**计算机组成原理复习资料**

**第一章知识总结**

· 冯·诺伊曼结构是一种将程序指令存储器和数据存储器合并在一起的存储结构，程序指令存储地址和数据存储地址指向同一个存储器的不同物理位置程序指令和数据宽度相同。

· 冯·诺伊曼结构的特点是：

（1）数字计算机的数制采用二进制

（2）计算机应该按照程序顺序执行。

· 基于冯·诺伊曼结构的计算机由五大部分组成：运算器，控制器，存储器，输入设备，输出设备。

· 今天的大多数计算机是基于冯·诺伊曼结构的。

· CPU由运算器和控制器组成。

· 微处理器的使用标志着微型计算机的发展。

· 计算机进化史：

第一代计算机：1946-1957 真空管（Vacuum Tubes）

第二代计算机：1958-1964 晶体管（Transistors）

第三代计算机：1965-1971 中小规模集成电路(SSI/MSI)，**操作系统**出现

第四代计算机：1972-1977 大规模集成电路出现（LSI）

第五代计算机：1978--- 超大规模集成电路(VLSI)

· 微处理器于1971年出现，并成为第四代微型计算机的核心。

· f指计算机时钟频率，IC指指令数，CPIave指执行指令的平均周期数

· MIPS(Million Instruction per Second),单字长定点指令平均执行速度，MIPS = f(Mhz)/CPIave。

· MFLOPS(Million Floating-point Operations per Second)，每秒百万个浮点数操作，MFLOPS = 浮点操作指令数/（执行时间\*10^6）

· CPU执行时间T: T（Sec） = IC\* CPIave/f(hz)

· 唯有程序运行时间才能反映真实的计算机性能。

第一章测验

1. The basic feature of Von Neumann computer is ( A ).

1．冯诺伊曼体系结构的计算机的基本特征是( A ).

A. access memory by address and execute instruction in sequenc通过地址访存并且按顺序执行指令

B. Multiple Instruction Stream Single Data Stream (MISD) 多指令流单数据流

C. operate stack 操作栈

D. access memory by content 按内容访存

2. A full computer should consists of ( B ).

2.全部的计算机应该由什么组成？( B ).

A. calculator, memory and controller运算器，存储器和控制器

B. hardware and software system硬件和软件系统

C. host and Peripheral主机和外设

D. host and program主机和程序

3. In 8-bits micro-computer system, multiplication and division are realized by ( D ).

3.在一个8位的微型计算机系统中，乘除法依赖于( D ).

A. firmware固件

B. hardware硬件

C. dedicated chips专用芯片

D. software软件

4. The vast majority of computer systems used today are constructed on ( B ) computer model.

4.今天被广泛使用的计算机系统的体系结构是( B ).计算机模型

A. intelligent智能的

B. Von Neumann冯诺伊曼

C. real time processing实时处理

D. parallel并行

5. The reason why the binary system of representation is widely adopted in computer is ( C ).

5.在计算机中二进制表示系统被广泛采纳的原因是( C ).

A. saving components存储组件

B. convenience for information processing方便信息处理

C. the restriction of the nature of physical devices硬件的性质的限制

D. computing speed fast计算速度更快

6. Although computer science and technology have changed tremendously both in hardware and in software, the basic model for computers has remained essentially the same, which was presented by 尽管计算机科学与技术已经极大地改变了不管是硬件还是软件，基础的计算机模型还是从本质上保留了下来，其代表者是( C ).

A. Newton牛顿

B. Einstein爱因斯坦

C. Von Neumann冯诺伊曼

D. Edison爱迪生

7. The operating system is appeared in ( A ).

7.操作系统出现在( A ).

A. the 3rd generation computers第三代计算机

B. the 2nd generation computers

C. the 4th generation computers

D. the 1st generation computers

8. The so called “PC” belongs to ( C ).

8.所谓的“PC”属于( C ).

A. Medium computers中型计算机

B. Mainframes主框架

C. Micro-computers微型计算机

D. Mini-computers迷你计算机

9. Resources management of computer software and hardware is the duty of ( D ).

9.计算机软硬件的资源管理是( D )的职责

A. Database Management System数据库管理系统

B. Application program应用程序

C. Language process program语言处理程序

D. Operating System操作系统

10. The components of CPU do not include ( D ). CPU组件不包括( D ).

A. register寄存器

B. controller控制器

C. Arithmetic unit算术逻辑运算单元

D. memory存储器

11. The computer has experienced 4 generations, which are计算机经历的四代是( D ).

A. Vacuum Tubes, Transistors, SSI/MSI circuit, Laser device

B. Transistors, SMI, Laser device, Optical medium

C. Vacuum Tubes, Digital tube, SSI/MSI circuit, Laser device

D. Vacuum Tubes, Transistors, SSI/MSI circuit, LSI/VLSI circuit

真空管，晶体管，中小规模集成电路，激光部件

晶体管，小规模集成电路，激光部件，光学媒介

真空管，数字管，中小规模集成电路，激光部件

真空管，晶体管，中小规模集成电路，大/超大规模集成电路

12. The use of ( D ) signified the development of micro-computer.

12.( D )的使用标志着微型计算机的发展？

A. software软件

B. disk磁盘

C. OS操作系统

D. Microprocessor微处理器

13.Which of the following languages can be implemented directly and edited by Mnemonic(助记符)( C ): ①Assembly language; ②machine language; ③High-level language; ④Operating system primitives; ⑤Regular language (修正为1、2)

13.以下哪种语言可以被助记符直接实现和编辑( C )？①汇编语言②机器语言③高级语言④操作系统原语⑤常规语言

A. ①, ④

B. ②, ⑤

C. ②, ①

D. ①, ③

14. ( A ) is not belonged to system program. 不属于系统程序

A. Database system数据库系统

B. Operating system操作系统

C. Compiler program编译系统

D. the above all以上都是

15. Data and instructions are stored in ( D ) when the program is running.

15.在程序运行时，数据和指令都存在( D )

A. operating system操作系统中

B. datapath数据路径中

C. disk磁盘中

D. memory存储器中

16. In computer terminology, CPU consists of calculator and controller.( A )

16.在计算机术语中，CPU由运算器和控制器组成. ( A )

17. The use of microprocessor signified the development of micro-computer. ( A )

17.微处理器的使用标志着微型计算机的发展( A )

18. The reason of binary representation for information in a computer is it can easily process the information.( B ) 在计算机中用二进制表示信息的原因是它容易处理信息( B )

原因是元件物理的特性限制。

19. CPU can process information of external memory directly. CPU直接处理存储器外的信息( B )

21. Host consists of CPU and I/O devices. ( B )

21.主机由CPU和I/O设备组成( B ) 还应该有存储器

22. MFLOPS is a performance index for express the speed of processing the floating point number.( A ) 22.MFLOPS 是一个表现标志用以表示浮点数处理速度( A )

23. Software is equivalent to hardware in logic function.( A )

23.在逻辑上软件是可以和硬件等价的( A )

24. In a computer based on the von Neumann model, instructions and data are all stored in memory, and CPU distinguish them according their address. ( B )

24.在一个基于冯诺伊曼的计算机模型上，指令和数据均存在存储器中，并且CPU按地址区分他们( B )

25. Computer hardware consists of calculator, memory, controller and I/O devices. ( A )

25.计算机的硬件由运算器，存储器，控制器和I/O设备组成。( A )

**第二章知识总结**

· 定点数的小数点固定，并且在定点数表示中，小数点均为隐含表示，不占位。

· 定点数分为定点纯整数和定点纯小数。

· 几进制中基数就是几。

· 原码表示法（Sign-magnitude）,符号位上，0表示正，1表示负，有效值用二进制的绝对值表示，此方法与真值最为接近。特点是简单，易于同真值进行转换，实现乘除运算规则简单，但是加减运算麻烦，有“+0”和“-0”之分。

· 补码表示法（2’s complement）,正数的补码是其本身，负数的补码，符号位取1，其余位按位取反，再在末尾加1便可得到，补码的优点是消除了减法。补码中“0”的表示唯一。

· 由[X补]求[-X补]这一过程叫做变补，在减法变加法的过程中使用，变补的做法是将[X补]连同符号位一起按位取反，末位加1。

· 反码（1’s complement）,正数的反码是自身，负数的反码，符号位取1，数值部分按位取反，也有“+0”和“-0”之分。

· 三种表示方法的范围：

定点小数：

原码： -（1-2-n） ≤ N ≤ 1-2-n

反码： -（1-2-n） ≤ N ≤ 1-2-n

补码： -1 ≤ N ≤ 1-2-n

定点整数：

原码： -（2n -1） ≤ N ≤ 2n -1

反码： -（2n -1） ≤ N ≤ 2n -1

补码： - 2n ≤ N ≤ 2n -1

· 定点数运算中，结果超出了计算机能表示的范围后，会发生溢出，基本原因是因为计算机字长的限制。溢出分为两种，一种是正溢出，一种是负溢出；正溢出是指结果超过了计算机所能表示的最大值，负溢出是指结果小于计算机所能表示的最小值。

· 溢出判断方法有三种，这里只介绍常用的两种（1）符号运算进位标志Cf和最高有效位进位标志C进行异或运算，结果为1则发生了溢出，结果为0则结果正确；（2）使用双符号位，首先把参与运算的数改写成双符号位，即把已有的符号位上的数字再多写一遍，如“1.1100”改写为“11.1100”，然后进行预算，符号位结果为“01”时，表明发生了正溢出；符号位结果为“10”时，表示发生了负溢出。符号位结果为“00”或“11”时表示结果正确。

· 定点数二进制运算器中，减法是通过进行补码的加法来实现的。

· 用二进制编码十进制数得到的码叫做BCD码（Binary-Code Decimal）,8421码是其一种，用0000，……，1001表示0-9。使用8421码做加法时，若和大于9则结果需要加6进行修正，小于则不需要修正。

· 计算机中使用无符号整数来表示地址。

**第二章测验**

If [X] 2’s complement = 0.1101010，then [X]sign-magnitude = ( D )

A.0.0010110

B.1.0010110

C.1.0010101

D.0.1101010

观察符号位为0，说明此数为正数，正数的补码表示和源码表示是一样的，因此选D。

2. ( B ) is used to represent address in computer.

1’s complement

Unsigned number

2’s complement

Sign magnitude

计算机中地址使用无符号数表示。

3. Numbers X1, X2 are integer, and 【X1】2’s compl = 10011011，【X2】 2’s compl = 00011011, then their value of decimal form are -101 and 27 .

基本运算，注意观察数字的正负，不可一律按位取反末位加一，正数的补码就是其本身

4. The sign-magnitude representation of ‘0’ is unique. ( B )

源码对“0”的表示并不唯一，有“+0”与“-0”之分。

5. Plus two 2’s complement numbers that adopt 1 sign bit, overflow must occur when ( C/D ).

carry signal is generated from the sign bit

XOR operation for carry signal generated from the sign bit and carry signal generated from the highest numerical bit is ‘0’.

XOR operation for carry signal generated from the sign bit and carry signal generated from the highest numerical bit is ‘1’.

XOR operation for carry signal generated from the sign bit and carry signal generated from the highest numerical bit is ‘1’.

将两个采用单符号位的补码表示的数相加，( C/D )时一定会溢出。

从符号位上产生了进位信号

对从符号位上产生的进位信号和从最高数位上产生的进位信号进行异或操作，结果为0

对从符号位上产生的进位信号和从最高数位上产生的进位信号进行异或操作，结果为1

对从符号位上产生的进位信号和从最高数位上产生的进位信号进行异或操作，结果为1

//C,D答案一样，选哪个都行。

6. The range of representation for a 1’s complement number system of 64 bits (including the sign bit) is ( A ).

0≤|N|≤263 – 1

0≤|N|≤262 – 1

0≤|N|≤264 – 1

0≤|N|≤263

除去符号位后，剩余63位可以用来表示数字，根据反码的表示范围1–2n≤ N ≤2n – 1得出答案

7. Fixed point number can be classified into pure decimal(纯小数) and pure integer(纯整数).( A )

8. In fixed point calculator, whether adopted double sign bit or single sign bit, it must has ( C ), which is often implemented by ( C ).

Decoding circuit, NAND gate

encoding circuit, NOR gate

overflow detection circuit, XOR gate

shift circuit, AND-OR gate

在定点数计算中，是否采取双符号位还是单符号位，它都必须有( C )，它经常使用( C )来实现

解码电路，与非门

译码电路，或非门

溢出检测电路，异或门

移位电路，与或门

一般来说，使用检测符号进位信号和最高数位进位信号的异或结果来进行溢出判断，因此需要异或门。

9. Arithmetic shift 2’s complement of a positive, sign bit remains unchanged, and the blank bit fills in ‘0’. Arithmetic left shift 2’s complement of a negative, sign bit remains unchanged, and the low bit fills 0 . Arithmetic right shift 2’s complement of a negative, sign bit remains unchanged, and the high bit fills 1 and truncate low bit.

对正数的补码进行算术移位，符号位保持不变，空余位填‘0’；对负数的补码进行算术左移，符号位保持不变，低位填‘0’， 对负数的补码进行算术右移，符号位保持不变，高位填‘1’，并且舍弃低位。

10.Let the word length is 8, the fixed point integer with 2’s complement representation of -1 is 11111111 .

“-1”，则最高位为“1”，后7位的真值为“0000001”，按位取反得“1111110”，再加一得到“1111111”，合起来为：“11111111”。注意，这是对于整数，对于定点小数来说，“-1”是“10000000”

11.In fixed point operation, it will be overflow when the result exceeds the represent range of the computer.( A )

在定点数操作中，当结果超出了计算机所能表示的范围时将会发生溢出。显然是对的

12. For a 8-bit 2’s complement representation integer number, its minimal value is -128 , its maximal value is 127 .

对于一个八位的补码表示的整数，最小值是-128，最大值是127。

13.A fixed point number is composed of sign bit and numerical part.( B )

14. The range of representation for a 2’s complement number system of 16 bits (including the sign bit) is ( A ).

-215 ~ + (215 -1)

- (215 –1) ~ + (215 –1)

-215 ~ + 215

- (215 + 1) ~ + 215

对于一个16位(包含符号位)的系统，补码的表示范围为- 215 ~ + (215 –1)

15.8-4-2-1 BCD code of a number is 0111 1000 1001， then its value is 789

16. The addition/subtraction algorithm for sign magnitude representation is rather simple.( B )

原码用于乘除法比较简单，补码用于加减法比较简单。因此错误。

18.The number represented in the computer sometimes will be overflow, the fundamental reason is the limited computer word length.( A )

计算机中的数字表示有时候会溢出，其基本原因是计算机字长限制。

19.For fixed point binary calculator, subtraction is implemented through ( B ).

2’s complement binary subtractor

2’s complement binary adder

sign magnitude decimal adder

sign magnitude binary subtractor

对于定点数二进制运算器，减法通过补码的加法来实现。

20.In 2’s complement addition/subtraction, using 2 sign bits for overflow detection, when the 2 sign bits ‘S1S2’ equals ‘10’, it means that ( C ).

result is positive, with no overflow

result is negative, with no overflow

result is overflow

result is underflow

在补码加减法中，使用双符号位进行溢出检测，当双符号位为“10”时，意味着结果已经溢出，并且是负溢出，当双符号位为“01”时，结果为正溢出。“00”或“11”时，表示结果正确。

21.The 2’s complement representation of -127 is 10000000.( B )

-127的补码为：10000001，10000000为-128的补码。

22. The minimal number of the following numbers is ( B ).

A. （100101）2

B. （100010）BCD

C. （50）8

D. （625）16

换算成10进制，A.37 B.22 C.40 D.1573

23. 2’s complement representation of ‘0’ equals to 1’s complement representation of ‘-1’.( B )

补码对“0”的表示：“00000000”，反码对“-1”的表示：“11111110”

24. If [X] 2’s complement = 1.1101010，then [X]sign-magnitude = ( B )

1.0010101

1.0010110

0.0010110

0.1101010

显然，X是负数，对.1101010减一，得.1101001，按位取反得.0010110，因此得1.0010110

25．For sign magnitude representation, 1’s complement representation, 2’s complement representation, sign magnitude and 1’s complement has 2 representations of ‘0’

“0”的表示在原码和反码中均不唯一，都有“+0”和“-0”之分。

26. The use of 2’s complement operation is adopted to simplify the design of computer.( A )

正确，为了简化加减法的运算。

27. Fixed point calculator is used for ( C ).

fixed point operation

floating point operation

fixed point operation and floating point operation

decimal addition

C正确，浮点数运算中的阶码运算是定点数的加减运算，还是会用到定点数运算器。因此选C

28. When -1<x<0, [x]sign-magnitude = ( A )

1-x

(2-2-n)-|x|

2+x

X

因为x<0,所以1-x = 1+|x|,且-1<x<0，所以|x|表示的是小数部分，数值不变，加的那个1恰好变成了符号位，1刚好代表负数，因此当-1<x<0时，x的原码刚好是1-X。

29. The maximal number of the following numbers is ( A ).

（227）8

（96）16

（10010101）2

（143）5

以上各数换算成十进制后的值为：A.151 B.150 C.149 D.48

30. 8-4-2-1 code is binary number.( B )

8421码确实是十进制数的二进制表示，说到底还是十进制数，。

31. A decimal number is 137.5, then its octal form is 211.4 , its hexadecimal form is 89.8

Octal:八进制；hexadecimal：十六进制

33. The ( C )representation of ‘0’ is unique.

A. sign magnitude and 1’s complement;

B. 1’s complement

C. 2’s complement

D. sign magnitude

只有补码对0的表示是唯一的，原码和反码的表示中，都有“+0”和“-0”之分

34. The range of representation for a unsigned binary number system of 16bits is 0 ~ 65535 .

题中说明是无符号数，因此范围为0~216

35. Given [x1] 2’s complement =11001100， [x2 ]sign magnitude=1.0110, the decimal value of x1 and x2 are -52 and -0.375 .

**第五章知识总结**

· 现今使用中的大多数计算机系统都是在冯·诺依曼计算机模型上构造的。该模型于1946年由冯·诺依曼提出。

· 冯·诺依曼计算机模型中计算机被看作是一个存储程序计算机。

· 一道程序是一个指令序列，其中每一条指令执行一个基本操作。执行前，程序和将要由它加工的数据一起存放到存储器中。

· 在程序执行中，它的指令一条一条地从存储器读出，送到处理单元中去。处理单元译码、取数，执行，并写回结果。

· 冯·诺依曼机型典型组成包含：存储器，CPU（运算器，控制器），I/O

· 算术逻辑单元（ALU）是CPU的心脏。通常ALU有一个二进制加法器，而ALU的性能主要取决于它的加法器

· 半加器只是对位进行运算，不考虑进位，全加器考虑进位。

· 串行级联的4位全加器，又称为行波进位加法器（Ripple-carry adder）,这种加法器因为进位延迟以及门延迟的累加，速度较慢。

· 采用“超前进位产生电路”同时形成各位进位，从而实现快速加法。我们称这种加法器为超前进位加法器。

· 算术逻辑单元（ALU）是一种功能较强的组合逻辑电路。它能进行多种算术运算和逻辑运算。ALU的基本逻辑结构是超前进位加法器。

· 在一个全加器中，第i位的进位产生变量G是Xi·Yi 的结果，即Xi 和Yi 均为1时，才产生进位；第i位的进位传递变量P是Xi＋Yi的结果，即Xi 和Yi 两者中有一个为1时，进位才可以传递。

· 商用芯片74181是一个四位的算术逻辑单元，可以提供16种不同的算术运算和16种不同的逻辑运算，M信号控制运算模式，M=1时，进行逻辑运算；M=0时，进行算术运算。

· 商用芯片74182是一个超前进位产生器，可以用来实现算术逻辑单元的组间并行，来提高速度，达到所有位均并行。74182有4队进位产生信号和进位传递信号引脚。

· 使用1个74182芯片和4个74181芯片可以实现一个全16位并行的算术逻辑单元； 使用5个74182芯片和16个74181芯片可以实现一个全64位并行的算术逻辑单元。

**第五章测验**

1. Calculator has many components, but data bus is the key part.( B )

“运算器中有许多组件，但数据总线是关键部分”，错误，算术逻辑单元才是关键。

2. In an adder, the carry generate variable (G) of bit ‘i’ is ( D ).

A. Xi⊕Yi

B. Xi·Yi·Ci

C. Xi＋Yi＋Ci

D. Xi·Yi

在一个全加器中，第i位的进位产生变量是Xi·Yi 的结果，即Xi 和Yi 均为1时，才产生进位。

3. The carry look-ahead circuit chip 74182 realizes the carry logic between groups in parallel.( A )

超前进位产生电路芯片74182可以实现进位逻辑组间并行。

4. The subtraction algorithm of fixed point binary is realized by ( C ).

A. subtraction for sign magnitude representation

B. addition for binary code decimal

C. addition for 2’s complement representation

D. subtraction for 2’s complement representation

定点二进制数的减法算法依赖于基于补码表示的加法。

5. The main function of ALU is ( D ).

A. arithmetic operation

B. only addition operation

C. logic operation

D. logic and arithmetic operation

ALU的主要功能是逻辑和算术运算，显然么，因为ALU叫做算术逻辑单元么。

6. In a ripple-carry adder, the key factor affecting the speed of the adder is ( D ).

A. Gate-level delay

B. speed of components

C. various speed of each full adder for bit i

D. carry propagation delay

在一个行波进位加法器中，影响加法其速度的关键因素是( D )

门级延迟

组件的速度

全加器对于各位的速度

进位积累延迟

在串行加法器内的一次运算中，进位信号经过的门会越来越多，在每一门的延迟都会被积累下来，因此进位积累的延迟成为了影响行波进位加法器运算速度的关键因素。

7. A calculator consists of many components, but the key component of calculator is ( A ).

A. arithmetic and logic unit

B. data bus

C. accumulate register

D. multi-switch

运算器的关键组件是算数逻辑单元。

8. An arithmetic-logic unit is the heart of the CPU, and it belongs to ( D ).

算术逻辑单元是CPU的心脏，它属于( D )

A. controller控制器

B. register寄存器

C. [sequential](http://dict.cn/sequential) [logical](http://dict.cn/logical) [circuit](http://dict.cn/circuit)顺序逻辑电路

D. [combinational](http://dict.cn/combinational) [logic](http://dict.cn/logic) [circuit](http://dict.cn/circuit)组合逻辑电路

算术逻辑单元是一种组合逻辑电路。

9. The commercial ALU chip 74181 is a 4-bit parallel adder with carry look-ahead circuit.（ A ）

商用ALU芯片74181是一个4位带有超前进位产生电路的并行加法器。应该记住。

10. ALU usually has a ripple-carry adder in order to improve the speed.( B )

“ALU经常使用行波进位加法器是为了提高速度”，显然是错的，行波进位加法器也叫串行加法器，因为进位延迟的问题要比并行加法器慢很多，因此：第一，行波进位加法器不能提高运算速度；第二，因为速度慢，ALU也不经常使用它。说法错误。

11. The commercial 4-bit ALU chip 74181 can only perform 16 different arithmetic operations( B )

“商用的4位ALU芯片74181仅能提供16种不同的算术运算”，错误，此芯片还可以提供16种不同的逻辑运算，是进行逻辑运算还是进行算术运算，是由控制信号M给出的，M=1时，进行逻辑运算，M=0时，进行算术运算。

12. 4-bit Arithmetic Logic Unit 74181 can perform ( D ).

A. 16 possible logic operations

B. 16 different arithmetic operations

C. 4-bit multiplication/division operations

D. 16 different arithmetic operations or 16 possible logic operations

同上题。

13. ALU belongs to ( A ).

A. calculator unit

B. control unit

C. memory

D. register

ALU属于运算器，必须~~

14.Using four 74181ALU chips and one 74182CLA chip can achieve the following carry propagation circuit: ( A ).

A. carry look-ahead of all 16 bits

B. ripple carry inside each 4-bit group and carry look-ahead across different groups

C. ripple-carry circuit

D. carry look-ahead inside each 4-bit group and ripple carry across different groups

使用4个74181ALU芯片和一个74182芯片可以实现下列哪个进位传播电路( A ).

A. 16均超前进位

B. 在每个4位组中串行，在组间超前进位。

C. 串行进位电路

D. 组内并行，组间串行

74181芯片是个4位的超前进位的芯片，因此组内一定是并行的，74182芯片有4对进位产生信号和进位传递信号的引脚，可以进行超前进位预测，因此也是可以实现组间并行的，所以使用4个74181芯片和一个74182芯片可以实现全16位并行。因此选A。

15. In an adder, the carry propagation variable (P) of bit ‘i’ is ( A ).

A. Xi+Yi

B. Xi·Yi·Ci

C. Xi＋Yi＋Ci

D. Xi·Yi

在一个全加器中，第i位的进位传递变量P是Xi＋Yi的结果，即Xi 和Yi 两者中有一个为1时，进位才可以传递。

**第六章知识总结**

· 浮点数的表示由三部分组成，符号位S , 阶码E 和 尾数M组成。在IEEE754标准中：

M(23 bit )

S( 1bit )

E( 8 bit )

32 位浮点数 X=(-1)S ·· (1.M) ·· 2E-127

M(52 bit )

S( 1bit )

E( 11 bit )

64 位浮点数 X=(-1)S · (1.M) ·· 2E-1023

· 浮点数的符号位中1表示负数，0表示正数，阶码E使用移码来表示，尾数M使用原码来表示。

·· 尾数中设置一个缺省的1，即1.M中的1.是隐含表示的，而M可以是任意的指定位二进制数值。

· 移码（biased code）[X]移=2n+X,例如对于一个字长为八位（带符号位）的计算机来说，-128的移码表示为0，-127的移码表示为1，-126的移码表示为2，0的表示为128，127的移码表示为255，以此类推。

· 移码的符号位为0时表示负数，符号位为1时表示正数。

· 浮点数的表示范围：

正上溢

负下溢 正下溢

负上溢

负数

正数

-Nmin

-Nmax

Nmin

Nmax

0

· 对于一个32位二进制数所表示的非零规格化浮点数x,其所能表示的最大正数，最小正数，最小负数，最大负数分别为：

最大正数： X= (2 - 2-23)×2127

最小正数： X= 2-128

最小负数： X= - (2 - 2-23)× 2127

最大负数： X= - 2-128

· 浮点数的加减法大致分为四步（1）检测能否简化操作（2）

比较阶码大小并完成对阶（3）尾数进行加减运算（4）结果规格化（5）舍入处理。

· 阶数不同的浮点数无法相加减，所以浮点数加减之前首先应该完成对阶操作，对阶操作要求阶小的浮点数向阶大的浮点数看齐，因为对阶的过程是阶码减小（增大）的同时尾数向左移（右移），当两个浮点数阶数相差巨大时，可能会导致一个浮点数的尾数因为对阶操作而移没了，也就是因为对阶而丢失掉一个数，所以为了保证精度，只能丢掉小数，这便是为什么阶小的数要向阶大的数看齐。

· 浮点数运算结果规格化分为两种，一种是右规格化，指浮点数运算后的结果发生了溢出（由双符号位判断得知），则可以使用尾数右移，阶码增大的方法来修正溢出；一种是左规格化，对于补码表示尾数的浮点数来说，要求符号位和最高数位不相同，否则就应该左规格化，即阶码减一，尾数左移一位，直至符号位和最高数位不相同。

· 结果规格化的目的是使尾数部分的绝对值尽可能以最大值的形式出现。

· 阶码部分在浮点数运算中只进行加减运算和对比操作。

· 浮点数的表示中，基数（radix）是隐含的

**第六章测验**

1. Exponent unit in floating point calculator can realize addition, subtraction, [multiplication](http://dict.cn/multiplication) and division operations.( B )

“阶码部分在浮点数操作中会进行加减乘除操作”，错误，浮点数相乘除，阶码相加减，不会进行乘除操作。次方的次方这样的运算是通过软件来实现的。

2.In addition/subtraction operation on two floating point numbers, x＝Mx·2Ex and y＝My·2Ey, it requires exponent equalization before arithmetic operation. If Ex>Ey, shift My ; if Ex<Ey, shift Mx ; if Ex=Ey, no shift.

在两个浮点数的加减法操作中，x＝Mx·2Ex 且y＝My·2Ey在对他们进行算术操作之前，需要进行对阶，如果Ex>Ey，对My移位，如果Ex<Ey，对Mx移,如果阶数相等，则不需要移位。

说明：浮点数运算时，可能会做加减法运算，阶数不同无法加减，所以阶数不同首先要对阶，对阶有一个原则，即对小不对大，因为对阶的过程中，尾数需要移位，当阶码大小相差过大时，尾数移位的尾数就会很多，甚至完全移没了，因此可能会丢失一个数，所以一大一小两数相加时，必须使小数向大数看齐，这样即便丢数，丢的也是小数，把损失降到了最低。

3.The mantissa of floating point number uses 2’s complement representation, the binary code of the mantissa before normalization is 1.10101. It needs left normalization, and it should shift 1 bit.

浮点数的尾数使用补码表示，尾数的二进制码在规格化之前是1.10101，它需要左规格化，应该被左移一位。

补码表示尾数的的规格化浮点数，其最高位数位应与符号位相反。这里的1.10101，小数点前面的1是符号位，最高数值位也是1，两者相同，应该左移一位，阶码减一。

4.( B ) representation is used in mantissa of floating point number.

A. biased code or excess-2q code

B. sign magnitude

C. 2’s complement

D. 1’s complement

浮点数中的尾数使用原码表示，因为方便乘除法。

5.In the representation of floating point numbers, （  B ）is implicit(隐含)

A. exponent

B. the radix of the number system to represent the mantissa

C. mantissa

D. sign bit

在浮点数的表示中，表示尾数的数字系统的基数是隐含的(显然基数必须是二么)

6. For a IEEE 754 standard Floating-Point number, its mantissa uses ( C ) representation.

A. biased code or excess-2q code

B. 1’s complement

C. sign magnitude

D. 2’s complement

对于一个IEEE754标准的浮点数，它的尾数使用原码表示。

7. Which of the followings is correct:( D )

A. Exponent unit can realize addition, subtraction, [multiplication](http://dict.cn/multiplication) and division operations.

B. Mantissa unit only realize [multiplication](http://dict.cn/multiplication) and subtraction operations.

C. Exponent unit only realize addition and subtraction for exponent.

D. Floating point calculator can be implemented by exponent and mantissa units.

以下哪个说法是正确的

A. 阶码部分会涉及加减乘除操作

B. 尾数部分只能进行乘法和减法操作

C. 阶码部分只对阶码做加减操作

D. 浮点运算器可以通过阶码和尾数单元来实现

8.The maximal positive number in IEEE754 standard for 32-bits format is （ B ）

A. +（2 – 2-23）×2+255

B. +（2 – 2-23）×2+127

C. +（1 – 2-23）×2+127

D. 2+127 + 227

对于32的形式，IEEE754标准中最大正数为：+（2 – 2-23）×2+127

9. Exponent unit in floating point calculator can realize operations of addition, subtraction and compare.( A )

浮点数运算器中，阶码单元可以实现加减和比较的操作。正确，比较是因为要对阶

10. Which is normalized Floating-Point number, if its mantissa is represented by 2’s complement format?( B )

A. 0.01110

B. 1.00010

C. 0.01010

D. 1.11000

以下哪个规格化的浮点数，它的尾数是由补码表示的？

根据用补码表示规格化浮点数的尾数的条件知，最高数位应与符号位不相同，否则应左规格化，因此选B.

11. In IEEE 754 standard, a floating point number is composed of sign bit s, exponent e, and mantissa m.( A )

在IEEE754标准中，一个浮点数应该由符号位s,阶码e和尾数m表示。正确，必须。

12.The sign bit ‘1’ of a biased code number represents the number for positive , while ‘0’ represents the number for negative .

移码的符号位为1时，表示正数，为0时表示负数。移码的符号位刚好和其它三种码相反。

13. In IEEE754 standard floating point, [mantissa](http://dict.cn/mantissa) is coded as sign-magnitude , exponent is coded as biased code .

IEEE754标准中，浮点数的尾数用原码表示，阶码用移码表示。

14.In IEEE 754 standard, the value of exponent is represented in excess-128 code.( B )

“IEEE754标准种阶码的真值是阶码-128”错，是-127。

15.In a algorithm for [normalized](http://dict.cn/normalized) float-point number, a number is 25×1.10101, with 2’s complement representation for mantissa. Then it (  B ).

在一个对浮点数规格化的算法中，一个数是25×1.10101，若使用补码表示其尾数，则应该

A. needs left shift 2 bits of mantissa for normalized尾数左移两位规格化

B. needs left shift 1 bit of mantissa for normalized. 尾数左移一位规格化

C. needs no normalized不需要规格化

D. needs right normalized需要右规格化

根据用补码表示规格化浮点数的尾数的条件知，最高数位应与符号位不相同，否则应左规格化，即尾数左移一位，阶码减一，直至最高数位应与符号位不相同。

16. The exponent, E, of a floating point number usually uses biased code representation, which is more convenient for comparing size or exponent equalization.( A )

阶码，E,浮点数中经常用移码表示，因为它方便比较和对阶。

17.The mantissa of a Floating-Point number is represented by 2’s complement, then whether the Floating-Point number is normalized is decided by ( ).

A. mantissa’s sign bit and the first bit of mantissa’s numerical part are identical

B. the sign bit of exponent and mantissa are identical

C. mantissa’s sign bit and the first bit of mantissa’s numerical part are different

D. the sign bit of exponent and mantissa are different

浮点数的尾数用补码表示，则浮点数是否规格化取决于( A )。

尾数的符号位和尾数的第一位相同。

阶码和尾数的符号位相同。

尾数的符号位和尾数的第一位不同。

尾数和阶码的符号位不同

B,D肯定不对。这个从汉语的角度来理解，AC选项的意思是一样的，因为若相同，可以决定它需要规格化，若不同，则可以决定它不需要规格化，因此汉语意思是一样的。但老师的本意是问什么时候应该规格化，所以我和老师反映此题最好改为need normalize，这样就明确选A了，也不知道老师会不会改题。

18.In the representation of floating point numbers, 基数(radix) is implicit and invisible to the computer hardware.

在浮点数的表示中，基数在计算机硬件中是隐含的和不可见的。因为计算机使用二进制表示数字么，所以基数当然是2了。因此隐含了。

19.The purpose of using normalized floating point number is ( D ).

A. to expand the range of data representation

B. to avoid for overflow

C. convenient for floating point operation

D. to ensure maximum accuracy of representation

使用规格化浮点数的目的是确保表示的最大精度,防治前导‘0’对数位的浪费，

20.(  A ) representation is used in exponent of Floating-Point number.

A. biased code or excess-2q code

B. 1’s complement

C. sign magnitude

D. 2’s complement

移码用来表示浮点数中的阶码。

**第七章知识总结**

· 指令系统设计是中央处理器设计的基础 , 软件通过指令系统来与硬件打交道， 指令系统是衡量一个计算机表现的关键因素。

· 指令格式对CPU的基本组织产生强有力的影响 : 操作码字段规定CPU实现的操作, 算数逻辑操作直接由ALU执行, 指令的地址字段和寻址方式对CPU的组织有显著影响（寄存器的数目和类型）。

· 一条指令必须包含操作码字段和地址码字段，操作码指明了该指令进行什么操作，不同操作有不同操作码（OPcode）；地址码指明了操作数本身或是操作数所在的地址或是下一条指令所在的地址。

· 根据一条指令中有几个操作数地址，可以将指令划分为零地址指令，一地址指令，二地址指令，三地址指令 。

· 在二地址指令格式中，又可根据操作数（operand）的物理位置分为三类，（1）存储器-存储器（SS）型，两个操作数以及操作结果都放在内存中，访存次数最多，速度最慢（2）寄存器-存储器型（RS）型（3）寄存器-寄存器（RR）型，操作数和结果都放在寄存器内，运算不需访存，速度最快。

· 微机中操作码的长度常常不固定。通常在指令字中用一个固定长度的字段来表示基本的操作码，而对于一部分少地址指令则把它们的操作码扩充到该指令的地址字段，即操作码长度可以改变。这种方法在不增加指令字长度的情况下可表示更多的指令，但增加了译码和分析难度，需更多硬件支持。此种方法叫做扩展操作码技术。

· 寻址方式分为指令寻址和操作数寻址两种。

· 指令寻址方式有两种，一种是顺序寻址，即接下来要执行的指令就是下一条指令，地址由程序计数器（PC）给出(因为PC一般来说总是+1)；另一种是跳跃寻址，是指下一条指令的地址码并不由程序计数器给出，而是由本条指令给出，执行到本条指令后，程序计数器内的内容也进行改变，以便跟踪新指令的地址。

· 程序控制指令的功能是改变程序的执行顺序。

· 操作数寻址大致有以下几种：隐含寻址，立即寻址，直接寻址，间接寻址，寄存器寻址，寄存器间接寻址，相对寻址，基址寻址，变址寻址，堆栈寻址。

· 隐含寻址指指令中不显式的给出操作数的地址，而是在指令中隐含着操作数的地址。如累加器（AC）工作时，指令中只有一个操作数地址，另一个操作数地址被隐含，就是累加器本身，并且运算完的结果也存放在累加器内。

· 立即寻址指指令中的地址码部分放的不是地址，放的就是操作数，因此对此类指令运算时根本不需寻址，所以速度最快，但是由于地址码部分字长有限，因此立即寻址能操作的数的范围也有限，这是它的缺点。

· 直接寻址克服了立即寻址的缺点，地址部分存放的就是操作数的地址，这样便可以使用一个机器字长来存放操作数，使得操作数的范围得以扩大。但这种方式仍然有缺点，因为地址部分的字长有限，因此指令中能访问的地址也有限，对于一些大内存机器，将出现内存无法完全访问的情况。

· 间接寻址又克服了直接寻址的缺点，指令中的地址部分存放的是操作数地址的地址，这样便可以扩大能访问内存的范围。但是因为两次访存，速度很慢，这种寻址方式只在早期的计算机中使用，现在较少使用。

· 寄存器寻址中，操作数存在于寄存器中，指令的地址部分给出的不是内存的地址，而是通用寄存器的编号，因为访问寄存器速度较快，所以此种方式快于直接寻址。

· RISC是精简指令计算机的简写，CISC是复杂指令计算机的简写。RISC由CISC发展而来。

· CISC的形成是因为计算机的不断升级扩充的同时还要兼容旧计算机的指令系统，因此指令系统便日趋复杂。复杂指令系统会增加硬件的复杂性，降低机器运行的速度。

· 经过分析，发现指令的使用频率相差悬殊，80%的指令很少使用（二八定律），并且复杂指令计算机增加了硬件复杂性，降低了机器运行速度，不利于微机向高档机发展。因此提出了精简指令系统（RISC）的概念

· RISC通过简化指令使计算机的结构更加简单合理，从而提高运算速度,它有一下几个特点：

RISC的指令系统中仅选使用频率高的一些简单指令和很有用但不复杂指令，指令条数少。

指令长度固定，指令格式少，寻址方式少。

只有取数/存数指令访问存储器，其余指令都在寄存器中进行，即限制内存访问，提高了速度。

CPU中通用寄存器数量相当多；大部分指令都在一个机器周期内完成。

程序控制上以硬布线逻辑为主，不用或少用微程序控制。

特别重视编译工作，以简单有效的方式支持高级语言，减少程序执行时间

**第七章测验**

1. Indirect addressing mode is designed to facilitate the access of data arrays.( B )

“间接寻址方式是为了促进对数组的访问而设计的”

2. Instruction set is a key factor to represent the performance of a computer.( A )

指令系统是一个对一台计算机表现表示的关键因素，正确。

3. Register-Register (RR) addressing mode is slower than Register-Storage (RS) addressing mode.( B )

寄存器-寄存器寻址方式最快，寄存器-存储器寻址稍慢，存储器-存储器寻址最慢。

4.The function of program control instructions is（ D ）.

A. to perform arithmetic and logic operations

B. to move data between I/O and CPU

C. to move data between memory and CPU

D. to change the program executing order

一个程序控制的指令的功能是：（ D ）.

A.提供算术和逻辑操作

B.在I/O和CPU之间转移数据

C.在存储器和CPU之间转移数据

D.改变一个程序的执行顺序

5. According to storage position of operand, the instruction set usually supports SS addressing mode.( B )“根据操作数的存储位置，指令集通常支持存储器-存储器寻址方式。”两次访存太慢，现在基本不用了。

6. An instruction word consists of Opcode and addresses part.( A )

一个指令字由操作码和地址组成，显然正确，必须

7. Format and function of instruction set only affect the hard structure of a computer.( B )

“指令集的功能和格式仅仅影响计算机的硬件结构。”错误，这还会影响计算机的架构，程序设计等等。

8. In register indirect addressing mode, the operand is in ( C ).

寄存器间接寻址模式中，操作数在

A．PC程序计数器

B．stack栈

C．memory内存

D．general register通用寄存器组

9.The operand is in a register, this addressing mode is called（ A ）.

操作数在一个寄存器中，这种寻址方式叫做：寄存器直接寻址

A．register direct addressing mode

B．direct addressing mode

C．indirect addressing mode

D．register indirect addressing mode

10.The address part in a program control instruction represents the address of next instruction that needs transfer.( A )

程序控制指令中的地址部分代表着下一个需要译码的指令地址，正确。

11. In the instruction addressing modes the fastest way to get the operand is ( D ).

A. register addressing mode

B. direct addressing mode

C. indirect addressing mode

D. immediate addressing mode

在指令寻址模式中能最快获得操作数的寻址方式是：立即寻址。访问寄存器再快也是浮云，因为立即寻址方式中的地址部分村的就是操作数，根本不用寻址，所以最快。

12.In order to implement arithmetic operation between two operands for one-address instruction, one operand is indicated by addresses part of instruction, another operand is specified by ( B ).

为了对一个一地址指令的两个操作数之间实现实现算术操作，一个操作数已经被指令的地址部分给出，那么另一个操作数通过（ B ）获得。

A. immediate addressing mode立即寻址

B. implied addressing mode隐含寻址

C. stack addressing mode栈指针寻址

D. indirect addressing mode间接寻址

题中已经说明是一地址指令，而其中一个操作数已经给出，因此另一个操作数必定是隐含的，使用的是隐含寻址方式获得，如累加器（AC）工作时。

13. By using different addressing mode, the instruction set can ( A ).

通过使用不同的寻址方式，指令集可以（ A ）

A. reduce the instruction length, expand addressing space, improve programming flexibility

减少指令长度，扩展指令空间，提升编程灵活性。

B. realize program store and program control实现程序存储和程序控制。

C. access external storage directly直接访问外存。

D. extend OPcode and decrease the trouble of instruction decoding. 扩展操作码并且减少指令译码问题。

14.For the number of instructions, addressing mode and instruction kinds, RISC is less than CISC.( A )

对于指令的数量，寻址方式以及指令种类，精简指令计算机都要少于复杂指令计算机。

15. There are two instruction addressing modes, one is sequential, and the other is jump. Jump addressing mode can perform ( B ).

有两种寻址方式，一种是顺序寻址，一种是跳转寻址，跳转寻址模式可以提供（ B ）

A. conditional branch of program程序的有条件转移

B. conditional or unconditional branch of program程序的有条件或无条件转移

C. unconditional branch of program程序的无条件转移

D. stack addressing栈指针寻址

16. The purpose of using extending Opcode in instruction format is（ C ）.

在指令格式中使用扩展操作码的目的是（ C ）

A.to keep the length of instructions, while increase the addressing space

当地址空间减小时保持指令长度

B. to increase the length of instructions增加指令长度

C. to keep the length of instructions, while increase the kinds of instruction operate当操作指令种类增加时，保持指令长度

D. to reduce the length of instructions减少指令长度

，最好会扩展操作码的计算题。

17.Which instruction has the maximal execution time（ C ）?

A. Program control instructions

B. RS instructions

C. SS instructions

D. RR instructions

显然是SS型指令，因为两次访存，速度最慢。

18. Let the valid address of operand is given in the address part of instruction, then the instruction adopts ( D ).

A. immediate addressing mode

B. indirect addressing

C. register direct addressing mode

D. direct addressing mode

在指令的地址部分给出操作数的合法地址，然后指令将采取**直接寻址**。定义，

19. OPcode in an instruction gives the character and function of the instruction.( A )

“在一个指令中操作码给出了指令的特点和功能。”这句话实际上是讲操作码指明了指令要做何种操作。

20. Instruction addressing mode is the way that form the address of instruction.( A )

“指令的寻址方式是形成指令地址的方式”正确，可以说是寻址方式的定义吧。

**第八章知识总结**

· 中央处理器可分为控制器和运算器，也可细分为控制器，ALU，寄存器，内部总线，而后三者又统称为数据路径。

· CPU内部至少应该有6类寄存器，它们是：存储器缓冲寄存器（MBR） ,指令寄存器（IR）,程序寄存器（PC）,存储器地址寄存器(MAR),通用寄存器（AR）,状态寄存器（SR/PSW）。

· 指令寄存器（IR）用来保存当前正在执行的一条指令。

· 程序计数器（PC）用来确定下一条要执行的指令的地址，也叫指令计数器,执行指令时，CPU自动更改PC中的内容，由于大多指令都是顺序执行，因此PC常常+1，当然，当遇到转移指令时，PC中的内容将从指令寄存器中的地址字段获得。

· 存储器地址寄存器（MAR）用来保存当前CPU所访问的存储单元的地址，由于要对存储阵列进行译码，所以必须用MAR保存地址信息，知道一次读/写过程完成。

· 通用寄存器（AR）,可以暂时存放一些数据，使CPU不用访存，以加快速度，因为通用寄存器数量较多，因此要给通用寄存器编址。

· 状态寄存器（PSW），保存由算数指令和逻辑指令运算或测试结果建立的各种条件代码，还保存中断和系统工作状态信息，以便CPU和系统能及时了解机器运行状态和程序运行状态。

· 根据设计方法的不同，操作控制器可以分为时序逻辑型和存储逻辑型两种，第一种称为硬布线控制器，也叫组合逻辑控制器，第二种成为微程序控制器。

· 一个指令周期通常由若干个CPU周期组成，CPU周期也称为机器周期，而一个CPU周期又由若干个时钟周期（通常称为节拍脉冲或者T周期，它处理最基本的操作）组成。

· 控制器由程序计数器（PC）,指令寄存器（IR），指令译码器，时序产生器和操作控制器组成。

· 精简指令计算机绝大多数采用超标量和超流水线结构。

**第八章测验**

1. The function of direct branch instruction is to transfer the address code of instruction into ( D ).

A. memory

B. accumulator

C. address register

D. PC程序计数器

“直接分支指令的功能是把指令的地址码转移到程序计数器。”是指下一条的指令地址由程序计数器给出，这样的指令也叫直接分支指令，而间接分支指令也叫跳计算，指下一条执行的指令地址，在寄存器中，而不是由程序计数器给出，类似于指令寻址方式中的跳跃寻址。

2．According to the generation mode of control signal, controller can be divided into 微程序控制 and 硬布线控制

3. In CPU, decoder is used for decode of instruction, addressing mode and address of operand.( A )

“在CPU中，译码器被用于指令译码，寻址方式和操作数寻址。”正确

4. Generally, serial register has the function of shift operation.( A )

“通常，串行寄存器有移位操作的功能”正确，串行寄存器由多个触发器组成，输入经过一个一个时钟周期从第一个触发器一步一步向输出端移动，因此具有移位功能。

5. Controller implementation by hardwire is also called（ C ）.

store logic controller

micro programmed controller

combinational logic controller组合逻辑控制器

calculator

通过硬布线实现的控制器也叫组合逻辑控制器。

6．A CPU at least has 6 kinds of register, which are IR， PC， MAR ，MBR general register and status register.

一个CPU中至少得有6类寄存器：指令寄存器（IR），程序计数器 (PC)，存储器地址寄存器（MAR），存储器缓冲寄存器(MBR) 通用寄存器和状态寄存器。

7．In CPU, the register storing the current instruction being executed is 指令寄存器（IR） ，pointing to the next instruction to be fetched is 程序计数器 (PC) 。

CPU中，存放正在执行的指令的寄存器是 指令寄存器（IR） ，指出下一条应该被取出的指令的寄存器是 程序计数器 (PC) 。

8．An instruction cycle is composed of some T cycles( B ).

“指令周期由若干个T周期组成”T指时钟周期，而一个指令周期由若干个CPU周期（机器周期）组成，而不是时钟周期。

9． Status register store the result of arithmetic operation. ( B )

“状态寄存器存储着算数操作的结果”，错误，说法过于片面。状态寄存器保存由算数指令和逻辑指令或测试结果建立的各种条件代码，除此之外，状态寄存器还保存中断和系统工作状态等信息。所以此说法太过片面。

10．In CPU, the register for pointing the next instruction is MAR. ( B )

“在CPU中指出下一条需要执行的指令的地址的寄存器是MAR（存储器地址寄存器）”，显然说法错误，应该是程序计数器。

11．The bits length of registers in CPU is decided by（ C ）.

memory size

pins of CPU

machine word length

instruction length

“在CPU中，寄存器的位长取决于机器字长”，CPU是多少位的，其内主要寄存器宽度就是多少位的，而CPU位数由机器字长决定。

12. For a n-bit CPU, n means data bus length 。

写机器字长（machine word length）也没错，但最好是说数据总线宽度。

13．In CPU the register pointing to the next instruction to be fetched is ( A ).

PC

IR

MAR

PSW

在CPU中，指出下一条应该被取出的指令的寄存器是程序计数器（PC）

14． Counter can be used not only for counting pulse, but also used for frequency divider(分频) and timer(定时器). ( A )

“计数器不仅可以被用于对脉冲计数，而且可以用于分频和定时器。”正确.

15． The register used to store the current instruction being executed is IR. ( A )

“被用来存放正在执行的指令的寄存器是指令寄存器（IR）”正确，必须！

16 The cycle of CPU frequency is（ A ）.

clock cycle

read/write cycle

instruction cycle

machine Cycle

“CPU频率周期是时钟周期”。 CPU频率，就是CPU的时钟频率，简单说是CPU运算时的工作的频率（1秒内发生的同步脉冲数）的简称。

.17 In CPU, register MAR is used to store the memory address during READ/WRITE operations. Register PSW is used to store the status bits as the result of execution of arithmetic, logic and testing instruction.

在CPU中， 存储地址寄存器（MAR） 在读写操作中被用来存储内存地址，而 状态寄存器(PSW) 被用来存储算数运算，逻辑运算和测试结果的结果位。

18 Intel 80486 is a 32 bits processor, while Pentium is（ A ）bits processor.

64 48 16 32

“Intel 80486是一个32位的处理器，而Pentium是一个（）位的处理器”，白中英的《计算机组成原理》第166页说Pentium是32位的，但这题选64。

19 .A CPU consists of （ D ）.

A. calculator and memory

B. controller, ALU and memory

C. controller

D. controller, ALU, registers and cache

根据咱老师的PPT(第八章第23页) CPU由4个功能部件组成：**ALU、寄存器组、内总线和控制器**。此处D最接近，因此选D

20. CPU does not includes ( B ).

instruction decoder

address decoder

MAR

IR

CPU中不包含地址译码器，地址译码器在存储器中，第十章的内容。

21．If the frequency of a computer is the highest, then its speed is the fastest. ( B )

22． PC (program counter) belongs to ( C ).

I/O

ALU

Controller

Memory

“程序计数器属于控制器”

23 The speed of a computer is related to frequency, and is also related to word length, computer architecture, etc. ( A )

“计算机的速度虽然与频率有关，但它业余计算机字长，计算机架构有关”，正确，计算机速度受到很多因素的限制，不一定频率越快速度就越快，要想想“木桶原理”。

24． In a computer, memory and registers can all store data( A )

“在计算机中，内存和寄存器都可以存储数据”

25 Which of the following statements for RISC is correct: ( C ).

RISC has complex instruction system

RISC is not necessary pipeline CPU

RISC must be pipeline CPU

CPU uses fewer general registers

精简指令计算机不可能有复杂的指令系统，A显然错误；精简指令计算机往往使用大量的寄存器组，因此D也错误，精简指令计算机为了提高处理速度，大多数都使用流水线结构和超标量结构，因此C贴近。

**第九章知识总结**

· CPU由运算器和控制器两大部分组成。

· 控制器由程序计数器（PC），指令寄存器（IR），指令译码器，时序产生器和操作控制器组成。

· 运算器由算术逻辑单元（ALU）,通用寄存器（AR），存储器缓冲寄存器（MBR）和状态寄存器（PSW）组成。

· 操作控制器可以分为时序逻辑型和存储逻辑型两种。第一种称为硬布线控制器，它采用时序逻辑技术来实现；第二种称为微程序控制器，采用存储逻辑来实现。

· 指令周期是指取出一条指令并执行这条指令的时间，指令周期通常分为两个阶段：取指周期和执行周期。一个指令周期通常由若干个CPU周期组成，越复杂的指令，组成它的CPU周期越多，而一个指令周期至少由两个CPU周期组成。

· CPU周期，也叫机器周期，通常用CPU从内存中读取一个指令字的最短时间来规定CPU周期。一个CPU周期可以 完成一个完整的基本操作，如取指，或者执行等。而一个CPU周期通常由若干个时钟周期组成。

· 时钟周期通常称为脉冲节拍或者T周期，是计算机操作的最小时间单位。

· 硬布线控制器通过逻辑电路直接连线而产生。又称组合逻辑控制方式，特点是速度快，但是难以对指令功能做更新和扩展。

· 微程序控制器是用软件的方法在设计操作控制，控制单元向执行单元发出的各种控制命令叫做微命令,执行单元接受微命令后执行的操作叫做微操作，在一个CPU周期内，一组实现一定功能的微命令的组合，叫做微指令，而一条机器指令的功能是用许多微指令组成的序列来实现来的，这个微指令序列就叫做微程序。

· 微操作可分为相容性和相斥性两种，相容性的微操作指在同时或同一个CPU周期内可以并行执行的微操作；相斥性的微操作指在同时或同一个CPU周期内不能并行执行的微操作。

· 实现全部指令系统的微程序，存放在控制存储器中，控存是一种只读存储器，一旦微程序固化，机器运行时只读不写。

· 微指令至少包含两部分信息，操作控制字段和顺序控制字段，，操作控制字段又称微操作码字段，用以产生某一步操作所需的各个微操作控制信号；顺序控制字段又称微地址码字段，用以控制产生下一条要执行的微指令地址。

· 后继微地址的形成方法有两种：计数器方式和多路转移方式。

· 计数器方式借鉴了用PC计数产生机器指令地址的方法，在微程序控制器中设置一个硬件计数器叫微程序计数器µPC，顺序执行微程序时，（µPC）+1→µPC微程序出现转移时，由微指令地址字段中转移部分结合转移条件把新地址送入µPC。

· 一条微指令存在多个转移分支的情况称为多路转移。后继微程序地址可由设计者指定或由设计者指定的测试判别字段控制产生。

**第九章测验**

1．Instruction cycle is the time that CPU fetches an instruction from memory and executes it. ( A )

“指令周期是指CPU从内存中取出并执行它的时间”，正确，指令周期的定义。！

2. In micro programmed controller, control unit send control signals to execute unit, the control signals are called ( B ).

A. micro instructions

B. micro commands

C. micro program

D. micro operations

“在微程序控制中，控制单元发送控制信号到执行单元，这个控制信号叫微命令”，正确，这种控制信号叫做微命令，形成的操作叫做微操作，而在一个CPU周期中，一组实现一定功能的微命令的组合叫做微指令，而一组微指令序列叫做微程序，这三个概念需。

3 .The hardwired controller run low and it is hard to modify and extend. ( B )

硬布线控制器运行速度快但是无法更新和扩展，这是硬布线控制器的特点

4． Micro-program utilizes software method to design the control operations. ( A )

正确，微程序控制确实是在用软件的方法设计控制操作。

5. Machine cycle is defined by（ B ）.

the average time for writing a data word to memory

the minimal time for reading an instruction word from memory

the maximal time for reading a data word from memory

the average time for reading a data word to memory

机器周期也叫CPU周期，是由CPU一次访存的最小读指令时间来作为规定的。

6． Micro-programs are stored in（ D ）.

RAM

IR

main memory

control memory

微指令存放在控存中。控制存储器用来存放实现全部指令系统的的微程序，是一种只读型存储器，一旦微程序固化，机器运行时只读不写。

7．Instruction cycle is（ A ）.

the time for reading and executing an instruction

the time for reading an instruction from memory

the time for executing an instruction

clock cycle

“指令周期是读取并执行这条指令的时间”，定义，。

8．The mirco-commands of a micro-instruction is mutually exclusive, then ( A ).

they cannot appear in the same time

they can appear in the same time

they can replace each other

they are fault-tolerance

“微指令中的微命令若是相斥性的，那么他们不可以同时执行“这里的同时指的是一个CPU周期内

9. CPU cycle is also called clock cycle. A CPU cycle consists of some machine cycles.( B )

CPU周期不叫时钟周期，而叫机器周期，一个CPU周期由若干个时钟周期组成而不是机器周期，因此错误。

10． Instruction cycle is also called CPU cycle. ( B )

指令周期没有别的名字，CPU周期也叫机器周期，和指令周期不是一个概念。

11．The instruction cycle for all the operations is the same. ( B )

“所有操作的指令都相同”错误。复杂的指令指令周期更长。

12. Comparing to micro-program controller, hardwired controller are: ( D )

low in execution, hard for modify and extend of instruction function.

low in execution, easy for modify and extend of instruction function.

fast in execution, easy for modify and extend of instruction function.

fast in execution, hard for modify and extend of instruction function.

比之于微程序控制，硬布线控制器有更高的执行速度，但是却很难扩展和更新指令功能。

13． Mutually exclusive micro-operations are the operations that cannot execute parallel in a CPU cycle. ( A )

“相斥性微操作是不能在一个CPU周期内并行执行的”，正确！这是相斥性微操作的定义。

14． Which unit is responsible for decode ( D ).

calculator

memory

ALU

Controller

译码是控制器的职责，控制器内含指令译码器，因此译码是它的职责。

15． Each machine instruction is interpreted and executed by a microcode consisting of a sequence of microinstructions. ( A )

“每一个机器指令都被解释为由微指令序列组成的微程序并执行。”正确，微程序就是这样解读机器指令的。

16． In micro programmed controller, the relationship between machine instruction and micro instruction is: ( C )

a program constitutes of some machine instructions can be implemented by a micro instruction.

a micro instruction is composed of some machine instruction

each machine instruction is interpreted and executed by micro- program which constitutes of some micro instructions

each machine instruction is executed by one micro instruction

在微程序控制器中，机器指令和微指令之间的关系是：( C )

一个由若干机器指令组成的城区可以由一个微指令来实现。

一个微指令由若干个机器指令组成。

每一个机器指令都会被由若干个微指令组成的微程序解释并执行。

每个机器指令由一个微指令来执行。

17．The basic idea of multiple transfer for fetch the address of the next micro-instruction is（ D ）.

using PC

using a specific field in instruction

using μPC

using judge field(控制字段) of μPC

“取出下一条指令地址的多路转移的基本思想是利用微程序计数器的控制字段”，这是利用微指令的顺序控制字段的“判别测试”和“条件状态”信息来选择多个“候选”微地址中的一个

18. Every instruction cycle needs at least 2 CPU cycles. ( A )

“每个指令周期至少需要两个CPU周期。”正确，因为取出至少一个，执行至少一个，而CPU周期又是能完成这些独立操作的最小时间单位。所以至少两个周期，而一些复杂的指令，还需要更多的CPU周期。

19．The function(s) of control unit is(are) ( C ).

to fetch an instruction from memory

to decode the OPcode of an instruction

to fetch instruction from memory and decode and generate corresponding control signals and execute

to generate sequential signals

控制器的功能是，从内存中取出指令并且对指令译码和产生相应的控制信号并执行。

20. Processer adopts micro programmed controller is called micro processer. ( B )

显然错误，具有中央处理器功能的大规模集成电路器件，被统称为微处理器，而中央处理器的操作控制方式均分为两种，硬布线控制器和微程序控制器，因此是不是微处理器与是否采用微程序控制无关。

**第十章知识总结**

· 如果存储器中的任何存储单元的内容都能被随机存取，且存取时间和存储单元的物理位置无关，这种存储器被称为随机存储器。如果存储器只能按某种顺序来存取，这种存储器被称为顺序存储器。

· 半导体存储器是随机存储器，磁带存储器就是顺序存储器，磁盘存储器是半顺序存储器。

· 有些半导体存储器的存储内容是不变的，即只能读出，因此被称为只读存储器（ROM）；既能读入又能读出的半导体存储器称为随机读写存储器（RAM）。

· 断电后存储信息消失的存储器，称为易失性存储器，也叫挥发性存储器，断电后仍能保存存储信息的存储器叫做非易失性存储器，也叫非挥发性存储器。RAM是挥发性存储器。

· 存储器的存储速度可以由三个指标来衡量：（1）存取时间：即从向存储器发出读操作命令到数据从存储器中读出所经历的时间；（2）存取周期：连续启动两次独立的访问存储器操作所需要的最小时间间隔，又称为访问周期、存取周期、读写周期等。（3）频带宽度：单位时间内能够访问到的数据个数，也叫做存储器的数据传输率。这3个参数中，存储周期是最重要的参数，它能够全面反映存储器的工作速度。

· 主存的速度总落后于CPU的需要，主存的容量总落后于软件的需要。为了解决对存储器要求容量大，速度快，成本低三者之间的矛盾，目前通常采用多级存储器体系结构，即使用高速缓冲存储器、主存储器和外存储器。

· 随机读写存储器RAM（Random Access Memory）按存储元件在运行中能否长时间保存信息来分，有静态存储器（SRAM）和动态存储器（DRAM）两种。

· 二进制代码位是存储器中最小的存储单位，称为存储元，由若干个存储元组成一个存储单元，再由若干个存储单元组成一个存储器。

· 地址译码驱动系统有两种译码方案，：一维译码方案和二位译码方案。二位译码方案的字线分为行译码字线和列译码字线。

· 存储芯片的容量有限，为了满足实际存储器的容量要求，需要对存储器进行扩展，主要方法有：（1）字扩展法（2）位扩展法（3）字位扩展法

· 位扩展法只加大字长，对片选信号没有要求；字扩展法尽在字项扩充，位数不变，由片选信号来区分各片地址。字位扩展法是位扩展法和字扩展法的结合。

· 一个SRAM存储器由存储体、读写电路、地址译码电路和控制电路等组成。SRAM能长久保持信息，不需刷新，工作稳定可靠。但缺点是：功耗大，集成度低。

· DRAM利用电容上的电荷来存储信息，由于漏电使电容上的电荷衰减，需要定期地重新进行存储，这个过程称为刷新。对整个DRAM必须在一定的时间间隔内完成一次全部单元内容的刷新，否则会出现信息错误。从整个DRAM上一次刷新结束到下一次刷新完为止的时间间隔叫刷新周期刷新方式有三种：集中式、分散式、异步式。

· 可编程ROM有PROM，EPROM，和E2PROM三种，PROM是一次性编程，EPROM叫做光擦除可编程只读存储器。E2PROM叫做电擦除可编程只读存储器

· Flash存储器也叫闪速存储器，是高密度非易失性读写存储器。它具有巨大的比特数目的存储容量，在没有电源的情况下，数据也可以长期保存，既有RAM的优点又有ROM的优点。

· cache是一种高速缓冲存储器，是为了解决CPU与主存之间速度不匹配而采用的一项重要技术，可以把它看作是主存的缓冲存储器，由高速的SRAM组成，为了追求高速，包括管理在内的全部功能均由硬件实现，对程序员透明。

· cache的工作原理是基于程序访问的局部性原则。

· 块是Cache与主存之间数据交换的单位，主存与Cache中块的大小相同但数目不等。

· 与主存相比，Cache的容量很小，它保存的内容只是主存的一个子集，为了把主存中的内容放到Cache中，必须采用某种方法把主存地址定位到Cache中，这称作地址映射。

· 主存与cache的地址映射和地址变换有三种方式：（1）全相联映射及其地址变换（2）直接映射及地址变换（3）组相联映射及其地址变换。

**第十章测验**

1. Fast cache memory is designed such that the main memory appears faster to the processor than it actually is. A

“cache被设计成相对于处理器来讲主存能表现的比它实际上更快一些”,听起来有些拗口，但就是这样的，cache的设计目的就是为了提高CPU对主存的访问速度。

2. In a computer system, all the following units can store information：①Primary memory; ②general registers in CPU; ③cache ④magnetic tape ⑤disk. According to access speed, the order by fast to low is ②③①⑤④ . Main memory includes ①③ ; Secondary memory includes ④⑤

在一个计算机系统中，以下所有单元均可以存储信息①主存②CPU中的通用寄存器③cache④磁带⑤硬盘，根据访问速度，从快到慢的顺序是②③①⑤④主存包括：①③，二级存储包括④⑤

3. Commonly the virtual memory is composed of（ ）, which is a two level storage structure.

A. memory-secondary storage

B. cache- secondary storage

C. cache-primary memory

D. general register-primary

通常虚拟存储由主辅存储构成，它是一个二级存储结构。。

5. A RAM is organized as 512×8bit, besides power supply and ground terminal, the minimal pins number of the chip is 19 .

“一个RAM被组织成一个512×8位的芯片，除去电源供应引脚和接地引脚外，至少还应该有 19 个引脚。”是这样数的：8位的芯片至少有8个引脚连接数据总线，而512（B）的容量要求地址总线至少为9根，以使得RAM容量达到512 = 29，除此之外，为了使得此RAM可以被扩展，它还应该有片选信号引脚。为了区分CPU对此RAM的操作是读还是写，此芯片还应该有读写控制信号引脚。所以总共的引脚数至少应该为：8+9+1+1=19（根）。

6. A SRAM chip is organized as 64K×16bit，then its address length is 16 ，its word length is16.

一个SRAM被组织为一个64K×16位的芯片，那么它的地址长度是 16 ，它的字长是16。64K（216）的容量要求它有16根总线，所以它的地址长度为16。

7. Dual-port memory can operate r/w in a fast way. That is because it adopts（ C ）

A. assembly line

B. new type device

C. two separate read/write circuit

D. high speed chip

双端口存储器可以更快的读写操作，这是因为它采用了（C）

A. 流水线

B. 新型硬件

C. 两套相互独立的读写电路

D. 高速芯片

8. In virtual memory, ( D ) is responsible for address mapping.

A. load program

B. complier

C. programmer

D. operating system

“虚拟存储器中，地址匹配是操作系统的责任。”。

A fully associative cache has high hit ratio and low cost. B

“采取全相连映射的cache有着高命中率和低造价。”错误，全相连的映射策略会有较高的命中率，但它的控制电路很复杂，所以造价不会低，也正是因为控制电路复杂的问题，全相连的映射策略只应用于容量较小的cache中。

10．A direct-mapped cache has high hit ratio and low cost. B

“采取直接相连策略的cache有着高命中率和低造价。”因为内存的每个块只能映射到cache中比较固定的几个行中，因此控制逻辑电路简单，造价也低，但是这种相对死板的映射方式有着较低的命中率，因此说法错误

11． In multi-level hierarchical structure of a computer memory system, register is the fastest, disk is the lowest.（答案错了,这里是对的）

(待定)12. Cache is a part of Memory, it can be accessed directly by instruction.

“cache是存储器的一部分，它可以被指令直接访问”

13．Multi-level hierarchical structure for a computer memory system is used to solve the speed bottleneck of memory.

“计算机存储系统应用多级分层结构是为了解决存储速度上的瓶颈”。错误。

14．A DRAM is organized as 512K×8bit, it has 19 address pins, 8 data pins.

“一个DRAM被组织成一个512K×8位的芯片，它应该有 19 根地址引脚， 8根数据引脚。”因为要保证8位的字长，芯片必须有8根数据总线的引脚。而要保证512K(219)的容量，应该有19根地址引脚。

（待定）15．Associative memory is accessed by address, and it is used for block table in cache.

B

“相连存储器是通过地址进行访问的，并且在cache中它被用于块表。”相连存储器是通过内容进行访问的。

16．The purpose of hierarchical structure in a computer memory system is: ( B ).

计算机存储系统中采用多级结构的目的是：（ B ）

A．to reduce the volume of the computer减少计算机的容量

B．to solve the contradictory between capacity, speed and price. 解决容量，速度和价格之间的矛盾。

C．easy to operate易于操作

D．easy to store huge data便于存储海量数据

B正确，因为内存和cache虽然速度快，但是容量小价格高，而磁盘闪存等容量大但是速度慢，所以为了兼顾速度和容量，计算机存储系统采取多级结构。

（待定）17．Using 16K\*1bit memory chips to form 64K\*8bit main memory module. It need expand 4 times in word, expand 8 times in bit.

使用16K\*1位存储芯片来制作一个64K\*8位存储模块。需要进行4次字拓展，8次位拓展。

18．Address mapping functions between main memory and cache use full-associative mapping scheme, direct mapping scheme and set-associative mapping scheme. A

“主存与cache的地址匹配有全相连匹配策略，直接相连匹配策略和组相连匹配策略。”正确！地址映射方式，书中介绍的就这三种。

19．The memory system for a computer is: A

cache, main memory and secondary storage

primary memory

ROM

RAM

“计算机的存储系统是cache，主存和辅助存储”

20．The purpose of virtual memory is: ( ).

A．to expand the capacity of secondary storage

B．to increase speed for access to primary memory

C．to expand the capacity of primary memory

D．to increase speed for access to secondary storage

“使用虚拟存储的目的是扩展主存的容量”一般来说主存的容量相对于用户来说还是比较小的，因此仍然需要扩展，将辅助存储和主存统一编址便产生了虚拟存储，其目的就是为了扩展贮存的容量。

21．CPU could not access directly to :

A．hard disk

B．register

C．primary memory

D．cache

“CPU不能够直接访问硬盘”

22．16 storage chips of 2K\*4 bit can form a 8K \*16bit memory module.

“16个2K\*4位芯片可以制作一个8K\*16位存储模块。”16个芯片每四个分为一组，做位拓展，可以分出4组，一组是2K的容量，一共是8K的容量。

23．SRAM is faster than DRAM, but its Integration is lower.

“SRAM比DRAM快，但是它的整合度要低些”

24．Memory is used to store（ B）.

A．micro-program

B．data and program

C．program

D．data

“主存被用来存储数据和程序。”简直没啥说的。

（待定）25．Let word length of a computer is 32 bit, the capacity of the memory is 64MB. If the memory is addressed by word, then its range of addressing is 0 ~ 16MB .

令一个计算机的字长为32位，他的容量是64MB如果按字存储为内存编址，那么地址范围为0~4294967296（2^32）。

26．Let the word length of a computer is 32 bit, the capacity of the memory is 4MB. If the memory is addressed by half word, then its addressing space is 64K <0-2M> .

(答案错了)

设计算机字长是32位，而内存的容量是4MB，如果内存按半字编址，那么他的内存空间是 64K 。

27．Refresh mode of DRAM are three ways that are centralization, distributed and asynchronous.

·DRAM的刷新方式有三种，分别是：集中刷新，分散刷新和异步刷新。

28．The purpose of setting a cache between CPU and primary memory is: （ ）

A．to expand the capacity of primary memory

B．to expand both of the capacity of primary memory and the number of registers in CPU

C．to expand the number of registers in CPU

D．to balance the speed between CPU and primary memory

“CPU和主存之间设置cache的目的是：为了平衡CPU和主存之间的速度”正确！主存速度要比CPU慢很多，根据木桶原理，要想提高计算机的速度，必须提高主存的速度，所以设计了cache。

29．Set-associative mapping scheme between main memory and cache is high flexibility, high hit ratio and low cost.

“主存和cache之间的组相连映射策略是很灵活的，高命中的和低开销的。”正确，组相连的方式继承了全相连和直接相连映射策略各自的优点，所以也被广泛使用。

30．Associative memory is a memory addressing by:（ C ）

stack

address and stack

content

address

“相连存储是一种按内容编址的存储器。”切记，这是相连存储器的最大特点。

**计算题**

**一：给出: x= 0.1011，y = - 0.0101**

求: [ 1/2 x] 2’s compl，[1/4 x] 2’s compl，[ - x ] 2’s compl，[1/2 y] 2’s compl，[1/4 y] 2’s compl，[ - y ] 2’s compl

[1/2x]补 = 0.01011, [1/4x]补 = 0.001011,[-x]补 = 1.0101,

[1/2y]补 = 1.11011, [1/4y]补 = 1.111011,[-y]补 = 0.0101。

**二：IEEE 754 format of X is (41360000)16, what is its decimal value?**

将十六进制数展开，可得二进制数格式为：

0 100 0001 0 011 0110 0000 0000 0000 0000

指数e=阶码-127=10000010-01111111= 00000011=(3)10

包括隐藏位1的尾数1.M=1.011 0110 0000 0000 0000 0000=1.011011

于是有：X=(-1)s\*1.M\*2e=+(1.011011)2\*23=+(1011.011)2= (11.375)10

**填空：Convert the following decimal numbers to 8-bit two’s complement numbers or indicate that the decimal number would overflow the range.**

a) (-59)10 = ( )2’s compl

b) (128)10 = ( )2’s compl

**Convert the following decimal numbers to 8-bit sign/magnitude numbers or indicate that the decimal number would overflow the range.**

a) -12810 = ( )sign/magnitude

b) 12710 = ( )sign/magnitude

**Convert the following hexadecimal numbers to decimal.**

a) (7C)16 = 124

b) (ED3A)16 = 60730

**三：设一个加法器的进位分别为C4, C3, C2, C1。C0是低位的进位标志，请分别给出C4, C3, C2, C1在串行模式下和进位先行模式下的逻辑表达式**

（1）串行进位方式：

C1 = G1 + P1 C0

C2 = G2 + P2 C1

C3 = G3 + P3 C2

C4 = G4 + P4 C3

其中： G1 = A1 B1 ，P1 = A1⊕B1

G2 = A2 B2 ，P2 = A2⊕B2

G3 = A3 B3 , P3 = A3⊕B3

G4 = A4 B4 , P4 = A4⊕B4

(2) 并行进位方式：

C1 = G1 + P1 C0

C2 = G2 + P2G1 + P2P1C0

C3 = G3 + P3G2 + P3P2G1 + P3 P2 P1 C0

C4 = G4 + P4G3 + P4P3G2 + P4P3P2G1 + P4P3P2P1 C0

其中 G1—G4 ，P1—P4 表达式与串行进位方式相同。

**四：Suppose a computer with a clock frequency of 100 MHz as four types of instructions, and the frequency of usage and the CPI for each of them are given in table.**

假设一个计算机的时钟频率是100 MHz，并且有4种指令，并且每种的指令的使用频率和CPI已在下表给出。

|  |  |  |
| --- | --- | --- |
| Instruction operation | Frequency of usage | Cycles per instruction |
| Arithmetic-logic | 40% | 2 |
| Load/store | 30% | 4 |
| Compare | 8% | 2.5 |
| Branch | 22% | 3 |

(1) Find the MIPS of the computer and the T (CPU time) required to run a program of 107 instructions.

(2) Combining comparing and branch instructions together so that compare instructions can be replaced and removed. Suppose each compare instruction was originally used with one branch instruction, and now each branch instruction is changed to a compare&branch instruction. Also suppose that the new proposal would decrease the clock frequency by 5%, because the new compare&branch instruction needs more time to execute. Find the new CPIave, MIPS, and T.

(1) 计算出这个计算机运行一个具有107条指令的程序的MIPS和周期。

(2) 把比较和分支指令结合在一起，从而去掉比较指令，假设比较指令被用于分支指令，现在每个分支指令都变成了比较和分支指令，也假设新的方案可以减少5%的时钟频率，因为新的比较和分支指令需要更多的时间去执行，计算出CPIave, MIPS, 和 T 。

(1)

CPIave=0.4\*2+0.3\*4+0.08\*2.5+0.22\*3=0.8+1.2+0.2+0.66=2.86

MIPS=f(MHz)/CPIave=100/2.86=35

T(sec)=IC×CPIave/f(Hz)=107\*2.86/(100\*106)=0.286s

(2)

CPIave=(0.4\*2+0.3\*4+0.22\*3)/0.92=2.66/0.92=2.9

MIPS=f(MHz)/CPIave=(100\*95%)/2.9=32.76

T=IC×CPIave/f(Hz)=(0.92\*107 )\*2.9/(0.95\*100\*106)=0.28s

五：给出一个十进制数 20.59375, 请用IEEE754的单精度浮点数的标准形式表示它。

Given a decimal number 20.59375, represent it as a normalized single-precision floating-point number in IEEE 754 standard format

首先分别将整数和分数部分转换成二进制数:20.59375=10100.10011

然后移动小数点,使其在第1,2位之间

10100.10011＝=1.010010011\*24　　e＝4

于是得到：

S＝0， M＝010010011

E＝e+127 = 4+127 = 131 = 1000 0011

二进制表示：

0100 0001 1010 0100 1100 0000 0000 0000

（41A4C000）16

已知cache 命中率 H=0.98，主存比cache 慢四倍，以知主存存取周期为200ns，求cache/主存的效率和平均访问时间。

解： R=Tm/Tc=4；Tc=Tm/4=50ns

E=1/[R+（1-R）H]=1/[4+（1-4）×0.98]=0.94

Ta=Tc/E=Tc×[4-3×0.98]= 50×1.06=53ns。

已知cache / 主存系统效率为85% ，平均访问时间为60ns，cache 比主存快4倍，求主存储器周期是多少？cache命中率是多少？

解：因为：ta = tc / e 所以 ：tc = ta×e = 60×0.85 = 510ns (cache存取周期)

tm = tc×r =510 ×4 = 204ns (主存存取周期)

因为：e = 1 / [r + (1 – r )H]

所以： H = 2.4 / 2.55 = 0.94

CPU执行一段程序时，cache完成存取的次数为3800次，主存完成存取的次数为200次，已知cache存取周期为50ns,主存为250ns,求cache / 主存系统的效率和平均访问时间。

解 ：命中率 H = Ne / （NC + Nm） = 3800 / (3800 + 200) = 0.95

主存慢于cache的倍率 ：r = tm / tc = 250ns / 50ns = 5

访问效率 ：e = 1 / [r + (1 – r)H] = 1 / [5 + (1 – 5)×0.95] = 83.3%

平均访问时间 ：ta = tc / e = 50ns / 0.833 = 60ns

CPU执行一段程序时，cache完成存取的次数为5000次，主存完成存取的次数为200次。已知cache存取周期为40ns，主存存取周期为160ns。求：

(1)．ache 命中率H，

(2)．Cache/主存系统的访问效率e，

(3)．平均访问时间Ta。

解：① 命中率 H = Nc/（Nc+Nm） = 5000/（5000+200）=5000/5200=0.96

② 主存慢于cache的倍率 R = Tm/Tc=160ns/40ns=4

访问效率：ｅ＝　１／［r + (1-r) h］＝1／[4 + (1-4) ×0.96］

＝89.3℅

③　平均访问时间 Ｔａ＝Ｔｃ／ｅ＝40／0.893＝45ns

某计算机系统的内存储器由 cache和主存构成，cache的存取周期为45纳秒，主存的存取周期为200纳秒。已知在一段给定的时间内，CPU共访问内存4500次，其中340次访问主存。问：

(1) cache的命中率是多少？

(2) CPU访问内存的平均时间是多少纳秒？

(3) Cache-主存系统的效率是多少？

解：cache的命中率H===0.92



CPU访存的平均时间Ta=H·Tc+(1-H)Tm=0.92×45+(1-0.92)×200=57.4ns

Cache-主存系统的效率e===0.78=78%



设某流水线计算机有一个指令和数据合一的cache，已知cache的读写时间为10ns，主存的读写时间为100ns，取指的命中率为98%，取数据的命中率为95%，在执行程序时，有1/5的指令需要存取一个操作数。为简化起见，假设指令流水线在任何时候都不阻塞。问设置cache后，与无cache比较，计算机的运算速度可提高多少倍？

解答：Ta = Tc\*h+Tm\*(1-h)

Ta指= 10\*0.98+100\*0.02 = 11.8

Ta数= 10\*0.95+100\*0.05 = 14.5

Ta = 11.8\*1+14.5\*0.2 = 14.7

(100\*6/5)/14.7 = 8

8-1 = 7 所以，提高7倍。

设有一个Cache的容量为2K字，每块16字，在直接映象方式下，求:

(1) 该Cache可容纳多少个块?

(2) 如果主存的容量为256K字,则有多少个块?

(3) 主存的地址格式? Cache的地址格式?

(4) 主存中的第032AB单元映象到Cache中哪一块?

解：(1) Cache可容纳的块数为:2K/16=27=128(块)

(2) 主存的可容纳的块数为: 256K/16=214(块)

(3) 主存地址格式为:

**块内地址(4位)**

**区内块号(7位)**

**区号(7位)**

Cache地址格式为:

**块内地址(4位)**

**区内块号(7位)**

(4) 主存中的032ABH单元:

032ABH=(0000 0011 0010 1010 1011)2

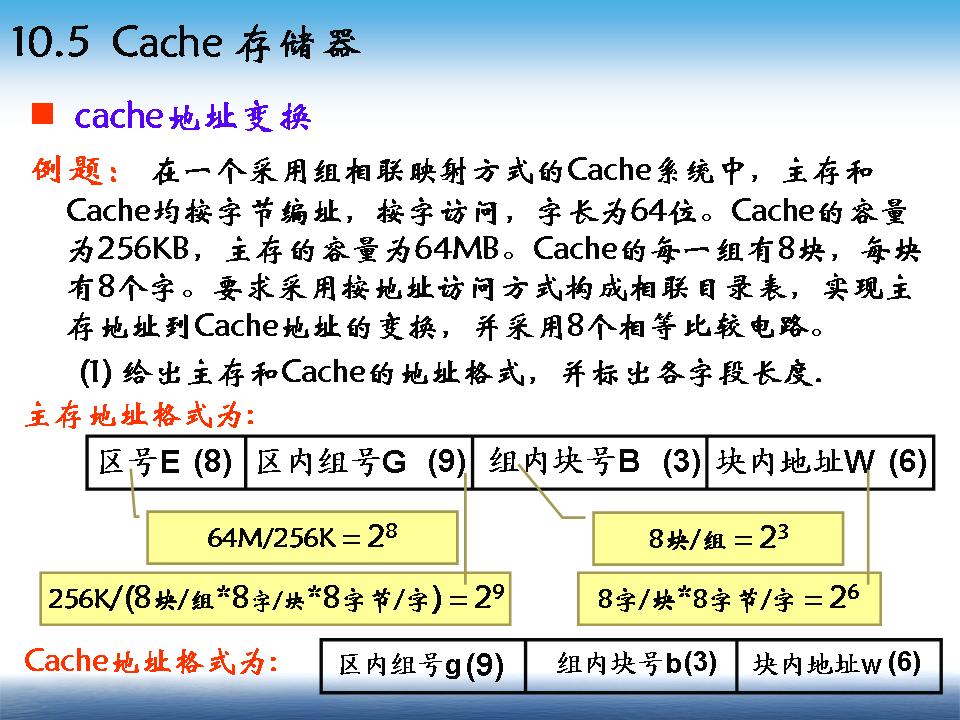
**6区**

**42块**

**11字**

在一个采用组相联映射方式的Cache系统中，主存和Cache均按字节编址，按字访问，字长为64位。Cache的容量为256KB，主存的容量为64MB。Cache的每一组有8块，每块有8个字。要求采用按地址访问方式构成相联目录表，实现主存地址到Cache地址的变换，并采用8个相等比较电路。

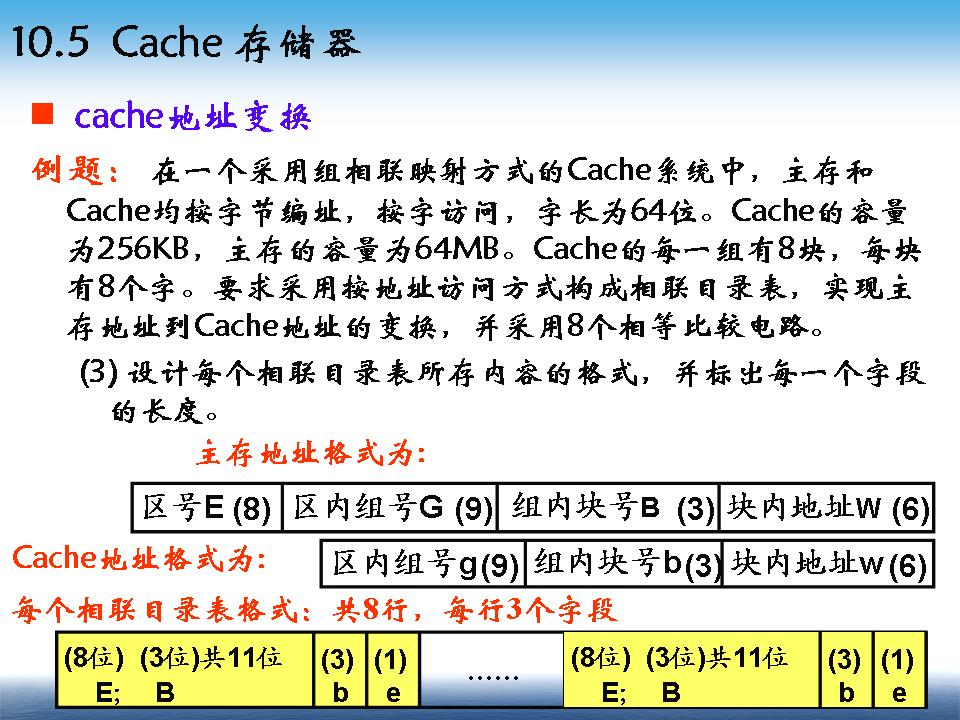
给出主存和Cache的地址格式，并标出各字段长度.



(2) 计算相联目录表的个数。

解：相联目录表的地址个数是29＝512个

设计每个相联目录表所存内容的格式，并标出每一个字段的长度。



(4) 计算每个比较电路的位数。

解：每个比较电路的位数是11位。

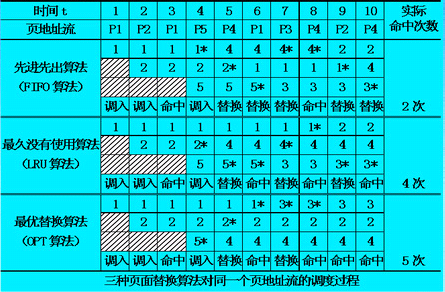
(5) Cache地址的哪些字段可从主存地址直接得到？哪些字段必须从相联目录表得到？

解：Cache地址组号g字段和块内地址w可从主存地址直接得到，组内块号b字段必须从相联目录表得到。

一个程序共有5个页面组成，在程序执行过程中，页面地址流如下，P1、 P2、 P1、 P5、 P5、 P1、 P3、 P4、 P3、 P4，假设在程序执行过程中分配给这个程序的主存储器只有3个页面。

（1）给出用FIFO、LRU、OPT三种页面替换算法对这3个主存的调度情况表，并统计页面命中次数。

（2）计算LRU页面替换算法的页面命中率。



已知某8位机的主存采用半导体存贮器，地址码为18位，若使用4K×4位RAM芯片组成该机所允许的最大主存空间，并选用模块条的形式，问：

若每个模块为32K×8位，共需几个模块条？

每个模块内共有多少片RAM芯片？

主存共需多少RAM芯片？

解：（1）由于主存地址码给定18位，所以最大存储空间为218 = 256K，主存的最大容量为256KB。现每个模块条的存储容量为32KB，所以主存共需256KB / 32KB = 8个模块条。

（2）每个模块条的存储容量为32KB，现使用4K×4位的RAM芯片拼成4K×8位（共8组），用地址码的低12（A0——A11）直接接到芯片地址输入端，然后用地址的高3位（A14——A12）通过3 ：8译码器输出分别接到8组芯片的选片端。共有8×2 = 16个RAM芯片。

（3）据前面所得，共需8个模块条，每个模条上有16片芯片，故主存共需8×16 =128片RAM芯片。

已知某16位机的主存采用半导体存贮器，地址码为18位，若使用8K×8位SRAM芯片组成该机所允许的最大主存空间，并选用模块条结构形式。问：

（1）若每个模块条为32K×16位，共需几个模块条?

（2）每个模块内共有多少片RAM芯片?

（3）主存共需多少RAM芯片？

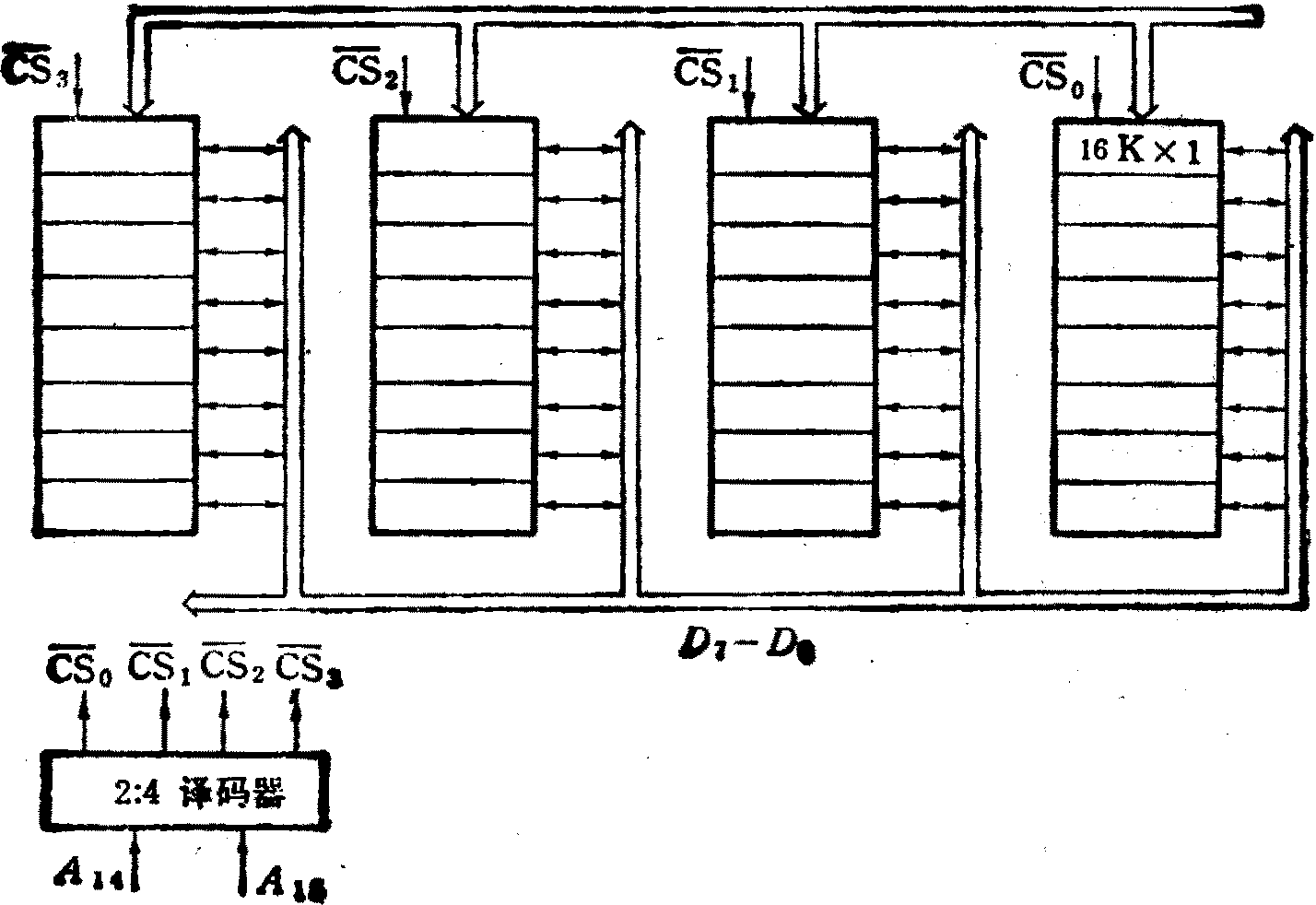
解： （1）由于主存地址码给定18位，所以最大空间为218=256K，主存的最大容量为256K\*16位。现在每个模块条的存贮容量为32K×16位，所以主存共需256K/32K=8块模块条。

（2）每个模块板的存贮容量为32K×16位，现用8K×8位的SRAM 芯片。每块模块条采用位扩展与字扩展相结合的方式：即用2片SRAM芯片拼成8K×16位（共4组），用地址码的低13位（A0 ~ A12）直接接到芯片地址输入端，然后用地址码的高2位（A13 ~ A14）通过 2：4 译码器输出分别接到4组芯片的片选端。共 4×2=8个SRAM

（3）根据前面所得，共虚8个模块条，每个模块条上有8片芯片，故主存共需8×8=64片芯片（SRAM）。

用16K × 1位的DRAM芯片构成64K × 8位的存贮器。要求：

画出该寄存器组成的逻辑框图。

设存贮器读 / 写周期均为0.5μs，CPU在1μs内至少要访存一次。试问采用哪种刷新方式比较合理？两次刷新的最大时间间隔是多少？对全部存贮单元刷新一遍，所需实际刷新时间是多少？

解：（1）根据题意，存贮器总量为64KB，故地址线总需16位。现使用16K×1位的动态RAM芯片，共需32片。芯片本身地址线占14位，2位经过译码形成4个片选逻辑。所以采用位扩展与字扩展结合的方法来组成整个存贮器，其组成逻辑框图如图10-1，其中使用一片2 ：4译码器。

图 10-1

（2）根据已知条件，CPU在1μs内至少需要访存一次，所以整个存贮器的平均读/ 写周期与单个存贮器片的读 / 写周期相差不多，应采用异步刷新比较合理。

对动态MOS存贮器来讲，两次刷新的最大时间间隔是2ms。RAM芯片读/ 写周期为0.5μs，假设16K ×1位的RAM芯片由128 × 128矩阵存贮元构成，刷新时只对128行进行异步方式刷新，则刷新间隔为2m / 128 = 15.6μs，可取刷新信号周期15μs。

**Cs** **ROM**

**RD**

**Cs** **RAM**

**WE**

74LS138

Am A0

Ak Ao

Dn Do

**2K×8位**

**8K×8位**

**32K×8位**

**1K×4位**

**2K×8位**

**8K×8位**

**16K×1位**

**4K×4位**

Dn Do

设CPU共有16根地址线，8根数据线，并用MREQ作访存控制信号（低电平有效），用R/W作读写控制信号（高电平为读，低电平为写），现有下列芯片及各种门电路（自定），如图。画出CPU与存储器的连接图。要求：

（1）存储芯片地址空间分配为：最大4K空间为系统程序区，相邻的4K为系统程序工作区，最小16K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）存储芯片地址空间分配：

最大4K空间为系统程序区;

相邻的4K为系统程序工作区;

最小16K为用户程序区；

**空闲（40K）**

**0**

**216=64k**

**ROM(4K)**

**RAM(4K)**

**RAM(16K)**

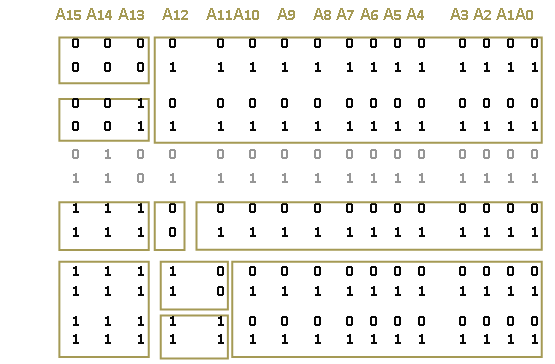
CS0 = Y0

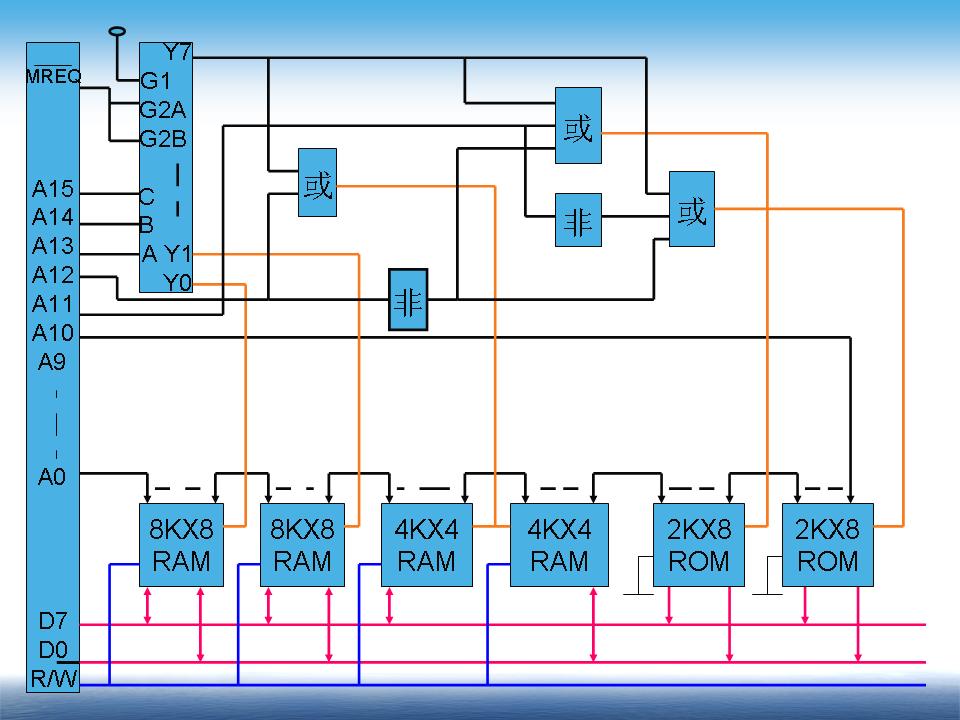
CS1 = Y1

CS2 = Y7+A12

CS3 = Y7+A12+A11

CS4 = Y7+A12+A11





用16M字×8位的存储芯片构成一个64M字×16位的主存储器。要求既能够扩大存储器的容量，又能够缩短存储器的访问周期（提高访问速度）。

(1)计算需要多少个存储器芯片。

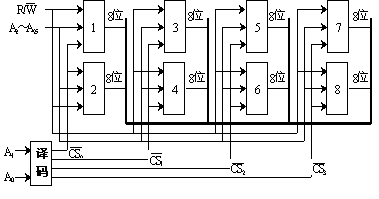
(2)存储器芯片和主存储器的地址长度各需要多少位？

(3)画出用存储器芯片构成主存储器的逻辑示意图。

(4)用16进制表示的地址1234567，其体内地址和体号是多少？

解：(1)计算需要多少个存储器芯片？

8个



**A2～A25**

**A1**

**A0**

解：(2)存储器芯片和主存储器的地址长度各需要多少位？

存储器芯片的地址长度为24位。

主存储器的地址长度为26位

解：(3)画出用存储器芯片构成主存储器的逻辑示意图。

如右图

解：(4) 地址1234567H，其体内地址和体号是多少？

1234567右移两位是48D159，所以其体内地址为：48D159

最低两位是11B，所以其体号为3 。

指令格式结构如下所示，试分析指令格式及寻址方式特点。

15 10 9 5 4 0

|  |  |  |
| --- | --- | --- |
| OP | 目标寄存器 | 源寄存器 |

解：指令格式及寻址方式特点如下：

二地址指令。

操作码OP可指定26=64条指令。

源和目标都是通用寄存器（可分别指定32个寄存器），所以是RR型指令，两个操作数均在寄存器中

这种指令格式常用于算术逻辑类指令。

指令格式结构如下，试分析指令格式及寻址方式特点。

15 10 7 4 3 0

|  |  |  |  |
| --- | --- | --- | --- |
| OP | － | 源寄存器 | 变址寄存器 |
| 位移量（16位） | | | |

解：指令格式与寻址方式特点如下：

二地址指令，用于访问存储器。操作码字段可指定64种操作。

RS型指令，一个操作数在通用寄存器（共16个），另一个操作数在主存中。

有效地址可通过变址寻址求得，即有效地址等于变址寄存器（共16个）内容加上位移量。

指令格式如下所示。OP为操作码字段，试分析指令格式特点。

31 26 22 18 17 16 15 0

OP —— 源寄存器 变址寄存器 偏移量

解：（1）操作码字段为6位，可指定26 = 64种操作，即64条指令。

（2）单字长（32）二地址指令。

（3）一个操作数在原寄存器（共有16个），另一个操作数在存储器中（由变址寄存器内容 + 偏移量 决定），所以是RS型指令。

（4）这种指令结构用于访问存储器。

指令格式如下所示，其中OP 为操作码，试分析指令格式特点。

18 12 10 9 5 4 0

|  |  |  |  |
| --- | --- | --- | --- |
| OP | ——— | 源寄存器 | 目标寄存器 |

解：

单字长二地址指令。

操作码字段OP可以指定27=128条指令。

源寄存器和目标寄存器都是通用寄存器（可分别指定32个），所以是RR型指令，两个操作数均存在寄存器中。

这种指令结构常用于算术逻辑类指令。

指令格式如下所示，OP为操作码字段，试分析指令格式特点。

31 26 22 18 17 16 15 0

OP ———— 源寄存器 变址寄存器 偏移量

解：（1）操作码字段为6位，可指定 26 = 64种操作，即64条指令。

（2）单字长（32）二地址指令。

（3）一个操作数在原寄存器（共16个），另一个操作数在存储器中（由变址寄

存器内容 + 偏移量决定），所以是RS型指令。

（4）这种指令结构用于访问存储器。

有一个字长为32位的浮点数，符号位1位，阶码8位，用移码表示；尾数23位，用补码表示；基数为2。请写出：

（1）最大数的二进制表示；

（2）最小数的二进制表示；

（3）规格化数所能表示的数的范围；

（4）最接近于零的正规格化数与负规格化数。

解：

最大正数值是由尾数的最大正数值与阶码的最大正数值组合而成的;

最小正数值是由尾数的最小正数值与阶码的最小负数值组合而成的。在负数区间;

最大负数值是由尾数的最大负数值与阶码的最小负数值组合而成的;

最小负数值是由尾数的最小负数值与阶码的最大正数值组合而成的。

设浮点数格式为X=2E•S，阶码为8位移码，则阶码的取值范围为 -128~+127；尾数是23位的补码，则尾数最大正数值为Smax=1-2-23；尾数最小正数值为Smin=2-23。尾数最大负值为-2-23；尾数最小负值为-1。

（1）最大数的二进制表示：

正数Xmax=2127•（1-2-23）=1111…11000…00 （23个1，104个0）

负数Xmax=2-128•（-2-23）= - 0.000……0001 （小数点后151个0）

（2）最小数的二进制表示：

正数Xmin=2-128•2-23=0.000……0001 （小数点后151个0）

负数Xmin=2127•（-1）=-10000……000

设有两个浮点数x=2Ex×Sx，y=2Ey×Sy，Ex=(-10)2,Sx=(+0.1001)2, Ey=(+10)2,Sy=(+0.1011)2。 若尾数4位，数符1位，阶码2位，阶符1位，求x+y=？并写出运算步骤及结果。

因为X+Y=2Ex×（Sx+Sy） （Ex=Ey），所以求X+Y要经过对阶、尾数求和及规格化等步骤。

对阶：

△J=Ex－EY=（-10）2－（+10）2=（-100）2 所以Ex<EY，则Sx右移4位，Ex+(100)2=(10)2=EY。SX右移四位后SX=0.00001001，经过舍入后SX=0001，经过对阶、舍入后，X=2（10）2×（0.0001）2

尾数求和： SX+SY

0001（SX）

+ 0. 1011（SY）

SX+SY=0. 1100

结果为规格化数。所以：

X+Y=2（10）2×（SX+SY）=2（10）2（0.1100）2=（11.00）2

设有两个浮点数 N1 = 2j1 × S1 , N2 = 2j2 × S2 ,其中阶码2位，阶符1位，尾数四位，数符一位。设 ：j1 = (-10 )2 ,S1 = ( +0.1001)2 j2 = (+10 )2 ,S2 = ( +0.1011)2

求：N1 ×N2 ，写出运算步骤及结果，积的尾数占4位，要规格化结果。

解（1）浮点乘法规则：

N1 ×N2 =（ 2j1 ×S1）× （2j2 × S2） = 2（j1+j2） ×（S1×S2）

码求和：

j1 + j2 = 0

（3） 尾数相乘：

被乘数S1 =0.1001，令乘数S2 = 0.1011，尾数绝对值相乘得积的绝对值，积的符号位 =

0⊕0 = 0。N1 ×N2 = 20×0.01100011

（4）尾数规格化、舍入（尾数四位）

N1 ×N2 = （+ 0.01100011）2 = （+0.1100）2×2（-01）2

已知X=2010×0.11011011，Y=2100×（-0.10101100），求X+Y。

解：为了便于直观理解，假设两数均以补码表示，阶码采用双符号位，尾数采用单符号位，则它们的浮点表示分别为：

[ X ]浮 = 00010 ， 0.11011011

[ Y ]浮 = 00100 ， 1.01010000

求阶差并对阶：

ΔE = Ex – Ey = [ Ex]补 + [ - Ey]补 = 00010 + 11100 = 11110

即ΔE为 –2，x的阶码小，应使Mx 右移2位，Ex加2，

[ X ]浮 = 00010 ， 0.11011011 （11）

其中（11）表示Mx 右移2位后移出的最低两位数。

尾数和

0 0 1 1 0 1 1 0 （11）

0 1 0 1 0 1 0 0

1 0 0 0 1 0 1 0 （11）

规格化处理

尾数运算结果的符号位与最高数值位为同值，应执行左规处理，结果为1.00010101 （10），阶码为00 011 。

舍入处理

采用0舍1入法处理，则有

0 0 0 1 0 1 0 1

+ 1

0 0 0 1 0 1 1 0

判溢出

阶码符号位为00 ，不溢出，故得最终结果为

x + y = 2011× (-0.11101010)

设[X]补=a0.a1a2···a6，其中ai取0或1，若要x＞–0.5，求a0，a1，a2，···，a6的取值。

解答：

[–0.5 ]原=1.1000000

[–0.5 ]补=1.1000000

[–0.5 ]移=0.1000000

所以，对于负数，即a0 = 1，则a1 = 1 ，且a2～ a6 任意一个为1即可。

对于正数，则a0 = 0，其他任意，就可满足条件。

若浮点数X的IEEE754标准存储格式为(41360000)16求其浮点数十进制数值。

解：将十六进制数展开，可得二进制数格式为：

0 100 0001 0 011 0110 0000 0000 0000 0000

指数e=阶码－127=10000010－01111111= 00000011 =（3）10

包括隐藏位1的尾数1.M = 1.011 0110 0000 0000 0000 0000 = 1.011011

于是有：X = (-1)s \* 1.M \* 2e = +(1.011011)2 \* 23 = + (1011.011)2 = (11.375)10

将数（20.59375）10转换成754标准的32位浮点数的二进制存储格式。

首先分别将整数和分数部分转换成二进制数： 20.59375 = 10100.10011

然后移动小数点，使其在第1，2位之间

10100.10011＝=1.010010011\*24　　　　　e＝4

于是得到：

S＝0， M＝010010011

E＝e+127 = 4+127 = 131 = 1000 0011

二进制表示：

0100 0001 1010 0100 1100 0000 0000 0000 （41A4C000）16

将下列十进制数表示成表示成IEEE754标准的32位浮点规格化数。

（1）27/64 （2）-27/64

解答：（1）27/64 =11011X2-6=1.1011X2-2

符号位：S=0；

阶码值：E=－2＋127＝125=01111101B；

尾数： M=1011 0000 0000 0000 0000 000。

浮点数：0011 1110 1101 1000 0000 0000 0000 0000＝3ED80000H

（2）- 27/64 =-11011×2-6=-1.1011×2-2

符号位：S=1；

阶码值：E=－2＋127＝125=01111101B；

尾数： M=1011 0000 0000 0000 0000 000。

浮点数：1011 1110 1101 1000 0000 0000 0000 0000＝BED80000H

将十进制数-0.75表示成单精度的IEEE754标准代码。

解答：- 0.75= - 0.11B＝-0.11 X 20=-1.1 X 2-1；

符号位：S=1；

阶码值：E=－1＋127＝126=01111110 B；

尾数：M=1000 0000 0000 0000 0000 000。

按浮点数编码格式表示为：1 01111110 1000 0000 0000 0000 0000 000＝BF400000H

将IEEE754单精度浮点数0C0B00000H用十进制数表示：

解答：将十六进制数展开，可得二进制数格式为：

1 10000001 0100 0000 0000 0000 0000 000

符号位S=1；阶码部分值：e＝E－127=129－127＝2；

尾数部分：1.M=1.01＝1.25；

根据IEEE754标准的表示公式，

其数值为——（-1）1 ×（1.25）× 22 = -1 ×1.25 × 4=-5.0

**设计题**

1. CPU has 16 address bus lines (A15-A0), 8 data bus lines (D7-D0), R/W (high level represents Read, while low level represents Write), MREQ control line for accessing memory (low level represents accessible).

Memory space allocation: The minimal 8K are used for system program, which is composed of Read Only Memory chip; the following 24K are used for user program; the last 2K are used for system working.

Now we have: EPROM 8K \* 8 (contains CS control line only);

SRAM 16K\*1, 2K\*8, 4K\*8, 8K\*8;

Decoder 74LS138;

and other logic gates

Questions:

(1) Select appropriate chips to form the required memory space. Which chips are needed? How many chips are needed? Descript the corresponding data bus length, address bus length and control bus line.

(2) Descript the address distribution of memory.

(3) Descript select chip logic functions (片选逻辑函数) of each chip.

(4) Descript the connection way among CPU, memory chips and 74LS138.

1. 一个CPU有16个地址总线(A15-A0)，8根数据总线(D7-D0)，读写控制线(高电平读，低电平写)，访存使能线(低电平可访存)。

内存分配：开始的8K被用于系统程序，由只读存储器芯片组成；接下来的24K被用于用户编程；最后的2K被用于系统工作。

现在有：EPROM 8K \* 8(只含有片选信号);SRAM 16K\*1, 2K\*8, 4K\*8, 8K\*8; 译码器 74LS138；和其他逻辑门电路；

问题:

(1) 选择适当的芯片组成要求的存储空间。需要哪些芯片? 需要多少芯片?说明相应的数据总线宽度, 地址总线宽度和控制总线。

(2) 说明存储器的地址分布。

(3) 说明每个芯片的片选逻辑函数。

(4) 描述CPU,存储芯片和74LS138之间的连接方式。

(1) 需要EPROM 8K \* 8一片，SRAM 8K\*8 3片，2K\*8 1片。译码器 74LS138一片。数据总线宽度为8，地址总线宽度为16，控制总线宽度为2。

(2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 内存区域 | 地址 | | | |
| EPROM起始 | 0000 | 0000 | 0000 | 0000 |
| EPROM结束 | 0001 | 1111 | 1111 | 1111 |
| SRAM用户起始 | 0010 | 0000 | 0000 | 0000 |
| SRAM用户结束 | 0111 | 1111 | 1111 | 1111 |
| SRAM系统起始 | 1111 | 1000 | 0000 | 0000 |
| SRAM系统结束 | 1111 | 1111 | 1111 | 1111 |

(3)因为各个芯片的片选信号来源于74LS138译码器的输出端，因此以输出端的值作为变量，各个芯片的片选逻辑函数如下：

CS(EPROM) = ;



CS(SRAM\_U1) = ;



CS(SRAM\_U2) = ;



CS(SRAM\_U3) = ;



CS(SRAM\_OS) = Y7的反与A与B后的结果再取反。

(4) A12-A0连接到每个芯片的地址线引脚上。CPU的读写端也相应连到各个芯片的读写引脚上，CPU的A15-A13地址线连到74LS138译码器的A，B，C三个输入端上。74LS138译码器的Y0输出端连至EPROM芯片的片选信号引脚上，Y1 ，Y2 ，Y3三个输出端分别连至三个8K\*8芯片的片选信号上，Y4 ，Y5 ，Y6空着不连，Y7输出端先连接一个非门后再与CPU的地址线A11，A12相与，得到的结果取反后再与那片2K\*8的SRAM相连。

2．We use 16M\*8bit memory chip to form a 64M\*16bit main memory module. Required that the capacity of storage be expand, the access time be reduced.

Questions:

(1) How many 16M\*8bit memory chips should be used?

(2) Give the address length of each memory chip and address length of main memory module.

(3) Descript select chip logic functions (片选逻辑函数) of each chip.

Descript the connection way among encoder, CPU and memory chips.

(4)For an address (2345678)16, give its body number and address inside the body.

2．我们使用16M\*8位的存储器芯片去做一个64M\*16位的主存模块，要求对存储容量扩展，访问时间减少。

问题：

应该用多少16M\*8位的芯片？

给出每个存储芯片的地址长度和主存的地址长度。

说明每个芯片的片选逻辑函数，并描述译码器，CPU和存储芯片之间的连接方式。

对于一个地址（2345678）16给出它的body number和body内的地址。

答：

应该用8片16M\*8位的芯片，每2个分为一组，组内做位拓展，把字长拓展到16位，组建做字拓展，把容量拓展到64M。

存储芯片的地址长度是24位；而主存的地址长度是26位。

这里要使用一个24译码器。CPU的地址线A0-A23分别连到每个存储芯片的地址线引脚上，因为芯片两个一组，所以组内芯片1的数据线引脚D7-D0连接到CPU的数据线D7-D0上，而组内芯片2的数据线引脚D7-D0连接到CPU的数据线D15-D8上。CPU的地址线A24-A25连至24译码器的输入端A，B，四个输出端Y0 ，Y1 ，Y2，Y3分别连在这四组的芯片的片选信号引脚上。CPU的读写控制连接到每个芯片的读写控制。

不知道啥意思。

3．CPU has 16 address bus lines (A15-A0), 8 data bus lines (D7-D0), R/W (high level represents Read, while low level represents Write), MREQ control line for accessing memory (low level represents accessible).

Memory space allocation: The minimal 4K are used for system program, which is composed of Read Only Memory chip; the following 4K are used for user program; the last 16K are used for system working.

Questions:

(1) As shown in figures, select appropriate chips to form the required memory space. Which chips are needed? How many chips are needed? Descript the corresponding data bus length, address bus length and control bus line.

(2) Descript the address distribution of memory.

(3) Descript select chip logic functions (片选逻辑函数) of each chip.

(4) Descript the connection way among 74LS138, CPU and memory chips.

3. CPU有16个地址总线(A15-A0)，8根数据总线(D7-D0)，读写控制线(高电平读，低电平写)，访存使能线(低电平可访存)。

内存分配：开始的4K被用于系统程序，由只读存储器芯片组成；接下来的4K被用于用户编程；最后的16K被用于系统工作。

问题：

在给出的数据中，选择适当芯片去组成所要求存储空间，需要哪些芯片？需要多少芯片，说明向相应的数据总线宽度，地址总线宽度和控制总线宽度。

说明内存的地址分布。

说明每个芯片的片选逻辑函数。

说明译码器74LS138，CPU和存储芯片之间的连接方式。

3. 存储器的层次结构主要体现在什么地方？为什么要分这些层次？计算机如何管理这些层次？

答：存储器的层次结构主要体现在Cache-主存和主存-辅存这两个存储层次上。

Cache-主存层次在存储系统中主要对CPU访存起加速作用，即从整体运行的效果分析，CPU访存速度加快，接近于Cache的速度，而寻址空间和位价却接近于主存。

主存-辅存层次在存储系统中主要起扩容作用，即从程序员的角度看，他所使用的存储器其容量和位价接近于辅存，而速度接近于主存。

综合上述两个存储层次的作用，从整个存储系统来看，就达到了速度快、容量大、位价低的优化效果。

主存与CACHE之间的信息调度功能全部由硬件自动完成。而主存与辅存层次的调度目前广泛采用虚拟存储技术实现，即将主存与辅存的一部分通过软硬结合的技术组成虚拟存储器，程序员可使用这个比主存实际空间（物理地址空间）大得多的虚拟地址空间（逻辑地址空间）编程，当程序运行时，再由软、硬件自动配合完成虚拟地址空间与主存实际物理空间的转换。因此，这两个层次上的调度或转换操作对于程序员来说都是透明的。

4. 说明存取周期和存取时间的区别。

解：存取周期和存取时间的主要区别是：存取时间仅为完成一次操作的时间，而存取周期不仅包含操作时间，还包含操作后线路的恢复时间。即：

存取周期 = 存取时间 + 恢复时间

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=10-9s）

6. 某机字长为32位，其存储容量是64KB，按字编址它的寻址范围是多少？若主存以字节编址，试画出主存字地址和字节地址的分配情况。

解：存储容量是64KB时，按字节编址的寻址范围就是64K，如按字编址，其寻址范围为：

64K / （32/8）= 16K

主存字地址和字节地址的分配情况：（略）。

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

8. 试比较静态RAM和动态RAM。

答：略。（参看课件）

9. 什么叫刷新？为什么要刷新？说明刷新有几种方法。

解：刷新：对DRAM定期进行的全部重写过程；

刷新原因：因电容泄漏而引起的DRAM所存信息的衰减需要及时补充，因此安排了定期刷新操作；

常用的刷新方法有三种：集中式、分散式、异步式。

集中式：在最大刷新间隔时间内，集中安排一段时间进行刷新，存在CPU访存死时间。

分散式：在每个读/写周期之后插入一个刷新周期，无CPU访存死时间。

异步式：是集中式和分散式的折衷。

10. 半导体存储器芯片的译码驱动方式有几种？

解：半导体存储器芯片的译码驱动方式有两种：线选法和重合法。

线选法：地址译码信号只选中同一个字的所有位，结构简单，费器材；

重合法：地址分行、列两部分译码，行、列译码线的交叉点即为所选单元。这种方法通过行、列译码信号的重合来选址，也称矩阵译码。可大大节省器材用量，是最常用的译码驱动方式。

11. 一个8K×8位的动态RAM芯片，其内部结构排列成256×256形式，存取周期为0.1μs。试问采用集中刷新、分散刷新和异步刷新三种方式的刷新间隔各为多少？

解：采用分散刷新方式刷新间隔为:2ms，其中刷新死时间为：256×0.1μs=25.6μs

采用分散刷新方式刷新间隔为：256×（0.1μs+×0.1μs）=51.2μs

采用异步刷新方式刷新间隔为:2ms

12. 画出用1024×4位的存储芯片组成一个容量为64K×8位的存储器逻辑框图。要求将64K分成4个页面，每个页面分16组，指出共需多少片存储芯片。

解：设采用SRAM芯片，则：

总片数 = （64K×8位） / （1024×4位）= 64×2 = 128片

题意分析：本题设计的存储器结构上分为总体、页面、组三级，因此画图时也应分三级画。首先应确定各级的容量：

页面容量 = 总容量 / 页面数 = 64K×8 / 4  = 16K×8位，4片16K×8字串联成64K×8位

组容量 = 页面容量 / 组数   = 16K×8位 / 16 = 1K×8位，16片1K×8位字串联成16K×8位

组内片数 = 组容量 / 片容量 = 1K×8位 / 1K×4位 = 2片，两片1K×4位芯片位并联成1K×8位

存储器逻辑框图：（略）。

13. 设有一个64K×8位的RAM芯片，试问该芯片共有多少个基本单元电路（简称存储基元）？欲设计一种具有上述同样多存储基元的芯片，要求对芯片字长的选择应满足地址线和数据线的总和为最小，试确定这种芯片的地址线和数据线，并说明有几种解答。

解：存储基元总数 = 64K×8位 = 512K位 = 219位；

**思路：**如要满足地址线和数据线总和最小，应尽量把存储元安排在字向，因为地址位数和字数成2的幂的关系，可较好地压缩线数。

解：设地址线根数为a，数据线根数为b，则片容量为：2a×b = 219；b = 219-a；

若a = 19，b = 1，总和 = 19+1 = 20；

 a = 18，b = 2，总和 = 18+2 = 20；

  a = 17，b = 4，总和 = 17+4 = 21；

  a = 16，b = 8，总和 = 16+8 = 24；

  ……     ……

由上可看出：片字数越少，片字长越长，引脚数越多。片字数减1、片位数均按2的幂变化。

结论：如果满足地址线和数据线的总和为最小，这种芯片的引脚分配方案有两种：地址线 = 19根，数据线 = 1根；或地址线 = 18根，数据线 = 2根。

14. 某8位微型机地址码为18位，若使用4K×4位的RAM芯片组成模块板结构的存储器，试问：

（1）该机所允许的最大主存空间是多少？

（2）若每个模块板为32K×8位，共需几个模块板？

（3）每个模块板内共有几片RAM芯片？

（4）共有多少片RAM？

（5）CPU如何选择各模块板？

解：（1）该机所允许的最大主存空间是：218 × 8位 = 256K×8位 = 256KB

（2）模块板总数 = 256K×8 / 32K×8 = 8块

（3）板内片数 = 32K×8位 / 4K×4位 = 8×2 = 16片

（4）总片数 = 16片×8 = 128片

（5）CPU通过最高3位地址译码输出选择模板，次高3位地址译码输出选择芯片。地址格式分配如下：

20120504141521001

15. 设CPU共有16根地址线，8根数据线，并用20120504141521002（低电平有效）作访存控制信号，20120504141521003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-FFFFH

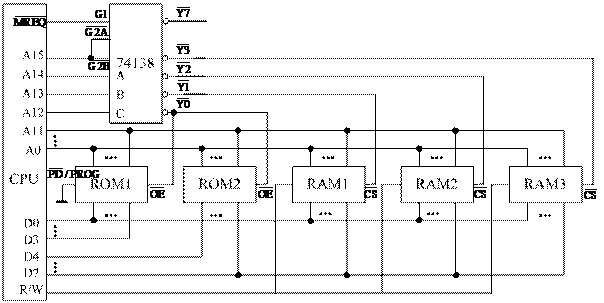
   （2）选片：ROM：选择4K×4位芯片2片，位并联

              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）

16. CPU假设同上题，现有8片8K×8位的RAM芯片与CPU相连，试回答：

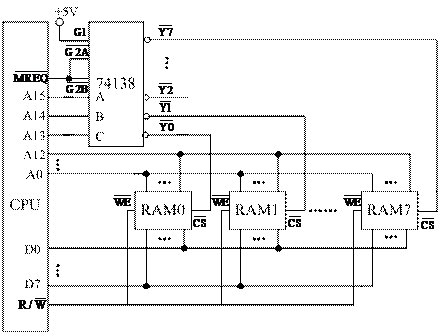
（1）用74138译码器画出CPU与存储芯片的连接图；

（2）写出每片RAM的地址范围；

（3）如果运行时发现不论往哪片RAM写入数据后，以A000H为起始地址的存储芯片都有与其相同的数据，分析故障原因。

（4）根据（1）的连接图，若出现地址线A13与CPU断线，并搭接到高电平上，将出现什么后果？

解：（1）CPU与存储器芯片连接逻辑图：



   （2）地址空间分配图：

        RAM0:0000H-1FFFH

        RAM1:2000H-3FFFH

        RAM2:4000H-5FFFH

        RAM3:6000H-7FFFH

        RAM4:8000H-9FFFH

        RAM5:A000H-BFFFH

        RAM6:C000H-DFFFH

        RAM7:E000H-FFFFH

（3）如果运行时发现不论往哪片RAM写入数据后，以A000H为起始地址的存储芯片(RAM5)都有与其相同的数据，则根本的故障原因为：该存储芯片的片选输入端很可能总是处于低电平。假设芯片与译码器本身都是好的，可能的情况有：

1）该片的-CS端与-WE端错连或短路；

2）该片的-CS端与CPU的-MREQ端错连或短路；

3）该片的-CS端与地线错连或短路。

（4）如果地址线A13与CPU断线，并搭接到高电平上，将会出现A13恒为“1”的情况。此时存储器只能寻址A13=1的地址空间(奇数片)，A13=0的另一半地址空间（偶数片）将永远访问不到。若对A13=0的地址空间（偶数片）进行访问，只能错误地访问到A13=1的对应空间(奇数片)中去。

17. 写出1100、1101、1110、1111对应的汉明码。

解：有效信息均为n=4位，假设有效信息用b4b3b2b1表示

校验位位数k=3位，（2k>=n+k+1）

设校验位分别为c1、c2、c3，则汉明码共4+3=7位，即：c1c2b4c3b3b2b1

校验位在汉明码中分别处于第1、2、4位

c1=b4⊕b3⊕b1

c2=b4⊕b2⊕b1

c3=b3⊕b2⊕b1

当有效信息为1100时，c3c2c1=011,汉明码为1110100。

当有效信息为1101时，c3c2c1=100,汉明码为0011101。

当有效信息为1110时，c3c2c1=101,汉明码为1011110。

当有效信息为1111时，c3c2c1=010,汉明码为0110111。

18. 已知收到的汉明码（按配偶原则配置）为1100100、1100111、1100000、1100001，检查上述代码是否

出错？第几位出错？

解：假设接收到的汉明码为：c1’c2’b4’c3’b3’b2’b1’

纠错过程如下：

P1=c1’⊕b4’⊕b3’⊕b1’

P2=c2’⊕b4’⊕b2’⊕b1’

P3=c3’⊕b3’⊕b2’⊕b1’

如果收到的汉明码为1100100，则p3p2p1=011，说明代码有错，第3位（b4’）出错，有效信息为：1100

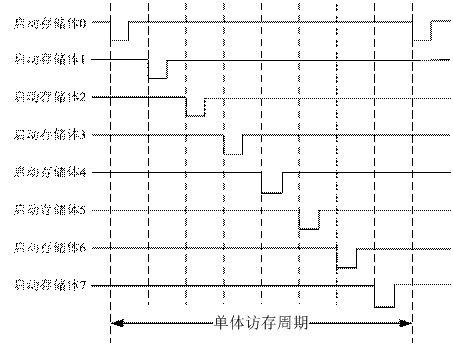
如果收到的汉明码为1100111，则p3p2p1=111，说明代码有错，第7位（b1’）出错，有效信息为：0110

如果收到的汉明码为1100000，则p3p2p1=110，说明代码有错，第6位（b2’）出错，有效信息为：0010

如果收到的汉明码为1100001，则p3p2p1=001，说明代码有错，第1位（c1’）出错，有效信息为：0001

22. 某机字长16位，常规的存储空间为64K字，若想不改用其他高速的存储芯片，而使访存速度提高到8倍，可采取什么措施？画图说明。

解：若想不改用高速存储芯片，而使访存速度提高到8倍，可采取八体交叉存取技术，8体交叉访问时序如下图：



23. 什么是“程序访问的局部性”？存储系统中哪一级采用了程序访问的局部性原理？

解：程序运行的局部性原理指：在一小段时间内，最近被访问过的程序和数据很可能再次被访问；在空间上，这些被访问的程序和数据往往集中在一小片存储区；在访问顺序上，指令顺序执行比转移执行的可能性大 (大约 5:1 )。存储系统中Cache—主存层次采用了程序访问的局部性原理。

25. Cache做在CPU芯片内有什么好处？将指令Cache和数据Cache分开又有什么好处？

答：Cache做在CPU芯片内主要有下面几个好处：

1）可提高外部总线的利用率。因为Cache在CPU芯片内，CPU访问Cache时不必占用外部总线。

2）Cache不占用外部总线就意味着外部总线可更多地支持I/O设备与主存的信息传输，增强了系统的整体效率。

3）可提高存取速度。因为Cache与CPU之间的数据通路大大缩短,故存取速度得以提高。

将指令Cache和数据Cache分开有如下好处：

1）可支持超前控制和流水线控制，有利于这类控制方式下指令预取操作的完成。

2）指令Cache可用ROM实现，以提高指令存取的可靠性。

3）数据Cache对不同数据类型的支持更为灵活，既可支持整数（例32位），也可支持浮点数据（如64位）。

**补充**：

Cache结构改进的第三个措施是分级实现，如二级缓存结构，即在片内Cache（L1）和主存之间再设一个片外Cache（L2），片外缓存既可以弥补片内缓存容量不够大的缺点，又可在主存与片内缓存间起到平滑速度差的作用，加速片内缓存的调入调出速度。

30. 一个组相连映射的CACHE由64块组成，每组内包含4块。主存包含4096块，每块由128字组成，访存地址为字地址。试问主存和高速存储器的地址各为几位？画出主存地址格式。

解：cache组数：64/4=16 ，Cache容量为：64\*128=213字，cache地址13位

主存共分4096/16=256区，每区16块

主存容量为：4096\*128=219字，主存地址19位，地址格式如下：

|  |  |  |
| --- | --- | --- |
| 主存字块标记（8位） | 组地址（4位） | 字块内地址（7位） |