

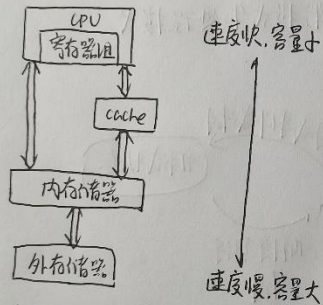
20375177 + 曹建秋

1/4

本人已知悉并将遵守《北京航空航天大学线上考试诚信承诺书》相关内容

- 一、(1) 1GB (2) 复杂指令集计算机(CISC) 精简指令集计算机(RISC)
- (3) 1439 EBCT (4) 0x2B1A 5675524
- (5) AAPCS (6) CPU的内部寄存器 内存
- (7) 3210739712 (8) 81.92 (9) 静态RAM 动态RAM NAND flash
- (10) 状态寄存器 中断/异常屏蔽寄存器 控制寄存器
- (11) 存储器统一编址方式 I/O端独立编址方式

三、1. 答: 计算机的层次式存储结构为



按照访问速度从高到低, 依次为CPU内部寄存器、高速缓存、内存和外存。  
解决CPU的处理速度与主存存取时间的差异是通过cache存储器实现的

2. 答: 总结 Arm Cortex 微处理器指令的寻址方式:

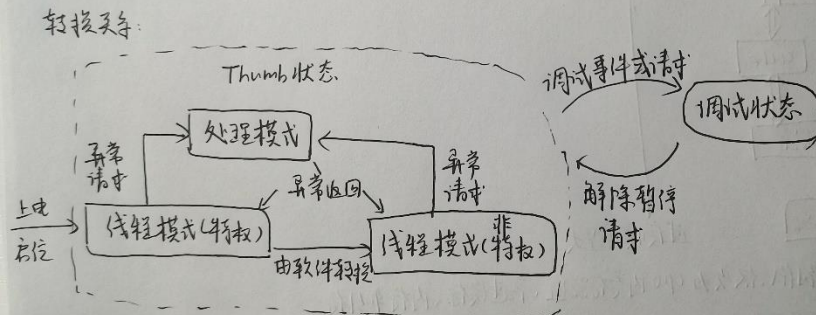
- ① 立即寻址: eg. SUB R1, #1
- ② 寄存器直接寻址: eg. SUB R1, R2, R3
- ③ 寄存器移位寻址: eg. SUB R1, R2, R3, LSR #2
- ④ 寄存器间接寻址: eg. STR R1, [R2]
- ⑤ 基址变址寻址: eg. STR R1, [R2, #4]
- ⑥ 多寄存器直接寻址: eg. LDMDB R0!, {R1-R3}
- ⑦ 堆栈寻址: eg. PUSH {R0-R5}

答: ARM系统总线包括数据总线DB、地址总线AB和控制总线CB  
地址总线用来传送地址信息,数据总线用来传送数据,控制总线传输控制信号  
CPU与I/O设备之间的接口信号包括数据,方向为双向;状态信息,方向为双向;  
控制信息,由CPU单向传送给I/O设备。

4. 答: CPU与外设传送数据方式包括: 无条件传送方式、查询传送方式、中断传送方式  
以及DMA方式。CPU与外设ADC之间有长时批量的数据传送,最好选用DMA方式,  
CPU与Timer之间有定时提醒需求,最好选用中断传送方式

5. 答: 主机发出一个起始信号开始一次总线操作,然后发出7个比特的地址0000001和一个比特的  
的命令0,然后地址为0x1的从机发送一个比特的有效应答信号0。此后主机传送字节0x3F,  
从机收到后发送一个比特的响应,主机最终发出一个停止信号结束

6. 答: 工作状态为调试状态和Thumb状态,模式有线程模式和处理模式



7. 答: 初始化的步骤

包括: 设定波特率和时钟频率、设置寄存器USART-BRR的值、时钟使能  
和配置USART-SR

8. 答: 简述中断过程:

① 初始化过程 → ② 中断请求过程 → ③ 中断响应过程  
→ ④ 中断服务 → ⑤ 中断返回

多个寄存器及返回地址压入栈  
取出中断向量  
同时更新程序计数器PC和内核寄存器的值

三. 1. 解:

```

* N      EQU      0x20000000
LDR      R1,      =N
CMP      R0,      #10
BGE      deadloop
CMP      R0,      #-10
BLE      deadloop
STR      R0,      [R1]

deadloop
B         deadloop
NOP
END

```

2. 解:

```

N1      EQU      456
N2      EQU      1278
N3      EQU      85
N4      EQU      -85
LDR      R0,      =N1
LDR      R1,      =N2
LDR      R3,      =N4
LDR      R2,      =N3
CMP      R0,      R1
MOVLE    R0,      R1
CMP      R0,      R2
MOVLE    R0,      R2
CMP      R0,      R3
MOVLE    R0,      R3

deadloop
B         deadloop
NOP
END

```



3. 解

GPIOA &= ~GPIO\_MODER - MODER0;

GPIOA → OTYPER &= ~GPIO\_OTYPER - OT\_0;

GPIOA &= ~GPIO\_MODER - MODER1;

GPIOA → PUPDR &= GPIO\_PUPDR - PUPDR1

4. 解

TheString DCB "The string is..."

LMP R0, #'z'

CMPLE R0, #'a'

ADPGE R0, #-32

LDRB R1, =TheString