline{MREQ}$（低电平有效）作访存控制信号，R/W 作读写命令信号（高电平为读，低电平为写）。现有下列存储芯片：**ROM**（**2K×8** 位，**4K×4** 位，**8K×8** 位），**RAM**（**1K×4** 位，**2K×8** 位，**4K×8** 位），及 **74138** 译码器和其他门电路（门电路自定）。试从上述规格中选用合适芯片，画出 **CPU** 和存储芯片的连接图。要求：

（1）最小 4K 地址为系统程序区，4096~16383 地址范围为用户程序区。

（2）指出选用的存储芯片类型及数量。

（3）详细画出片选逻辑。

解：（1）地址空间分配图：

系统程序区（ROM 共 4KB）：0000H-0FFFH

用户程序区（RAM 共 12KB）：1000H-3FFFH

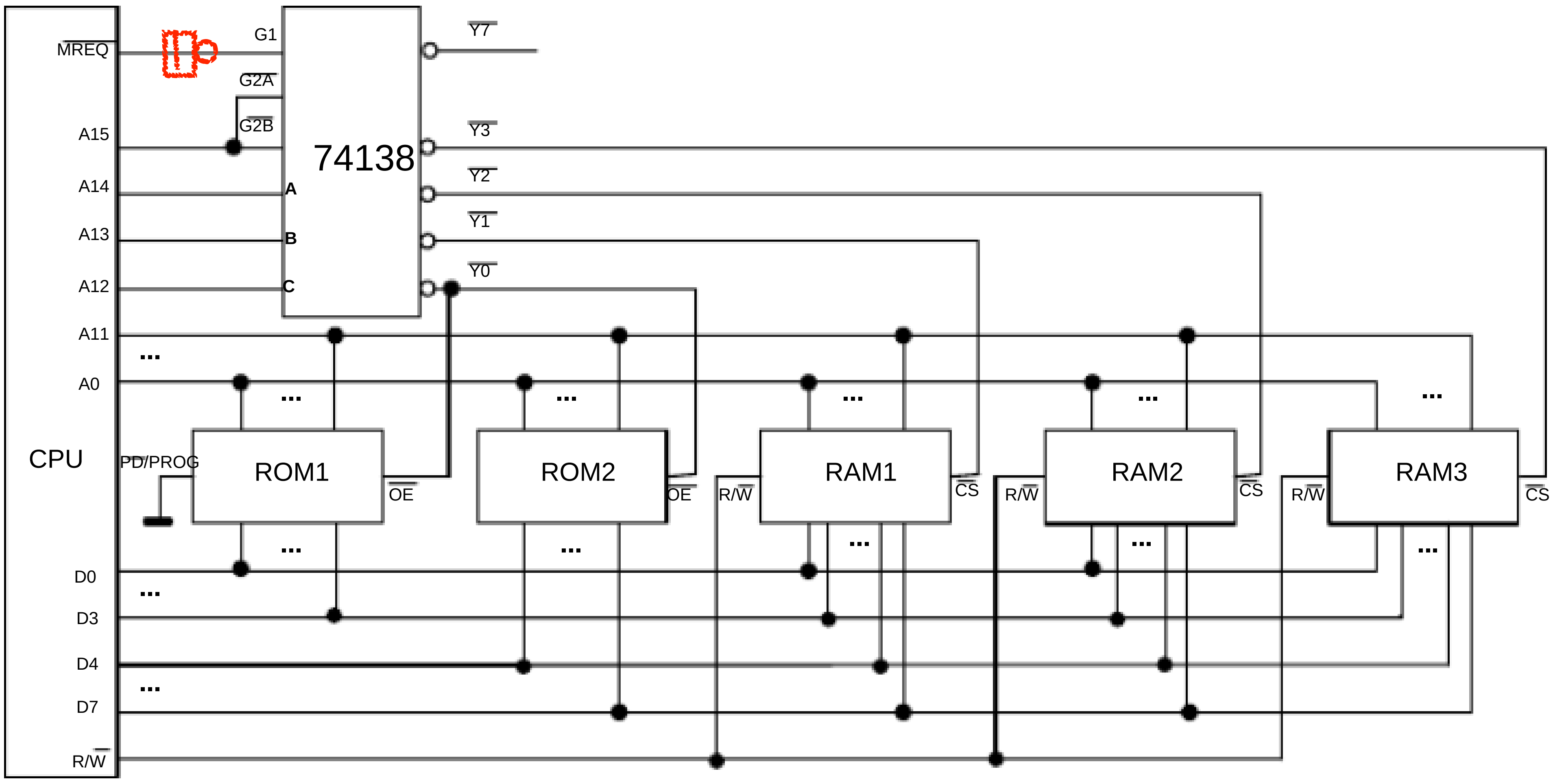
（2）选片：ROM：选择 4K×4 位芯片 2 片，位并联

RAM：选择 4K×8 位芯片 3 片，字串联 (RAM1 地址范围为 :1000H-1FFFH，RAM2 地址范围为 2000H-2FFFH, RAM3 地址范围为 :3000H-3FFFH)

（3）各芯片二进制地址分配如下：

	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
ROM1,2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
RAM1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
RAM2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
RAM3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CPU 和存储器连接逻辑图及片选逻辑如下图 (3) 所示：



图（3）

28. 设主存容量为 **256K** 字，**Cache** 容量为 **2K** 字，块长为 **4**。

（**1**）设计 **Cache** 地址格式，**Cache** 中可装入多少块数据？

（**2**）在直接映射方式下，设计主存地址格式。

（**3**）在四路组相联映射方式下，设计主存地址格式。

（**4**）在全相联映射方式下，设计主存地址格式。

（**5**）若存储字长为 **32** 位，存储器按字节寻址，写出上述三种映射方式下主存的地址格式。

解：（1）Cache 容量为 2K 字，块长为 4，Cache 共有 $2K/4=2^{11}/2^{2}=2^{9}=512$ 块，

Cache 字地址 9 位，字块内地址为 2 位

因此，Cache 地址格式设计如下：

Cache 字块地址（ 9 位）	字块内地址（ 2 位）

（2）主存容量为 256K 字$=2^{18}$字，主存地址共 18 位，共分 $256K/4=2^{16}$ 块，

主存字块标记为 18-9-2=7 位。

直接映射方式下主存地址格式如下：

主存字块标记（ 7 位）	Cache 字块地址（ 9 位）	字块内地址（ 2 位）

（3）根据四路组相联的条件，一组内共有 4 块，得 Cache 共分为 $512/4=128=2^{7}$ 组，

主存字块标记为 18-7-2=9 位，主存地址格式设计如下：

主存字块标记（ 9 位）	组地址（ 7 位）	字块内地址（ 2 位）

（4）在全相联映射方式下，主存字块标记为 18-2=16 位，其地址格式如下：

主存字块标记（ 16 位）	字块内地址（ 2 位）

（5）若存储字长为 32 位，存储器按字节寻址，则主存容量为 $256K*32/4=2^{21}$B，

Cache 容量为 $2K*32/4=2^{14}$B，块长为 $4*32/4=32B=2^{5}$B，字块内地址为 5 位，

在直接映射方式下，主存字块标记为 21-9-5=7 位，主存地址格式为：

主存字块标记（ 7 位）	Cache 字块地址（ 9 位）	字块内地址（ 5 位）

在四路组相联映射方式下，主存字块标记为 21-7-5=9 位，主存地址格式为：

主存字块标记（ 9 位）	组地址（ 7 位）	字块内地址（ 5 位）

在全相联映射方式下，主存字块标记为 21-5=16 位，主存地址格式为：

主存字块标记（ 16 位）	字块内地址（ 5 位）

29. 假设 **CPU** 执行某段程序时共访问 **Cache** 命中 **4800** 次，访问主存 **200** 次，已知 **Cache** 的存取周期为 **30ns**，主存的存取周期为 **150ns**，求 **Cache** 的命中率以及 **Cache-**主存系统的平均访问时间和效率，试问该系统的性能提高了多少倍？

解：Cache 被访问命中率为： 4800/(4800+200)=24/25=96%

则 Cache-主存系统的平均访问时间为： t_a=0.96*30ns+(1-0.96)*150ns=34.8ns

Cache-主存系统的访问效率为： e=t_c/t_a*100%=30/34.8*100%=86.2%

性能为原来的 150ns/34.8ns=4.31 倍，即提高了 3.31 倍。

32. 设某机主存容量为 **4MB**，**Cache** 容量为 **16KB**，每字块有 **8** 个字，每字 **32** 位，设计一个四路组相联映射（即 **Cache** 每组内共有 **4** 个字块）的 **Cache** 组织。

（**1**）画出主存地址字段中各段的位数。

（**2**）设 **Cache** 的初态为空，**CPU** 依次从主存第 **0**，**1**，**2**，…，**89** 号单元读出 **90** 个字（主存一次读出一个字），并重复按此次序读 **8** 次，问命中率是多少？

（**3**）若 **Cache** 的速度是主存的 **6** 倍，试问有 **Cache** 和无 **Cache** 相比，速度约提高多少倍？

解：（1）根据每字块有 8 个字，每字 32 位（4 字节），得出主存地址字段中字块内地址为 3+2=5 位。

根据 Cache 容量为 16KB=2^{14}B，字块大小为 8*32/8=32=2^5B，得 Cache 地址共 14 位，Cache 共有 2^{14-5}=2^9 块。

根据四路组相联映射， Cache 共分为 $2^9/2^2=2^7$ 组。

根据主存容量为 4MB=2^{22}B，得主存地址共 22 位，主存字块标记为 22-7-5=10 位，故主存地址格式为：

主存字块标记（ 10 位）	组地址（ 7 位）	字块内地址（ 5 位）

（2）由于每个字块中有 8 个字，而且初态为空，因此 CPU 读第 0 号单元时，未命中，必须访问主存，同时将该字所在的主存块调入 Cache 第 0 组中的任一块内，接着 CPU 读第 1~7 号单元时均命中。同理，CPU 读第 8，16，…，88 号时均未命中。可见， CPU 在连续读 90 个字中共有 12 次未命中，而后 8 次循环读 90 个字全部命中，命中率为：

$$\frac{90\times 8-12}{90\times 8}=0.984$$

（3）设 Cache 的周期为 t，则主存周期为 6t，没有 Cache 的访问时间为 6t*90*8，有 Cache 的访问时间为 t（90*8-12）+6t*12，则有 Cache 和无 Cache 相比，速度提高的倍数为：

$$\frac{6t\times 90\times 8}{(90\times 8-12)t+6t\times 12}-1\approx 5.54$$

39. 某磁盘存储器转速为 **3000** 转/分，共有 **4** 个记录盘面，每毫米 **5** 道，每道记录信息 **12 288** 字节，最小磁道直径为 **230mm**，共有 **275** 道，求：

（**1**）磁盘存储器的存储容量。

（**2**）最高位密度（最小磁道的位密度）和最低位密度。

（**3**）磁盘数据传输率。

（**4**）平均等待时间。

解：（1）存储容量 = 275 道 ×12 288B/道 ×4 面 = 13 516 800B

（2）最高位密度 = 12 288B/（π×230）= 17B/mm = 136 位/mm（向下取整）

最大磁道直径 =230mm+2× 275 道/(5 道/mm) = 230mm + 110mm = 340mm

最低位密度 = 12 288B /(π×340)= 11B/mm = 92 位 / mm （向下取整）

（3）磁盘数据传输率 = 12 288B ×3000 转/分=12 288B ×50 转/秒=614 400B/s

（4）平均等待时间 = 1s/50 / 2 = 10ms

第 6 章 计算机的运算方法

4. 设机器数字长为8位（含1位符号位在内），写出对应下列各真值的原码、补码和反码。

-13/64，29/128，100，-87

解：真值与不同机器码对应关系如下：

真值		原 码	反 码	补 码
十进制	二进制			
-13/64	-0.00 1101	1.001 1010	1.110 0101	1.110 0110
29/128	0.001 1101	0.001 1101	0.001 1101	0.001 1101
100	110 0100	0,110 0100	0,110 0100	0,110 0100
-87	-101 0111	1,101 0111	1,010 1000	1,010 1001

5. 已知$[x]_{补}$，求$[x]_{原}$和x。

$[x1]_{补}$=1. 1100；$[x2]_{补}$=1. 1001；$[x3]_{补}$=0. 1110；$[x4]_{补}$=1. 0000；
$[x5]_{补}$=1，0101；$[x6]_{补}$=1，1100；$[x7]_{补}$=0，0111；$[x8]_{补}$=1，0000；

解：$[x]_{补}$与$[x]_{原}$、x的对应关系如下：

$[x]_{补}$	$[x]_{原}$	x（二进制）	x（十进制）
1.1100	1.0100	-0.0100	-1/4
1.1001	1.0111	-0.0111	-7/16
0.1110	0.1110	+0.1110	+7/8
1.0000	无	-1.0000	-1
1，0101	1，1011	-1011	-11
1，1100	1，0100	-0100	-4
0，0111	0，0111	+0111	+7
1，0000	无	-10000	-16

9. 当十六进制数9B和FF分别表示为原码、补码、反码、移码和无符号数时，所对应的十进制数各为多少（设机器数采用一位符号位）？

解：真值和机器数的对应关系如下：

十六进制	真值	无符号数	原码	反码	补码	移码
9BH	二进制	1001 1011	-11 011	-1100100	-1100101	+11011
	十进制	155	-27	-100	-101	+27
FFH	二进制	1111 1111	-1111111	-0000000	-0000001	+1111111
	十进制	255	-127	-0	-1	+127

注意：1）9BH、FFH为机器数，本身含符号位。
2）移码符号位与原、补、反码相反，数值同补码。

12. 设浮点数格式为：阶码 **5** 位（含 **1** 位阶符），尾数 **11** 位（含 **1** 位数符）。写出 **51/128**、**-27/1024** 所对应的机器数。要求如下：

（**1**）阶码和尾数均为原码。

（**2**）阶码和尾数均为补码。

（**3**）阶码为移码，尾数为补码。

解：据题意画出该浮点数的格式：

阶符 1 位	阶码 4 位	数符 1 位	尾数 10 位

将十进制数转换为二进制： x1= 51/128= 0.0110011B= 2^{-1} * 0.110 011B

x2= -27/1024= -0.0000011011B = 2^{-5}*(-0.11011B)

则以上各数的浮点规格化数为：

（1）[x1] 浮=1， 0001；0.110 011 000 0

[x2] 浮=1， 0101；1.110 110 000 0

（2）[x1] 浮=1， 1111；0.110 011 000 0

[x2] 浮=1， 1011；1.001 010 000 0

（3）[x1] 浮=0， 1111；0.110 011 000 0

[x2] 浮=0， 1011；1.001 010 000 0

16. 设机器数字长为 **16** 位，写出下列各种情况下它能表示的数的范围。设机器数采用一位符号位，答案均用十进制表示。

（**1**）无符号数；

（**2**）原码表示的定点小数。

（**3**）补码表示的定点小数。

（**4**）补码表示的定点整数。

（**5**）原码表示的定点整数。

（**6**）浮点数的格式为：阶码 **6** 位（含 **1** 位阶符），尾数 **10** 位（含 **1** 位数符）。分别写出其正数和负数的表示范围。

（**7**）浮点数格式同（ **6**），机器数采用补码规格化形式，分别写出其对应的正数和负数的真值范围。

解：（1）无符号整数： 0 —— 2^{16} - 1，即： 0—— 65535；

无符号小数： 0 —— 1 - 2^{-16} ，即： 0 —— 0.99998；

（2）原码定点小数： -1 + 2^{-15}—— 1 - 2^{-15} ，即：-0.99997 —— 0.99997

（3）补码定点小数： - 1—— 1 - 2^{-15} ，即： -1—— 0.99997

（4）补码定点整数： -2^{15}—— 2^{15} - 1 ，即： -32768—— 32767

（5）原码定点整数： -2^{15} + 1—— 2^{15} - 1，即： -32767 —— 32767

（6）据题意画出该浮点数格式，当阶码和尾数均采用原码，非规格化数表示时：

最大负数 = 1， 11 111；1.000 000 001 ，即 $-2^{-9}\times2^{-31}$

最小负数 = 0， 11 111；1.111 111 111，即 $-(1-2^{-9})\times2^{31}$

则负数表示范围为： $-(1-2^{-9})\times2^{31}$ —— $-2^{-9}\times2^{-31}$

最大正数 = 0， 11 111；0.111 111 111，即 $(1-2^{-9})\times2^{31}$

最小正数 = 1， 11 111；0.000 000 001，即 $2^{-9}\times2^{-31}$

则正数表示范围为： $2^{-9}\times2^{-31}$ —— $(1-2^{-9})\times2^{31}$

（7）当机器数采用补码规格化形式时，若不考虑隐藏位，则

最大负数 =1， 00 000；1.011 111 111，即 $-2^{-1}\times2^{-32}$

最小负数 =0， 11 111；1.000 000 000，即 -1×2^{31}

则负数表示范围为： -1×2^{31} —— $-2^{-1}\times2^{-32}$

最大正数 =0， 11 111；0.111 111 111，即 $(1-2^{-9})\times2^{31}$

最小正数 =1， 00 000；0.100 000 000，即 $2^{-1}\times2^{-32}$

则正数表示范围为： $2^{-1}\times2^{-32}$ —— $(1-2^{-9})\times2^{31}$

19. 设机器数字长为 **8** 位（含 **1** 位符号位），用补码运算规则计算下列各题。

（**1**）**A=9/64**， **B=-13/32**，求 **A+B**。

（**2**）**A=19/32**， **B=-17/128**，求 **A-B**。

（**3**）**A=-3/16**， **B=9/32**，求 **A+B**。

（**4**）**A=-87**， **B=53**，求 **A-B**。

（**5**）**A=115**， **B=-24**，求 **A+B**。

解：（1）A=9/64= 0.001 0010B, B= -13/32= -0.011 0100B

[A] 补=0.001 0010, [B] 补=1.100 1100

[A+B] 补= 0.0010010 + 1.1001100 = 1.1011110 —— 无溢出

A+B= -0.010 0010B = -17/64

（2）A=19/32= 0.100 1100B, B= -17/128= -0.001 0001B

[A] 补=0.100 1100, [B] 补=1.110 1111 , [-B] 补=0.001 0001

[A-B] 补= 0.1001100 + 0.0010001= 0.1011101 —— 无溢出

A-B= 0.101 1101B = 93/128B

（3）A= -3/16= -0.001 1000B, B=9/32= 0.010 0100B

$[A]_{补}$=1.110 1000, $[B]_{补}$ = 0.010 0100

$[A+B]_{补}$ = 1.1101000 + 0.0100100 = 0.0001100 —— 无溢出

A+B= 0.000 1100B = 3/32

（4） A= -87= -101 0111B, B=53=110 101B

$[A]_{补}$=1 010 1001, $[B]_{补}$=0 011 0101, $[-B]_{补}$=1 100 1011

$[A-B]_{补}$ = 1 0101001 + 1 1001011 = 0 1110100 —— 溢出

（5）A=115= 111 0011B, B= -24= -11 000B

$[A]_{补}$=0 1110011, $[B]_{补}$=1，110 1000

$[A+B]_{补}$ = 0 1110011 + 1 1101000 = 0 1011011 —— 无溢出

A+B= 101 1011B = 91

A+B= 101 1011B = 91

20. 用原码一位乘、两位乘和补码一位乘（Booth算法）、两位乘计算x·y。

（1）x= 0.110 111，y= -0.101 110；

（2）x= -0.010 111，y= -0.010 101；

（3）x= 19， y= 35；

（4）x= 0.110 11， y= -0.111 01。

解：先将数据转换成所需的机器数，然后计算，最后结果转换成真值。

（1）$[x]_{原}$=x=0.110111，$[y]_{原}$=1.101110

x^*=0.110111， y^*=0.101110

$x_0=0$，$y_0=1$，$z_0=x_0 \oplus y_0=0 \oplus 1=1$

$x^* \times y^*$=0.100 111 100 010

$[x \times y]_{原}$=1.100 111 100 010

x·y= -0. 100 111 100 010

（2） x= -0.010111， y= -0.010101

$[x]_{原}$=1.010111， $[y]_{原}$=1.010101

x^*=0. 010111， y^*=0. 010101

$[-x^*]_{补}$=1.101001，$2x^*$=0.101110

$[-2x^*]_{补}$=1.010010

$x_0=1$，$y_0=1$，$z_0=x_0 \oplus y_0=1 \oplus 1=0$

$[x]_{补}$=1.101001， $[y]_{补}$=1.101011

$[-x]_{补}$=0.010111，$[2x]_{补}$=1.010010

$[-2x]_{补}$=0.101110

$x^* \times y^*$=0.000 111 100 011

$[x \times y]_{原}$=0.000 111 100 011

$[x \times y]_{补}$=0.000 111 100 011 0

x·y= 0. 000 111 100 011

运算过程如下：

（3） x= 19， y= 35

x=（10 011）$_2$，y=（100 011）$_2$

$x^*=[x]_{原}=[x]_{补}$= 0，010 011

$y^*=[y]_{原}=[y]_{补}$= 0，100 011

$[-x^*]_{补}=[-x]_{补}$= 1，101 101

$2x^*=[2x]_{补}$= 0，100 110

$[-2x^*]_{补}=[-2x]_{补}$= 1，011 010

$x_0=0$，$y_0=0$，$z_0=x_0 \oplus y_0=0 \oplus 0=0$

$x \cdot y=x^* \times y^*=[x \times y]_{原}=[x \times y]_{补}$

= 0，001 010 011 001

运算过程如下：

（4） x= 0. 110 11， y= -0.111 01

$x^*=[x]_{原}=[x]_{补}$= 0. 110 11

$[y]_{原}$=1.111 01，y^*=0. 111 01

$[y]_{补}$=1.000 11

$[-x^*]_{补}=[-x]_{补}$= 1.001 01

$2x^*=[2x]_{补}$= 01.101 10

$[-2x^*]_{补}=[-2x]_{补}$= 10.010 10

$x_0=0$，$y_0=1$，$z_0=x_0 \oplus y_0=0 \oplus 1=1$

$x^* \times y^*$=0.110 000 111 1

$[x \times y]_{原}$=1.110 000 111 1

$[x \times y]_{补}$=1.001 111 000 10

x·y= -0. 110 000 111 1

运算过程如下：

21. 用原码加减交替法和补码加减交替法计算x÷y。

(1) x=0.100111，y=0.101011；

(2) x=-0.10101，y=0.11011；

(3) x=0.10100，y= -0.10001；

(4) x=13/32，y= -27/32。

解：

(1) $x^*=[x]_{原}=[x]_{补}=x=0.100\ 111$

$y^*=[y]_{原}=[y]_{补}=y=0.101\ 011$

$[-y^*]_{补}=[-y]_{补}=1.010\ 101$

$q_0=x_0\oplus y_0=0\oplus 0=0$

$x\div y=x^*\div y^*=[x\div y]_{原}=0.111\ 010$

$r^*=0.000\ 010\times 2^{-6}=0.000\ 000\ 000\ 010$

计算过程如下：

(2) x= -0.101 01，y=0.110 11

$[x]_{原}=1.101\ 01$

$x^*=0.101\ 01$

$y^*=[y]_{原}=[y]_{补}=y=0.110\ 11$

$[-y^*]_{补}=[-y]_{补}=1.001\ 01$

$[x]_{补}=1.010\ 11$

$q_0=x_0\oplus y_0=1\oplus 0=1$

$x^*\div y^*=0.110\ 00$

$[x\div y]_{原}=1.110\ 00$

$x\div y=-0.110\ 00$

$r^*=0.110\ 00\times 2^{-5}$

$=0.000\ 001\ 100\ 0$

计算过程如下：

(3) x= 0.101 00，y= -0.100 01

$x^*=[x]_{原}=[x]_{补}=x=0.101\ 00$

$[y]_{原}=1.100\ 01$

$y^*=0.100\ 01$

$[-y^*]_{补}=1.011\ 11$

$[y]_{补}=1.011\ 11$

$[-y]_{补}=0.100\ 01$

$q_0=x_0\oplus y_0=0\oplus 1=1$

$x^*\div y^*=1.001\ 01$ —— 溢出

$[x\div y]_{原}$：无定义

$x\div y=-1.001\ 01$

$r^*=0.010\ 11\times 2^{-5}$

$=0.000\ 000\ 101\ 1$

计算过程如下：

(4) x=13/32=（0.011 01）$_2$

y= -27/32=（-0.110 11）$_2$

$x^*=[x]_{原}=[x]_{补}=x=0.011\ 01$

$[y]_{原}=1.110\ 11$

$y^*=0.110\ 11$

$[-y^*]_{补}=1.001\ 01$

$[y]_{补}=1.001\ 01$

$[-y]_{补}=0.110\ 11$

$q_0=x_0\oplus y_0=0\oplus 1=1$

$x^*\div y^*=0.011\ 11$

$[x\div y]_{原}=1.011\ 11$

$x\div y=(-0.011\ 11)_2=-15/32$

$r^*=0.010\ 11\times 2^{-5}$

$=0.000\ 000\ 101\ 1$

26.按机器补码浮点运算步骤，计算 [x ±y]补.

(1) $x=2^{-011}\times 0.101\ 100$，$y=2^{-010}\times(-0.011\ 100)$；

(2) $x=2^{-011}\times(-0.100\ 010)$，$y=2^{-010}\times(-0.011\ 111)$；

(3) $x=2^{101}\times(-0.100\ 101)$，$y=2^{100}\times(-0.001\ 111)$。

解：先将 x、y 转换成机器数形式：

(1) $x=2^{-011}\times 0.101\ 100$，$y=2^{-010}\times(-0.011\ 100)$

[x] 补=1，101；0.101 100, [y] 补=1，110；1.100 100

[Ex] 补=1,101, [y] 补=1,110, [Mx] 补=0.101 100, [My] 补=1.100 100

1）对阶：

[ΔE] 补=[Ex] 补+[-Ey] 补 = 11,101+ 00,010=11,111 < 0，

应 Ex 向 Ey 对齐，则： [Ex] 补+1=11，101+00，001=11，110 = [Ey] 补

[x] 补=1，110；0.010 110

2）尾数运算：

[Mx] 补+[My] 补= 0.010 110 + 11.100 100=11.111010

[Mx] 补+[-My] 补=0.010 110 + 00.011100= 00.110 010

3）结果规格化：

[x+y] 补=11，110；11.111 010 = 11，011；11.010 000 （尾数左规 3 次，阶码减 3）

[x-y] 补=11，110；00.110 010，已是规格化数。

4）舍入：无

5）溢出：无

则：$x+y=2^{-101}\times(-0.110\ 000)$

$x-y=2^{-010}\times 0.110\ 010$

（2）$x=2^{-011}\times(-0.100010)$，$y=2^{-010}\times(-0.011111)$

[x] 补=1，101；1.011 110, [y] 补=1，110；1.100 001

1） 对阶：过程同 (1)的 1)，则

[x] 补=1，110；1.101 111

2）尾数运算：

[Mx] 补+[My] 补= 11.101111 + 11. 100001 = 11.010000

[Mx] 补+[-My] 补= 11.101111 + 00.011111 = 00.001110

3）结果规格化：

[x+y] 补=11，110；11.010 000,已是规格化数

[x-y] 补=11，110；00.001 110 =11，100；00.111000 （尾数左规 2 次，阶码减 2）

4）舍入：无

5）溢出：无

则：$x+y=2^{-010}\times(-0.110\ 000)$

$x-y=2^{-100}\times 0.111\ 000$

（3）$x=2^{101}\times(-0.100\ 101)$，$y=2^{100}\times(-0.001\ 111)$

[x] 补=0，101；1.011 011, [y] 补=0，100；1.110 001

1）对阶：

[ΔE] 补=00，101+11，100=00，001 >0，应 Ey 向 Ex 对齐，则：

[Ey] 补+1=00，100+00，001=00，101=[Ex] 补

[y] 补=0，101；1.111 000（1）

2）尾数运算：

[Mx] 补+[My] 补= 11.011011+ 11.111000（1） = 11.010011（1）

[Mx] 补+[-My] 补= 11.011011+ 00.000111（1） = 11.100010（1）

2） 结果规格化：

[x+y] 补=00，101；11.010 011（1），已是规格化数

[x-y] 补=00，101；11.100 010（1）=00，100；11.000 101 （尾数左规 1 次，阶码减 1）

4）舍入：

[x+y] 补=00，101；11.010 011（舍）

[x-y] 补 不变

5）溢出：无

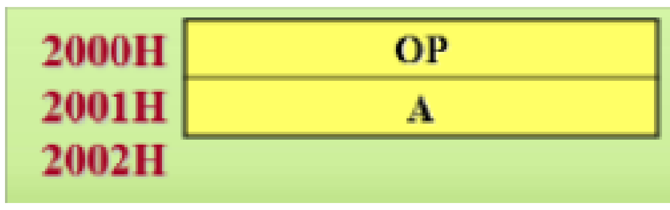
则：$x+y=2^{101}\times(-0.101\ 101)$

$x-y=2^{100}\times(-0.111\ 011)$

第 7 章 指令系统

14. 设相对寻址的转移指令占两个字节，第一个字节是操作码，第二个字节是相对位移量，用补码表示。假设当前转移指令第一字节所在的地址为 **2000H**，且 **CPU** 每取出一个字节便自动完成（ **PC**）+1 PC 的操作。试问当执行 “ **JMP *+8** ”和 “**JMP *- 9** ”指令时，转移指令第二字节的内容各为多少？

解：据题意，相对寻址的转移指令格式如下：



当执行 JMP 指令时，指令第二字节的内容不变，PC 的内容变为 2002H。此时转移指令第二字节内容各为：

A1= +8 = 0000 1000 = 08H

A2= -9 = 1111 0111 = F7H

其有效地址各为：

EA1= (PC) +8 = 2002H+0008H = 200AH

EA2= (PC) -9 =2002H+FFF7H = 1FF9H

16. 某机主存容量为 **4M** ×**16** 位，且存储字长等于指令字长，若该机指令系统可完成 **108** 种操作，操作码位数固定，且具有直接、间接、变址、基址、相对、立即等六种寻址方式，试回答：

（**1**）画出一地址指令格式并指出各字段的作用；

（**2**）该指令直接寻址的最大范围；

（**3**）一次间址和多次间址的寻址范围；

（**4**）立即数的范围（十进制表示）；

（**5**）相对寻址的位移量（十进制表示）；

（**6**）上述六种寻址方式的指令哪一种执行时间最短？哪一种最长？为什么？哪一种便于程序浮动？哪一种最适合处理数组问题？

（**7**）如何修改指令格式，使指令的寻址范围可扩大到 **4M** ？

（**8**）为使一条转移指令能转移到主存的任一位置，可采取什么措施？简要说明之。

解：（1）单字长一地址指令格式：

OP（7 位）	M（3 位）	A（6 位）

OP 为操作码字段，共 7 位，可反映 108 种操作；

M 为寻址方式字段，共 3 位，可反映 6 种寻址操作；

A 为地址码字段，共 16-7-3=6 位。

（2）直接寻址的最大范围为 2^6=64。

（3）由于存储字长为 16 位，故一次间址的寻址范围为 2^{16}；若多次间址，需用存储字的最高位来区别是否继续间接寻址，故寻址范围为 2^{15}。

（4）立即数的范围为 -32—— 31（有符号数），或 0—— 63（无符号数）。

（5）相对寻址的位移量为 -32—— 31。

（6）上述六种寻址方式中，因立即数由指令直接给出，故立即寻址的指令执行时间最短。间接寻址在

指令的执行阶段要多次访存 (一次间接寻址要两次访存，多次间接寻址要多次访存)，故执行时间最长。变址寻址由于变址寄存器的内容由用户给定，而且在程序的执行过程中允许用户修改，而其形式地址始终不变，故变址寻址的指令便于用户编制处理数组问题的程序。相对寻址操作数的有效地址只与当前指令地址相差一定的位移量，与直接寻址相比，更有利于程序浮动。

（7）方案一：为使指令寻址范围可扩大到 4M，需要有效地址 22 位，此时可将单字长一地址指令的格式改为双字长，如下图示：

OP（7 位）	MOD （3 位）	A（高 6 位）
A（低 16 位）		

方案二：如果仍采用单字长指令（ 16 位）格式，为使指令寻址范围扩大到 4M，可通过段寻址方案实现。安排如下：

硬件设段寄存器 DS（16 位），用来存放段地址。在完成指令寻址方式所规定的寻址操作后，得有效地址 EA（6 位），再由硬件自动完成段寻址，最后得 22 位物理地址。 即：物理地址 =（DS）$\times 2^6$ + EA

注：段寻址方式由硬件隐含实现。在编程指定的寻址过程完成、 EA 产生之后由硬件自动完成，对用户是透明的。

方案三：在采用单字长指令（ 16 位）格式时，还可通过页面寻址方案使指令寻址范围扩大到 4M。安排如下：

硬件设页面寄存器 PR（16 位），用来存放页面地址。指令寻址方式中增设页面寻址。当需要使指令寻址范围扩大到 4M 时，编程选择页面寻址方式，则： EA =（PR）‖A （有效地址 =页面地址“拼接”6位形式地址），这样得到 22 位有效地址。

（8）为使一条转移指令能转移到主存的任一位置， 寻址范围须达到 4M，除了采用 (7) 方案一中的双字长一地址指令的格式外，还可配置 22 位的基址寄存器或 22 位的变址寄存器，使 EA = (BR) + A （BR 为 22 位的基址寄存器)或 EA =（IX）+ A(IX 为 22 位的变址寄存器)，便可访问 4M 存储空间。还可以通过 16 位的基址寄存器左移 6 位再和形式地址 A 相加，也可达到同样的效果。

总之，不论采取何种方式，最终得到的实际地址应是 22 位。

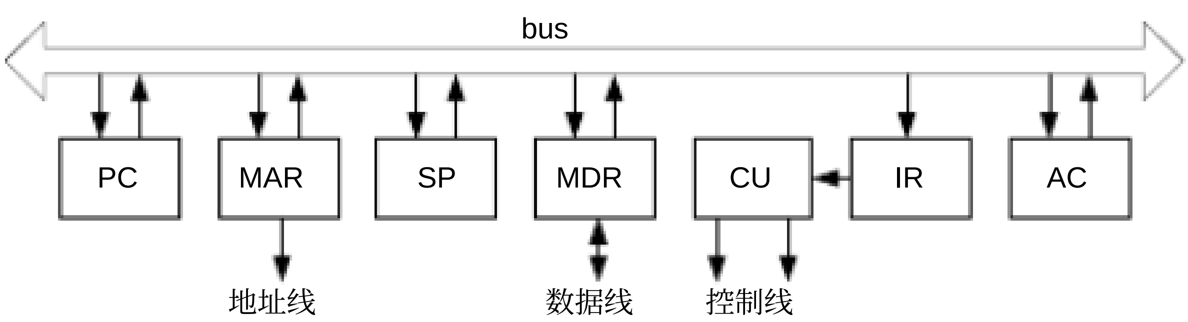
第 8 章 CPU 的结构和功能

4. 设 **CPU** 内有下列部件： **PC**、**IR**、**SP**、**AC**、 **MAR**、**MDR** 和 **CU**。

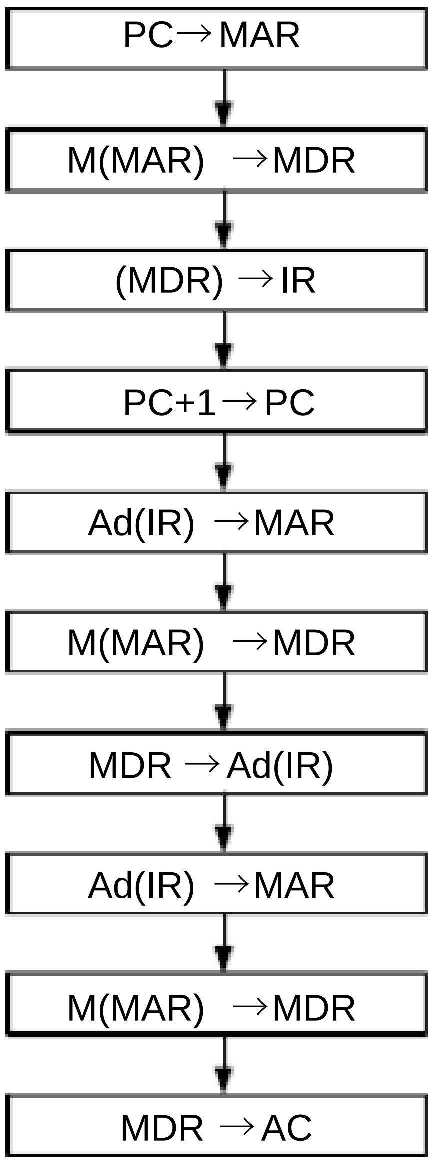
（**1**）画出完成间接寻址的取数指令 **LDA@X** （将主存某地址单元 **X** 的内容取至 **AC** 中）的数据流（从取指令开始）。

（**2**）画出中断周期的数据流。

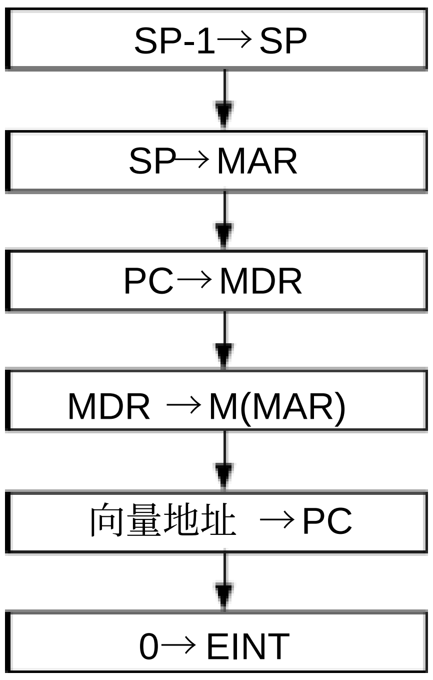
解：CPU 中的数据流向与所采用的数据通路结构直接相关，不同的数据通路中的数据流是不一样的。常用的数据通路结构方式有直接连线、单总线、双总线、三总线等形式，目前大多采用总线结构，直接连线方式仅适用于结构特别简单的机器中。为简单起见，本题采用单总线将题中所给部件连接起来，框图如下：



（1） LDA@X 指令周期数据流程图：



（2）中断周期流程图如下：



25. 某机有五个中断源 **L0**、**L1**、**L2**、**L3**、**L4**，按中断响应的优先次序由高向低排序为 **L0→ L1→ L2→ L3→ L4**，根据下示格式，现要求中断处理次序改为 **L1→ L4→ L2→ L0→ L3**，根据下面的格式，写出各中断源的屏蔽字。

解：各中断源屏蔽状态见下表：

中断源	屏蔽字				
	0	1	2	3	4
I0	1	0	0	1	0
I1	1	1	1	1	1
I2	1	0	1	1	0
I3	0	0	0	1	0
I4	1	0	1	1	1

表中：设屏蔽位 =1，表示屏蔽；屏蔽位 =0，表示中断开放。

第 9 章 控制单元的功能

9.1设 **CPU** 内有这些部件： **PC**、**IR**、**AC**、**MAR**、**MDR** 和 **CU**。

（**1**）写出取值周期的全部微操作。

（**2**）写出减法指令 **SUB X**、取数指令 **LDA X**、存数指令 **STA X(X** 均为主存地址 **)**在执行阶段所需的全部微操作。

答：（1）

PC→MAR 当前指令地址送 MAR，

1→R 启动读操作,

M(MAR)→MDR 当前指令从存储器读至 MDR，

MDR→IR 当前指令送 IR,

OP(IR)→CU 指令的操作码送至 CU 译码,

(PC+1)→PC 形成下一指令地址。

（2）减法指令 SUB X 执行阶段所需全部微操作：

Ad(IR)→MAR 指令的地址码送 MAR，

1→R 启动读操作,

M(MAR)→MDR 操作数从存储器中读至 MDR，

(AC)- MDR→AC 两数相减结果送至 AC。

取数指令 LDA X 执行阶段所需全部微操作：

Ad(IR)→MAR 指令的地址码送 MAR，

1→R 启动读操作,

M(MAR)→MDR 操作数从存储器中读至 MDR，

MDR→AC 操作数送 AC。

存数指令 STA X 执行阶段所需全部微操作：

Ad(IR)→MAR 指令的地址码送 MAR，

1→W 启动写操作,

ACC→MDR 写入的数据送 MDR，

MDR→M(MAR) 数据写入存储器中。

3. 什么是指令周期、机器周期和时钟周期？三者有何关系？

答：CPU 每取出并执行一条指令所需的全部时间叫指令周期；

机器周期是在同步控制的机器中，执行指令周期中一步相对完整的操作（指令步）所需时间，通常安排机器周期长度等于主存周期；

时钟周期是指计算机主时钟的周期时间，它是计算机运行时最基本的时序单位，对应完成一个微操作所需时间，通常时钟周期等于计算机主频的倒数。

6. 设某机主频为 **8MHz**，每个机器周期平均含 **2** 个时钟周期，每条指令平均有 **4** 个机器周期，试问该机的平均指令执行速度为多少 **MIPS**？若机器主频不变，但每个机器周期平均含 **4** 个时钟周期，每条指令平均有 **4** 个机器周期，则该机的平均指令执行速度又是多少 **MIPS**？由此可得出什么结论？

解：先通过主频求出时钟周期，再求出机器周期和平均指令周期，最后通过平均指令周期的倒数求出平均指令执行速度。计算如下：

时钟周期 $=1/8\text{MHz}=0.125\times 10^{-6}\text{s}$

机器周期 $=0.125\times 10^{-6}\text{s}\times 2=0.25\times 10^{-6}\text{s}$

平均指令周期 $=0.25\times 10^{-6}\text{s}\times 4=10^{-6}\text{s}$

平均指令执行速度 $=1/10^{-6}\text{s}=1\text{MIPS}$

当参数改变后：机器周期 $=0.125\times 10^{-6}\text{s}\times 4=0.5\times 10^{-6}\text{s}$

平均指令周期 $=0.5\times 10^{-6}\text{s}\times 4=2\times 10^{-6}\text{s}$

平均指令执行速度 $=1/(2\times 10^{-6}\text{s})=0.5\text{MIPS}$

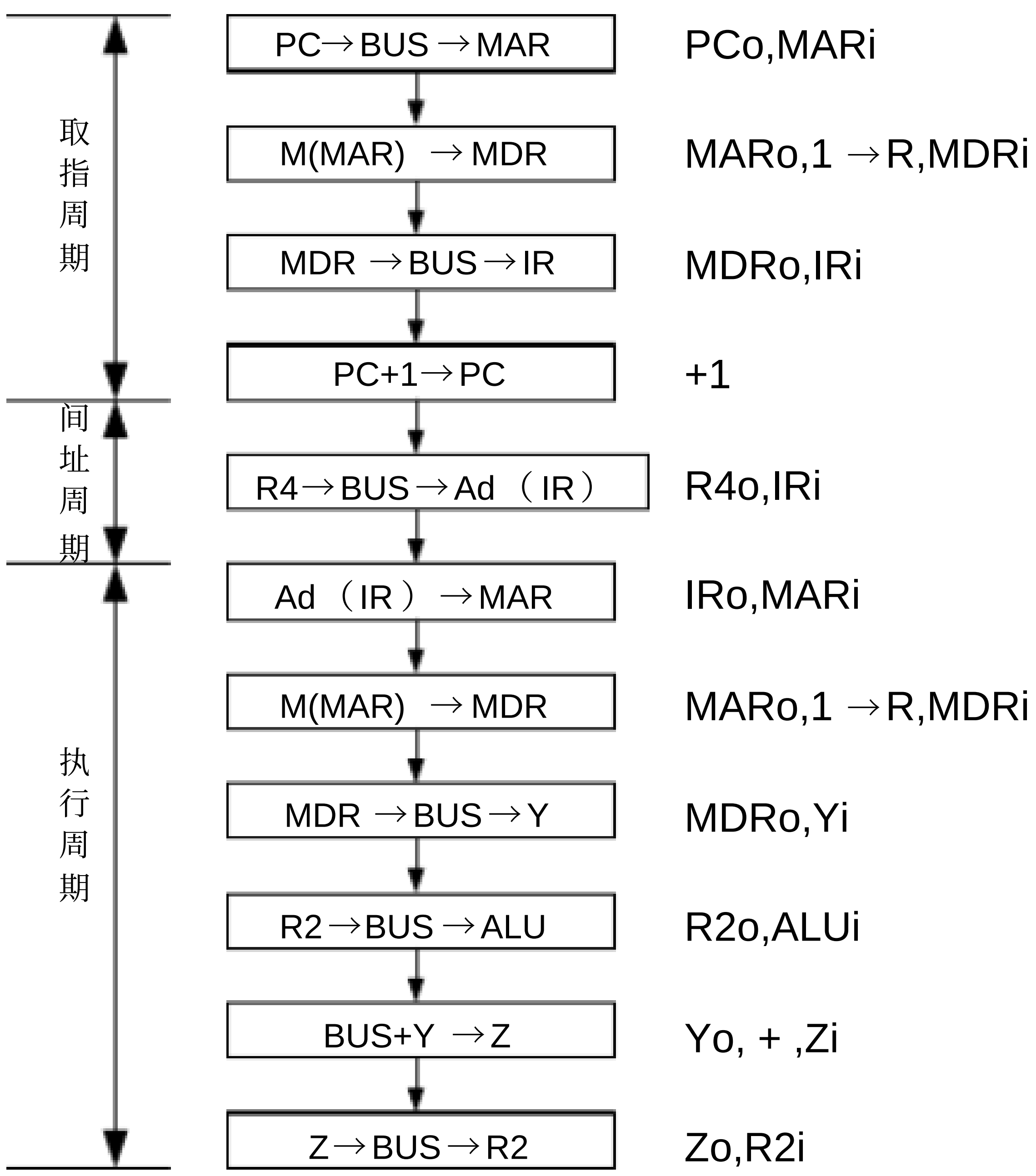
结论：两个主频相同的机器，执行速度不一定一样。

13. 设 **CPU** 内部结构如图 **9.4** 所示，此外还设有 **R1~R4** 四个寄存器，它们各自的输入和输出端都与内部总线相通，并分别受控制信号控制（如 **R2i** 为寄存器 **R2** 的输入控制；**R2o** 为 **R2** 的输出控制）。要求从取指令开始，写出完成下列指令所需的全部微操作和控制信号。

（**1**）**ADD R2，@R4**；**((R2)+((R4)) →R2**，寄存器间接寻址 **)**

（**2**）**SUB R1,@mem**；**((R1)-((mem)) → R1**，存储器间接寻址 **)**

解：（1）ADD R2，@R4 的指令周期信息流程图及微操作控制信号如下：



（2） SUB R1,@mem 指令周期信息流程图及微操作控制信号如下：