

电子科技大学 2021-2022 学年第 2 学期期 末 考试 A 卷

考试科目: 微处理器与嵌入式系统设计 考试形式: 一本书开卷 考试日期: 2022 年 6 月 7 日

成绩构成比例: 研讨班: 平时 30 %, 实验 20 %, 期末 50 %

普通班: 平时 20 %, 实验 20 %, 期末 60 %

本试卷由 三 部分构成, 共 10 页。考试时长: 120 分钟 注: 只允许带一本教材, 不能夹带习题集

题号	一	二	三(1)	三(2)	三(3)	三(4)	三(5)	三(6)	三(7)	合计
得分										

注意: 请将第一、二题答案填入指定位置。

得 分

一、选择题答案 (共 25 分, 共 25 题, 每题 1 分)

1	2	3	4	5	6	7	8	9	10	11	12	13
D	C	B	B	D	A	B	B	B	C	A	B	D
14	15	16	17	18	19	20	21	22	23	24	25	
A	B	A	C	C	C	C	C	D	A	B	C	

得 分

二、填空题答案 (共 20 分, 共 20 空, 每空 1 分)

- ① 流水线级数或段数
- ① 控制冒险, ② 数据冒险, ③ 结构/资源冒险 (后两空可不按顺序)
- ① 指令系统 (或指令集)
- ① PC 或程序计数器
- ① 数据传送类, ② 数据处理类, ③ 控制类 (可不按顺序)
- ① 命令或地址, ② 总线仲裁
- ① 独立, ② 统一
- ① 控制
- ① 立即数不合法
- ① 0x40000000
- ① R0~R3, ② 4
- ① 进程调度管理, ② 设备驱动程序

1. 根据 Flynn 分类法, 计算机体系结构可划分为以下 4 种类型, 其中 () 无实际机型对应。
A. SISD B. SIMD C. MIMD D. MISD
2. 以下主存设计中属于计算机系统结构角度考虑的是 ()。
A. 采用 MOS 还是 TTL B. 采用单体还是多体交叉
C. 容量和编址方式 D. 工作时钟频率
3. 下列**不属于** RISC 计算机常规设计原则的是 ()。
A. 指令条数少, 格式简单, 易于译码
B. 提供多种访问内存的方式, 以增加指令执行的灵活性
C. 提供足够的寄存器, 充分利用流水线
D. 指令由硬件直接执行, 在单个周期内完成
4. 某微处理器采用超标量结构, 是因为该处理器 ()。
A. 不仅能进行 32 位运算, 也能进行 64 位运算
B. 内部含有多条指令流水线和多个执行部件
C. 数据传输速度很快, 每个总线周期能传送多个数据
D. 芯片内部集成的晶体管数超过 100 万个, 功耗很大
5. 某微处理器指令中, 以寄存器中存放数值作为存储单元地址, 这种方式称为 () 寻址。
A. 立即 B. 寄存器直接 C. 存储器直接 D. 寄存器间接
6. CPU 的字长通常与系统中的 () 保持一致。
A. 数据总线宽度 B. 地址总线宽度 C. 寄存器个数 D. 控制总线宽度
7. 计算机系统中软硬件在逻辑上是等效的, 提高软件功能实现的比例将会 ()。
A. 提高解题速度 B. 提高系统灵活性 C. 增加成本 D. 减少所需存储容量
8. 微码控制器的特点是 ()。
A. 指令集设计必须和控制器设计一起同时考虑
B. 控制单元的输入和输出之间被视为一个存储器系统
C. 适合简单指令集的情况
D. 逻辑门数目达到最小化的设计
9. 在并行同步总线的一个操作周期中 ()。
A. 先传送数据, 再传输地址 B. 先传送地址, 再传输数据
C. 只传输数据 D. 只传输地址
10. 某总线采取菊花链仲裁方式, 则下列关于总线优先级的描述中, 正确的是 ()。
A. 总线授权的优先级由总线仲裁器确定 B. 由程序设定
C. 越靠近总线仲裁器的设备优先级越高 D. 响应速度越快的设备优先级越高
11. 串行通信使用奇偶校验时, 以下数据满足奇校验特征的是 ()。
A. 011001011 B. 111100101 C. 10101010 D. 010101010
12. 下列关于 AHB 总线的说法中, **不正确**的是 ()。
A. 支持突发传输 B. 提供丰富的外设接口 C. 支持多从机 D. 支持多主机

13. 一次总线事务中, 主设备只给出一个首地址就能够完成若干个连续地址中数据的读/写, 这种总线事务方式称为 ()。
- A. 并行传输 B. 同步传输 C. 异步传输 D. 突发传输
14. 微处理器系统中, 通常“中断向量”定义为 ()。
- A. 中断服务程序第一条指令的地址 B. 中断服务程序第一条指令
C. 中断发生的先后顺序 D. 中断发生的优先级
15. 下列叙述中, 可用于描述 RAM 存储器特点的是 ()。
- A. 可随机读写数据, 断电后数据不会丢失
B. 可随机读写数据, 断电后数据将全部丢失
C. 只能顺序读写数据, 断电后数据将部分丢失
D. 只能顺序读写数据, 断电后数据将全部丢失
16. 下列因素中, 与 Cache 命中率无关的是 ()。
- A. 主存存取时间 B. 块的大小 C. Cache 组织方式 D. Cache 容量
17. 微型计算机中的内存储器, 通常采用 ()。
- A. 磁表面存储器 B. 磁芯存储器 C. 半导体存储器 D. 光存储器
18. 在采用查询方式的输入输出接口中, 状态寄存器一般用来存放 ()。
- A. CPU 给外设的命令 B. 外设给 CPU 的命令
C. 外设的工作状态 D. CPU 的工作状态
19. 微处理器系统中, 接口电路通常都应该是 () 的。
- A. 可编程 B. 可中断 C. 可寻址 D. 可定时
20. 常用的输入/输出方式中, 便于 CPU 处理突发事件的是 () 控制方式。
- A. 无条件 B. 查询 C. 中断 D. DMA
21. ARM 微处理器系统中, 发生异常嵌套时, 异常处理程序应首先保护现场, 即将 LR、SPSR 及其它数据寄存器的值压入堆栈保存。在嵌套的异常处理程序返回前, 微处理器还必须恢复现场, 即将堆栈中保存的数据恢复到原来的寄存器中。下列汇编指令中, 能够完成“恢复现场”功能、从嵌套的异常处理程序正确返回的是 ()。
- A. LDMFD R13!, {R0,R2-R7,PC} B. LDMFD R13!, {R0,R2-R7}
C. LDMFD R13!, {R0,R2-R7,PC}^ D. LDMFD R13!, {R0,R2-R7}^
22. ARM 微处理器系统中, 设 IRQ 异常服务程序执行时 CPSR 中的 I=1。如果此时又发生数据中止异常, 则 ()。
- A. IRQ 异常服务程序返回后, 再执行数据中止异常服务程序。
B. 立即执行数据中止异常服务程序, 不再执行 IRQ 异常服务程序。
C. IRQ 异常服务程序返回后, 由于 I=1, 不执行数据中止异常服务程序。
D. 暂停 IRQ 异常服务程序的执行, 立即执行数据中止异常服务程序。
23. ARM 汇编语言程序中, 调用延时子程序 DELAY 应该使用的指令是 ()。
- A. BL DELAY B. BX DELAY C. B DELAY D. 以上均不正确

24. ARM 汇编语言程序中, 指令 `STMED SP!,{R0-R4,R7}` 操作的堆栈类型是 ()。
- A. 空递增 B. 空递减 C. 满递增 D. 满递减
25. S3C2440A 系统为低速外设提供的时钟是 ()。
- A. FCLK B. HCLK C. PCLK D. UCLK

一、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)

- “加速比”指程序在流水线上的执行速度与在等功能非流水线上的执行速度之比。理想情况下, 微处理器的最大加速比等于①_____。
- 流水线执行转移类指令时, 由于无法立即给出目标地址而产生的冒险称为①_____。流水线中其它常见冒险还包括②_____和③_____。
- 微处理器的实际硬件能力由它的①_____ 定义。
- 计算机在执行程序的过程中, 下一条待取指令的存放地址由CPU内部的①_____部件指示。
- 微处理器的基本指令主要可以分为 ①_____、②_____、③_____三大类。
- 总线主模块具有总线控制能力, 可向连接在总线上的其它模块发送①_____。在多主系统中, 某一时段的总线控制权只能由一个主模块掌握。因此在发生争用总线的情况时, 系统就必须进行②_____ 以确定当前哪一个主模块能够获取总线控制权。
- Intel x86系统的I/O端口采用①_____编址方式, 而ARM系统则采用②_____编址方式。
- 在计算机系统的三总线结构中, 用于传送读/写信号的是①_____总线。
- ARM微处理器系统中, 指令 `MOV R0,#0xf0c` 的错误原因是①_____。
- ARM微处理器系统中, 若寄存器R0的值是0x40000000, 则指令 `LDR R1,[R0,#8]` 执行后, R0的值应该是①_____。
- ARM的汇编语言程序可以调用C函数, 通用的C语言程序也可以调用ARM汇编子程序。根据ATPCS规范, 如果上述程序调用时需要传递参数, 则系统可以使用①_____来完成前②_____个参数的传递。
- 嵌入式Linux内核通常包括进程调度管理、内存管理、文件系统和设备驱动程序等4个部分的功能, 其中①_____是操作系统最核心、最基本的功能; 而设备树则属于②_____的一部分, 是一种用于描述输入输出接口信息的数据结构。

二、综合题（共7题，共55分）

得分

1. （6分）某测试程序在一个 100 MHz 的处理器上运行，其目标代码有 10000 条指令，由如下各类指令及其时钟周期计数混合组成，试确定这个程序的有效 CPI、MIPS 值以及总的执行时间 T。

指令类型	指令计数	时钟周期数
整数算术	5 000	1
数据传送	2 000	2
浮点数	2 000	2
控制传送	1 000	2

$$CPI = (5000/10000) * 1 + (2000/10000) * 2 + (2000/10000) * 2 + (1000/10000) * 2 = 1.5 \quad (2 \text{ 分})$$

$$MIPS = 100 / 1.5 = 66.67 \quad (2 \text{ 分})$$

$$T = (10000 * 1.5) * (1 / (100 * 10^6)) = 0.15 \text{ ms} \quad (2 \text{ 分})$$

得分

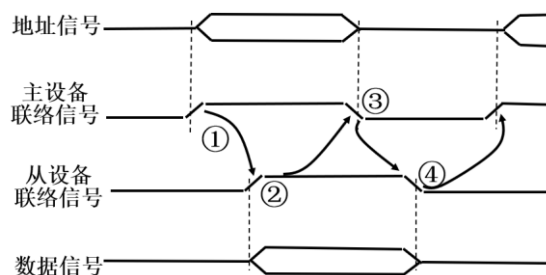
2. （6分）串行接口和并行接口相比，各自有什么特点？

串行接口是指数据一位位地顺序传送（1分）。其优点是信号线少、硬件简单、成本低，特别适用于远距离通信（1分）；缺点是相同时钟工作时，速度低于并行接口（1分）。

并行接口是指数据的各位同时进行传送（1分）。其优点是传输速度快（1分）；缺点是线路复杂、成本高，且高速速率下并行信号线之间的串扰严重（1分）。

得分

3. （8分）总线同步方式主要包括哪几种？下图所示时序属于哪种？这种同步方式有何优缺点？



总线同步方式主要包括同步（半同步）和异步几种（2分）。

图示为异步并行总线的时序关系（2分）。

异步总线通过主、从控制信号之间应答信号的交替变化来保证总线上地址和数据信息的准确传输，速度适应能力极强（2分）。

缺点是握手控制电路复杂，且握手信号来回应答时间长，效率较低（2分）。

得 分

4. （8分）现需要将16进制数据 0x123456789ABCDEF 以大端格式存放到内存中，首地址（16进制）为 0x37FFFFFC。请在下表中列出数据与内存地址之间的对应关系。

（每行1分）

地 址	数 据
0x37FFFFFC	0x01
0x37FFFFFD	0x23
0x37FFFFFE	0x45
0x37FFFFF	0x67
0x38000000	0x89
0x38000001	0xAB
0x38000002	0xCD
0x38000003	0xEF

得 分

5. （8分）试从设计开销和性能两个方面详细比较随机逻辑体系结构和微码体系结构的优缺点。

（1）指令集的改变导致不同的硬件设计开销：

在设计随机逻辑结构时，指令集和硬件必须同步设计和优化，因此设计随机逻辑的结构比设计微码结构复杂得多，而且硬件和指令集二者中任意一个变化，就会导致另外一个变化。（2分）

在微码结构中，指令设计通过为微码 ROM 编写微码程序来实现的，指令集的设计并不直接影响现有的硬件设计。因此，一旦修改了指令集，并不需要重新设计新的硬件。（2分）

（2）从性能上比较：

随机逻辑在指令集和硬件设计上都进行了优化，因此在采用相同指令集时随机逻辑更快一些。（2分）

但微码结构可以实现更复杂指令集，因此可以用较少的指令完成复杂的功能，尤其在存储器速度受限时，微码结构性能更优。（2分）

得 分

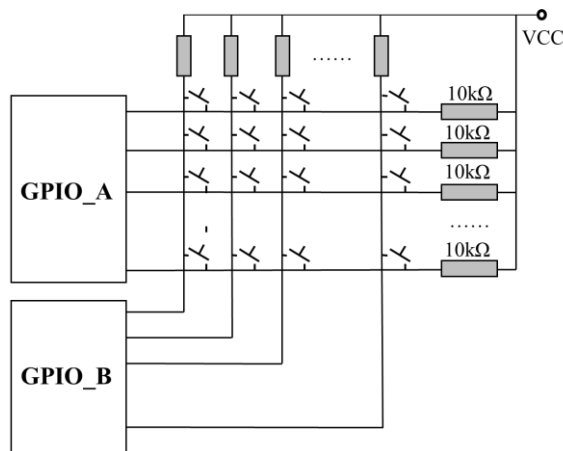
6. （8分）简述高速缓冲存储器技术和虚拟存储器技术的异同。

（1）相同点：a.以存储器访问的局部性为基础；（1分）b.采用的调度策略类似；（1分）c.对用户都是透明的。（2分）

（2）不同点：a.划分的信息块的长度不同（2分；）b.Cache 技术由硬件实现，而虚拟存储器由 OS 的存储管理软件辅助硬件实现（2分）。

得分

7. (11 分) 某 Zynq SoC 系统中构造了一个 8 行×8 列 (64 个按键) 的矩阵键盘, 设计时使用了两个 8bit GPIO 模块 (通道 1), 如下图所示。题后附图给出了 GPIO 模块地址分配情况以及端口寄存器说明。



- 1) (3 分) 请补充下面汇编程序段中空缺的部分, 完成对端口寄存器地址的说明定义。

```
GPIO_A_DATA EQU 0x41200000 ; GPIO_A 数据寄存器
GPIO_B_DATA EQU 0x41210000 ; GPIO_B 数据寄存器
GPIO_A_TRI EQU 0x41200004 ; GPIO_A 三态控制寄存器
GPIO_B_TRI EQU 0x41210004 ; GPIO_B 三态控制寄存器
GPIO_B_GIER EQU 0x4121011C ; GPIO_B 全局中断寄存器
GPIO_B_IER EQU 0x41210128 ; GPIO_B 中断允许寄存器
GPIO_B_ISR EQU 0x41210120 ; GPIO_B 中断状态寄存器
```

- 2) (8 分) 若使用行反转法进行键值判断, 请补充下面汇编程序段中空缺的部分, 完成第一个步骤“判断是否有按键按下”。

```
LDR R0, =GPIO_A_TRI
MOV R1, 0
STR R1, [R0] ; 将 GPIO_A 配置为输出端口
LDR R0, =GPIO_B_TRI
MOV R1, 0x000000FF
STR R1, [R0] ; 将 GPIO_B 配置为输入端口

LDR R0, =GPIO_A_DATA ; 令 GPIO_A 输出全 0
MOV R1, 0
STR R1, [R0]
LDR R0, =GPIO_B_DATA ; GPIO_B 以查询方式读入信息
CHECK LDRB R1, [R0]
CMP R1, 0xFF ; 判断是否有按键按下
BEQ CHECK ; 若无按键按下则反复查询
```

Cell	Slave Interface	Base Name	Offset Address	Range	High Address
processing_system7_0					
Data (32 address bits : 0x40000000 [1G])					
GPIO_A	S_AXI	Reg	0x4120_0000	64K	0x4120_FFFF
GPIO_B	S_AXI	Reg	0x4121_0000	64K	0x4121_FFFF

Address Space Offset ⁽³⁾	Register Name	Access Type	Default Value	Description
0x0000	GPIO_DATA	R/W	0x0	Channel 1 AXI GPIO Data Register.
0x0004	GPIO_TRI	R/W	0x0	Channel 1 AXI GPIO 3-state Control Register.
0x0008	GPIO2_DATA	R/W	0x0	Channel 2 AXI GPIO Data Register.
0x000C	GPIO2_TRI	R/W	0x0	Channel 2 AXI GPIO 3-state Control.
0x011C	GIER ⁽¹⁾	R/W	0x0	Global Interrupt Enable Register.
0x0128	IP IER ⁽¹⁾	R/W	0x0	IP Interrupt Enable Register (IP IER).
0x0120	IP ISR ⁽¹⁾	R/TOW ⁽²⁾	0x0	IP Interrupt Status Register.

AXI GPIO Data Register (GPIOx_DATA)

The AXI GPIO data register is used to read the general purpose input ports and write to the general purpose output ports. When a port is configured as input, writing to the AXI GPIO data register has no effect.

There are two GPIO data registers (GPIO_DATA and GPIO2_DATA), one corresponding to each channel. The channel 1 data register (GPIO_DATA) is always present; the channel 2 data register (GPIO2_DATA) is present only if the core is configured for dual channel (**Enable Dual Channel = 1**).

The AXI GPIO Data Register is shown in [Figure 2-1](#), and [Table 2-6](#) details the functionality of this register.

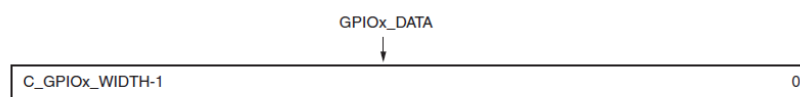


Figure 2-1: AXI GPIO Data Register

Table 2-6: AXI GPIO Data Register Description

Bits	Field Name	Access Type	Reset Value	Description
[GPIOx_Width-1 :0]	GPIOx_DATA	Read/Write	GPIO: Default Output Value GPIO2: Default Output Value	AXI GPIO Data Register. For each I/O bit programmed as input: <ul style="list-style-type: none"> R: Reads value on the input pin. W: No effect. For each I/O bit programmed as output: <ul style="list-style-type: none"> R: Reads to these bits always return zeros W: Writes value to the corresponding AXI GPIO data register bit and output pin.

AXI GPIO 3-State Control Register (GPIOx_TRI)

The AXI GPIO 3-state control register is used to configure the ports dynamically as input or output. When a bit within this register is set, the corresponding I/O port is configured as an input port. When a bit is cleared, the corresponding I/O port is configured as an output port.

There are two AXI GPIO 3-state control registers (GPIO_TRI and GPIO2_TRI), one corresponding to each channel. The channel 2 3-state control register (GPIO2_TRI) is present only if the core is configured for dual channel **Enable Dual Channel = 1**.

The AXI GPIO 3-state control register is shown in Figure 2-2; the register function is described in Table 2-7.

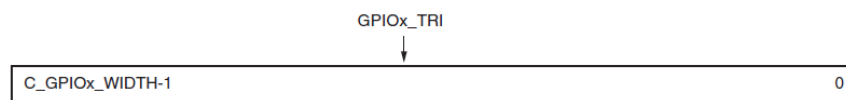


Figure 2-2: AXI GPIO Three-State Register

Table 2-7: AXI GPIO Three-State Register Description

Bits	Field Name	Access Type	Reset Value	Description
[GPIOx_Width-1 :0]	GPIOx_TRI	Read/Write	GPIO: Default Tri State Value GPIO2: Default Tri State Value	AXI GPIO 3-State Control Register. Each I/O pin of the AXI GPIO is individually programmable as an input or output. For each of the bits: 0 = I/O pin configured as output. 1 = I/O pin configured as input.

Global Interrupt Enable Register (GIER)

The Global Interrupt Enable register provides the master enable/disable for the interrupt output to the processor. This is a single-bit read/write register as shown in Figure 2-3. This register is valid only if the Enable Interrupt parameter is set.

Note: Because this is the master bit to control interrupt generation, it must be set to generate interrupts, even if the interrupts are enabled in the IP Interrupt Enable Register (IP IER). The bit definition for Global Interrupt Enable Register is given in Table 2-8.

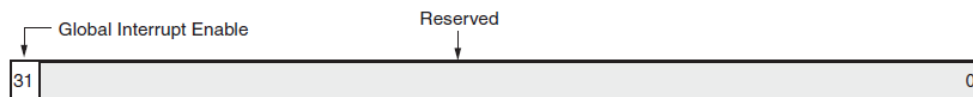


Figure 2-3: Global Interrupt Enable Register

Table 2-8: Global Interrupt Enable Register Description

Bits	Name	Core Access	Reset Value	Description
31	Global Interrupt Enable	Read/Write	0	Master enable for the device interrupt output to the system interrupt controller: 0 = Disabled 1 = Enabled
30 – 0	Reserved	N/A	0	Reserved. Set to zeros on a read.

IP Interrupt Enable (IPIER) and IP Status Registers (IPISR)

The IP Interrupt Enable Register (IPIER) and IP Interrupt Status Register (IPISR), shown in [Figure 2-4](#), provide a bit for each of the interrupts. These registers are valid only if the Enable Interrupt parameter is set.



IMPORTANT: The interrupt enable bits in the IP Interrupt Enable Register have a one-to-one correspondence with the status bits in the IP Interrupt Status Register. The interrupt events are registered in the IP Interrupt Status Register by the AXI4-Lite clock, and therefore the change in the input port must be stable for at least one clock period to guarantee interrupt capture. Each IPISR register bit can be set or cleared through software by the Toggle-On-Write behavior.

The bit definitions for IP Interrupt Enable Register and IP Interrupt Status Register are given in [Table 2-9](#) and [Table 2-10](#) respectively.

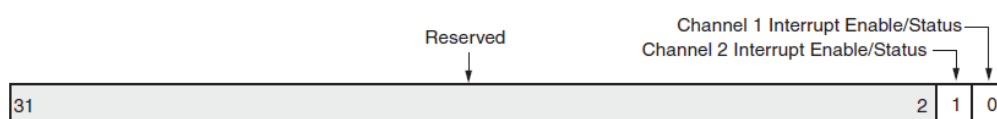


Figure 2-4: IP Interrupt Enable and IP Interrupt Status Register

Table 2-9: IP Interrupt Enable Register Description

Bits	Name	Core Access	Reset Value	Description
31-2	Reserved	N/A	0	Reserved. Set to zeros on a read.
1	Channel 2 Interrupt Enable	Read/Write	0	Enable Channel 2 Interrupt. 0 = Disabled (masked) 1 = Enabled
0	Channel 1 Interrupt Enable	Read/Write	0	Enable Channel 1 Interrupt. 0 = Disabled (masked) 1 = Enabled

Table 2-10: IP Interrupt Status Register Description

Bits	Name	Core Access	Reset Value	Description
31-2	Reserved	N/A	0	Reserved. Set to zeros on a read.
1	Channel 2 Interrupt Status	Read/TOW ⁽¹⁾	0	Channel 2 Interrupt Status 0 = No Channel 2 input interrupt 1 = Channel 2 input interrupt
0	Channel 1 Interrupt Status	Read/TOW ⁽¹⁾	0	Channel 1 Interrupt Status 0 = No Channel 1 input interrupt 1 = Channel 1 input interrupt

Notes:

1. Toggle-On-Write (TOW) access toggles the status of the bit when a value of 1 is written to the corresponding bit.