

一、选择题（共 25 分，共 25 题，每题 1 分）（答案请填写入第一页答案处）

- 下列不属于冯·诺依曼体系结构定义的计算机硬件组成部分的是（ ）。  
A. 运算器                      B. Cache                      C. 控制器                      D. 输入输出设备
- 计算机的软硬件分层模型中，从低往上层次排列正确的是（ ）。  
A. 指令系统、数字逻辑电路、操作系统、MOC 管、应用软件  
B. 指令系统、MOS 管、数字逻辑电路、操作系统、应用软件  
C. MOS 管、数字逻辑电路、指令系统、操作系统、应用软件  
D. 数字逻辑电路、MOS 管、操作系统、指令系统、应用软件
- 处理器执行指令的过程包括①生成程序计数器 PC，②读取存储器中的指令，③送出存储器单元的地址，④存储器地址译码，⑤产生相应的控制信号，⑥指令译码等步骤。正确的指令执行的顺序是（ ）。  
A. ①②③④⑤⑥                      B. ①④⑤②③⑥  
C. ①③④②⑥⑤                      D. ①③②④⑤⑥
- 下列关于微处理器的描述中，符合哈佛结构特点的是（ ）。  
A. 指令和数据的表述可以不用二进制  
B. 指令和数据存储分别使用不同的存储器和独立的总线  
C. 采用 Cache 技术改变了存储器访问的串行性特征  
D. 采用流水线结构改变指令执行的串行性特征
- 对于一个 3 级流水线，当三段的延时为下列（ ）情况时，吞吐量最大。  
A. 50ps,80ps,150ps                      B. 50ps,80ps,160ps  
C. 60ps,80ps,120ps                      D. 30ps,80ps,150ps
- 根据 Flynn 分类法，传统的冯诺依曼计算机属于（ ）结构。  
A. SISD                      B. SIMD                      C. MIMD                      D. MISD
- 超标量技术可以在一定程度上缓解流水线的（ ）冲突。  
A. 控制                      B. 结构                      C. 数据                      D. 以上所有
- 若完成相同的计算任务，RISC CPU 的执行速度优于 CISC CPU，则很可能是因为（ ）。  
A. RISC 所使用的指令数量更少  
B. RISC 更易于使用流水线技术  
C. 程序在 RISC 上编译后的目标程序较短  
D. RISC 功能简单
- 下列寻址方式中，获取操作数最快的是（ ）。  
A. 寄存器直接寻址                      B. 基址变址寻址  
C. 存储器直接寻址                      D. 寄存器间接寻址
- 指令流水线各段之间会放置流水线寄存器，这种寄存器的主要作用是（ ）。  
A. 增强驱动                      B. 暂存中间结果，实现并行操作  
C. 缓解总线冲突                      D. 避免流水线冲突

11. 下列指标中,与数据通路宽度关系最小的是( )。
- A. 数据总线宽度                      B. 内部寄存器位数  
C. 指令长度                          D. 运算器位数
12. 键盘接入 AMBA 总线时,通常应连接到( )总线上。
- A. ASB                      B. APB                      C. AMB                      D. AHB
13. CPU 完成一次“存储器读”操作所需的时间称为一个( )。
- A. 指令周期                      B. 总线周期                      C. 时钟周期                      D. 中断周期
14. 下列总线中,不属于串行总线的是( )。
- A. DDR                      B. USB                      C. Ethernet                      D. PICE
15. 在单主控制器系统中,并行总线的操作周期分为寻址和传数两个阶段,下列描述不正确的是( )。
- A. 地址总线一般是单向的,仅能由主控设备发出  
B. 数据总线一般是双向的,但在某一个传数期一般也是单向的  
C. 寻址期除了发送地址信息外,还必须送出读写控制信号  
D. 寻址期还可能需要传输仲裁信号
16. Cache 技术和虚拟存储技术的相同点不包括( )。
- A. 都以访问局部性原理为基础                      B. 对用户均是透明的  
C. 采用类似的调度策略                      D. 均完全利用硬件技术实现
17. 设某异步串行通信的帧格式包含 8 位数据位、1 位校验位、1 位停止位。若波特率因子为 32,传输速率为 200 个字符/秒,则接收端时钟频率应为( )。
- A. 32KHz                      B. 51.2KHz                      C. 64KHz                      D. 70.4KHz
18. 在主存储器设计过程中,属于计算机系统结构范畴问题的是( )。
- A. 存储器的读写时钟频率                      B. 是否使用多体交叉结构  
C. 存储器的容量和数据对齐模式                      D. 片选译码方式
19. 若某微机系统中,存储器地址为 0x0000~0x1FFF,并行接口芯片地址范围为 0x0100~0x0103,可判断该系统中 I/O 采用了( )编址方式。
- A. 独立编址                      B. 统一编址  
C. 全译码编址                      D. 部分译码编址
20. 微处理器系统中,I/O 接口电路中控制端口信息一般来自( )。
- A. 系统地址总线                      B. 系统控制总线  
C. 系统数据总线                      D. 外设
21. ARM 等 RISC 微处理器采用 Load/Store 结构的主要目的在于( )。
- A. 减少处理器引脚                      B. 简化数据通路设计  
C. 减少寄存器数量                      D. 改善 I/O 性能
22. 当异常发生时,ARM 微处理器会自动将返回地址保存到寄存器( )中。
- A. R12                      B. R13                      C. R14                      D. R15
23. ARM 微处理器系统中,不能利用 MOV 指令传送到寄存器的立即数是( )。
- A. 0x1FF                      B. 0xAB                      C. 0x10C                      D. 0x104

24. 设 ARM 指令“STR R1, [R0, #16]!”执行前 R0=0xA000 0000, 则执行后 R0 的值为 ( )。

- A. 0xA0000000      B. 0xA0000016      C. 0xA0000010      D. 不确定

25. ARM 汇编源程序中, 关于“GPGCON EQU 0x56000060”语句的解释不正确的是 ( )。

- A. EQU 伪指令定义了一个标号或符号常量, 其值为 0x56000060  
B. EQU 伪指令定义了一个变量, 其初值为 0x56000060  
C. EQU 伪指令本身不占内存  
D. GPGCON 本身不占内存

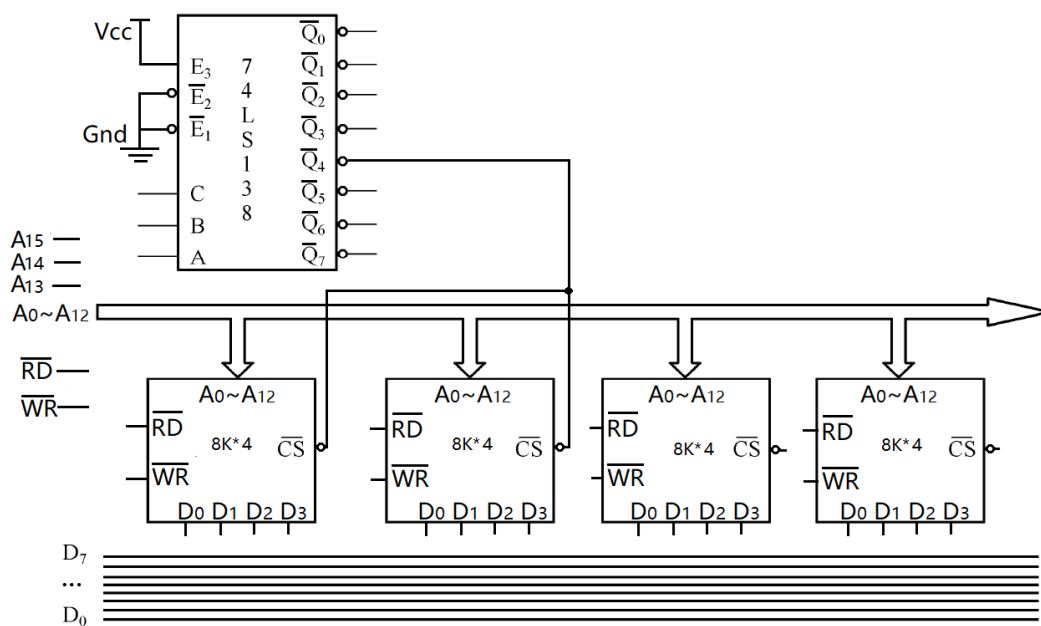
二、填空题 (每空 1 分, 共 20 分) (答案请集中填入第一页填空题答案处)

1. 串行性作为冯诺依曼型计算机的本质特点, 具体主要表现在①\_\_\_\_\_和②\_\_\_\_\_两方面。
2. 计算机系统可以采取多种措施提高并行性, 基本思想包括①\_\_\_\_\_, ②\_\_\_\_\_和③\_\_\_\_\_。
3. 流水线中的每个子过程及其功能部件称为流水线的级(段), 级(段)数也称为①\_\_\_\_\_。
4. 微处理器芯片设计的本质就是将抽象的处理器体系结构映射到半导体物理工艺上。在微处理器芯片的设计过程中, 首先需要确定产品规格书, 其中最重要的步骤就是拟定①\_\_\_\_\_。
5. 在采用微程序结构的处理器中, 其控制单元输入和输出之间的关系可被视为一个①\_\_\_\_\_, 而采用随机逻辑结构的微处理器则用②\_\_\_\_\_来表示控制单元输入和输出之间的关系。
6. 机器指令要素一般包含①\_\_\_\_\_, ②\_\_\_\_\_, ③\_\_\_\_\_和④\_\_\_\_\_等四个部分。
7. 微处理器系统中, 若数据压入堆栈时 SP 的值增加、且 SP 总是指向最后一个入栈数据, 则这种堆栈称为①\_\_\_\_\_堆栈。而按照 ATPCS 规则约定, ARM CPU 采用“C+汇编”混合编程方式时, 数据栈应采用②\_\_\_\_\_类型。
8. 将 0xAABBCCDD 存入 0x80000000 开始的内存单元, 若采用大端模式, 则 0x80000002 单元存放的字节是①\_\_\_\_\_。
9. 并行同步总线传输效率高, 但存在两个缺点: 一是由于存在①\_\_\_\_\_, 总线长度一般较短; 二是接入设备工作速度相差较大时, 传输速度主要受②\_\_\_\_\_限制。
10. 几乎所有的 ARM 指令都可以条件执行, 若欲表达“有符号数大于或等于”条件, 应在操作码后使用①\_\_\_\_\_条件助记符。
11. 在 ARM 微处理系统中, 通常异常的发生时间都具有不确定性。但是编程人员也可以人为安排①\_\_\_\_\_类型异常, 以便在确定时间引发异常处理。

### 三、综合题（共5题，共55分）

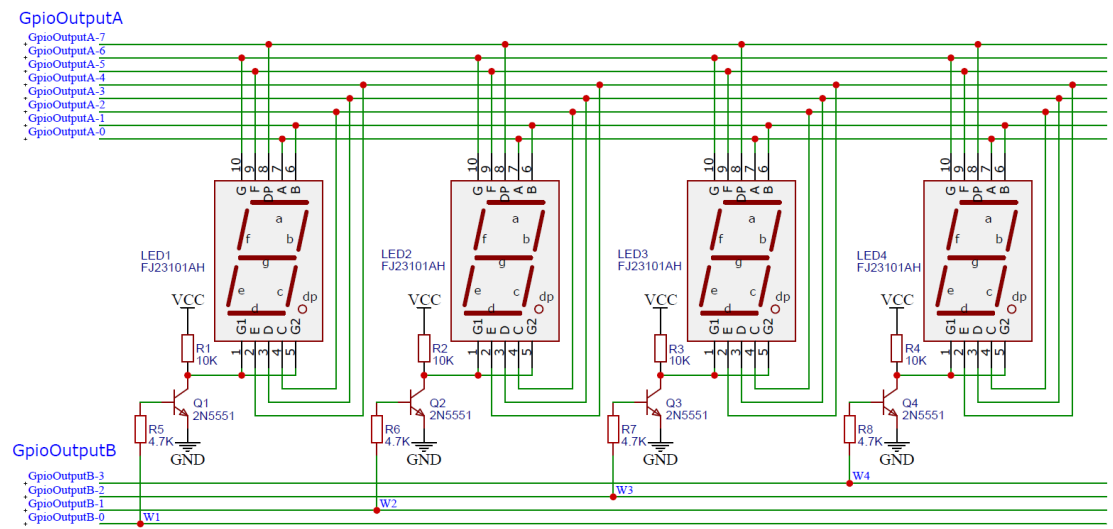
得 分

1. （8 分）请简述微型计算机系统总线的层次化结构特点，以及由此带来的性能改善？
2. （10 分）机器指令包括操作码和操作数两个部分。简述机器指令中的操作数可能来自于哪些硬件模块？为能够访问这些来自不同模块的操作数，在指令中分别可以使用哪些寻址方式？
3. （6 分）请简述无条件 I/O 接口和程序查询控制 I/O 接口各自的工作方式特点。
4. 某 8 位计算机系统中，地址总线宽度为 16bit。现需要给系统设计一块扩展存储器，要求其占用的地址空间为 0x8000~0xBFFF。  
 (1) （3 分）该扩展存储器的容量为多少 KB？  
 (2) （8 分）若使用单片容量为 8K\*4 位的 SRAM 芯片来搭建扩展存储器，则共需 4 片（分为 2 组）这样的 SRAM 芯片（如下图所示）。试解释为什么，并给出各芯片的地址范围。  
 (3) （6 分）在下图基础上添补连线，完成系统的连接示意图。



得分

5. 下图为 EES331 嵌入式系统开发板上的 8 段数码管接口电路：
- 位选端口 GpioOutputB 每一位对应一个数码管，从左至右为 LED1-LED4 数码管，对应位选的低位到高位。
  - 段选端口 GpioOutputA 每一位对应数码管中的每个显示段，从低位到高位依次对应 a,b,c,d,e,f,g,dp。
  - 数码管引脚 G1、G2 为数码管的公共端。



- (1) 4 个数码管中仅 LED1 被选中进行显示的时候，图中 W1 点为高电平“1”，W2~W4 点为低电平“0”，请问该数码管是共阴结构还是共阳结构的数码管？为什么？（4 分）
- (2) 若需在选中的 LED1 上显示字样“H.”，则端口 GpioOutputA 输出的段码应该是多少？（3 分）
- (3) 如果希望同时选中上述 4 个数码管（LED1-LED4）进行显示，且显示的内容完全一样，则端口 GpioOutputB 输出的位码应该为多少？（2 分）
- (4) 现希望四个数码管动态显示“1234”，请在下方指定位置给出后续驱动程序缺失的 C 代码，并给出必要注释。（基础的 I/O 端口设置函数、I/O 端口访问函数以及指向设备的指针，设备 ID，通道号等均已预先定义）（4 分）

库函数功能及相关参数说明

- int XGpio\_Initialize(XGpio \* InstancePtr, u16 DeviceId) :
- 功能：初始化 GPIO
- 参数：InstancePtr：指向设备的指针
- DeviceId：设备 id，可在 xparameters.h 中找到
- void XGpio\_SetDataDirection(XGpio \*InstancePtr, unsigned Channel,u32 DirectionMask) :
- 功能：设置 GPIO 为输入/输出

**参数：**InstancePtr：指向设备的指针

**Channel：**通道（1 或 2）

**DirectionMask：**标志设备输入输出（0xffffffff 为输入，0x0 为输出）

➤ **void XGpio\_DiscreteWrite(XGpio \* InstancePtr, unsigned Channel, u32 Data)：**

**功能：**读取 GPIO 的值

**参数：**InstancePtr：指向设备的指针

**Channel：**通道（1 或 2）

**Data：**将要写入 InstancePtr 指向寄存器的值

➤ **u32 XGpio\_DiscreteRead(XGpio \* InstancePtr, unsigned Channel)：**

**功能：**读取 InstancePtr 指向的寄存器的值，返回读取的数值

**参数：**InstancePtr：指向设备的指针

**Channel：**通道（1 或 2）

```
#include"xparameters.h"  
#include"xgpio.h"  
#include"xil_cache.h"
```

```
#define GPIO_SEG_W_DEVICE_ID 1  
#define GPIO_SEG_D_DEVICE_ID 0  
#define SEG_CHANNEL 1
```

```
XGpio GpioOutputB;  
XGpio GpioOutputA;
```

```
void main()
```

```
{  
    int status;  
    int w_val;  
    int d_val;  
    int data;
```

```
    w_val = 0x01;  
    data = 0x01;
```

```
    status = XGpio_Initialize(&GpioOutputB, GPIO_SEG_W_DEVICE_ID);  
    if (status != XST_SUCCESS)  
    {  
        return XST_FAILURE;
```

```
}
XGpio_SetDataDirection(&GpioOutputB, SEG_CHANNEL, 0x0);

status = XGpio_Initialize(&GpioOutputA, GPIO_SEG_D_DEVICE_ID);
if (status != XST_SUCCESS)
{
    return XST_FAILURE;
}
XGpio_SetDataDirection(&GpioOutputA, SEG_CHANNEL, 0x0);

while(1)
{
    XGpio_DiscreteWrite(&GpioOutputB, SEG_CHANNEL, w_val);
    w_val = w_val << 1;
    if (w_val > ① _____)
        w_val = ② _____;

    switch (data)
    {
        case 0:
            d_val = 0x3f;
            break;
        case 1:
            d_val = 0x06;
            break;
        case 2:
            d_val = 0x5b;
            break;
        case 3:
            d_val = 0x4f;
            break;
        case 4:
            d_val = 0x66;
            break;
        case 5:
            d_val = 0x6d;
            break;
        case 6:
            d_val = 0x7d;
            break;
        case 7:
            d_val = 0x07;
            break;
        case 8:
            d_val = 0x7f;
            break;
        case 9:
            d_val = 0x6f;
            break;
        default:
            d_val = 0x00;
            break;
    }
    XGpio_DiscreteWrite(&GpioOutputA, SEG_CHANNEL, d_val);
    data += 1;
    if (data > ③ _____)
        data = ④ _____;
} }
```