

一、(14 分) 假设通过某不可靠信道传输 8 位原始数据, 具体数据如下

$D_8D_7D_6D_5D_4D_3D_2D_1=0110\ 1100$

生成多项式为 $x^6+x^5+x^3+x^2+x+1$

(1) 则生成多项式编码为 1101111, 假设校验位置于低位, 请给出最终的 CRC 校验码的十六进制编码。(4 分)

CRC 编码为: $D_8D_7D_6D_5D_4D_3D_2D_1R_5R_4R_3R_2R_1R_0=01101100\ 011110=0x1B1E$

(2) 如果接收方接收到的 CRC 校验码中校验位不变, 原始数据 $D_8D_7D_6D_5D_4D_3D_2D_1$ 变成 0110 1110, 请简单说明接收方如何定位错误。(4 分)

$1000\ 0000 \div 1101111 = 10001$ 余数不为零, CRC 编码不同位发生 1 位错的余数值是固定的, 所以可以根据余数的值定位出错位

(3) 在本课程 CRC 编码实验中我们尝试对 16 位汉字编码进行 CRC 编码, 你最终选择的生成多项式是_____位, 余数是_____位, 在假定没有 3 位以上错误发生的前提下, 该生成多项式生成的 CRC 编码检错时能否区分 1 位错和 2 位错? 为什么? (6 分)

答案 1: 多项式 6 位, 余数 5 位, 不能区分, 二者余数相同, 需要增加一个总校验位。

答案 2: 多项式 7 位, 余数 6 位, 能区分, 二者余数不同, 且都不为零, 可以直接根据余数区分。

二、(12 分) 已知 $[x]_{\text{补}} = 100001$, $[y]_{\text{补}} = 011101$, 用补码一位乘法计算 $[x \times y]_{\text{补}} = ?$ (单符号位)
将答案填写在下面, 并将计算过程填写在表格中。

1) $[-x]_{\text{补}} = \underline{011111}$ $[x \times y]_{\text{补}} = \underline{47D}$ (16 进制) (8 分)

2) 补码一位乘法运算速度较慢, 如何进一步优化乘法器速度? (2 分)

补码 2 位乘, 阵列乘法器, 乘法流水线

3) 假设运算结果只有 7 位, 第 1 问中的乘法运算是否溢出, 判断依据是什么? (2 分)

溢出, 高位应该与符号位完全相同。

#	运算	部分积	移出位	判断位 $y_n y_{n+1}$
1		<u>000000</u>		<u>0111010</u>
2	$+ [-x]_{\text{补}}$	<u>011111</u>		
3	=	<u>011111</u>		
4	\rightarrow	<u>001111</u>	1	<u>011101</u>
5	$+ [x]_{\text{补}}$	<u>100001</u>		
6	=	<u>110000</u>	1	
7	\rightarrow	<u>111000</u>	01	<u>01110</u>
8	$+ [-x]_{\text{补}}$	<u>011111</u>		
9	=	<u>010111</u>	01	
10	\rightarrow	<u>001011</u>	101	<u>0111</u>
11	+0	0		
12	=	<u>001011</u>	101	
13	\rightarrow	<u>000101</u>	1101	<u>011</u>
14	+0	0		
15	=	<u>000101</u>	1101	
16	\rightarrow	<u>000010</u>	11101	<u>01</u>
17	$+ [x]_{\text{补}}$	<u>100001</u>		
18	=	<u>100011</u>	11101	

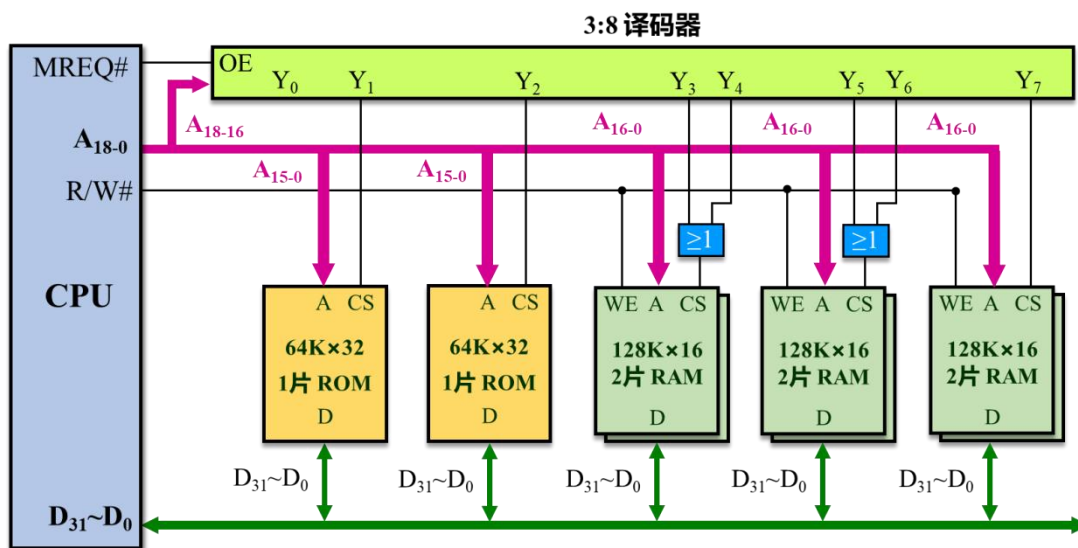
三、(12分) 某计算机字长为 32 位，主存容量为 $512K \times 32$ 位，其地址空间划分为：00000H ~ 0FFFFH 是保留区域，10000H ~ 2FFFFH 是 ROM 区域，30000H ~ 7FFFFH 是 RAM 区域。请用 $64K \times 32$ 位的 ROM 芯片和 $128K \times 16$ 位的 RAM 芯片为该计算机设计一个主存储器。

1) 共需要 2 片 ROM 芯片和 6 片 RAM 芯片，给出简单计算分析 (4 分)。

ROM 芯片规格为 $64K \times 32$ 位，构建 $128K \times 32$ 位存储器需要字数扩展至 128K，因此需要 **2 片 ROM 芯片**；

RAM 芯片规格为 $128K \times 16$ 位，构建 $320K \times 32$ 位存储器需要字长扩展至 32 位，字数扩展至 320K，因此需要 **6 片 RAM 芯片**。

2) 绘制 CPU 与存储芯片的连接示意图，注意标注译码器输出信号、地址线连接编号 (8 分)。



四、(12分) 在一个拥有 TLB 和一级数据缓存 (L1 d-cache) 的小存储系统中：存储器按字节寻址的，一次访问一个字节，虚拟地址是 14bit，物理地址是 12bit，页面大小 64B，TLB 四路组相联，总共 16 个条目，L1 d-cache 是物理地址寻址，直接映射，行大小为 4 字节，总共有 16 个组。下图展示了小存储系统的一个快照，包括 TLB (a)，部分页表 (b)，和 L1 高速缓存 (c)。

Set	Tag	PPN	Valid	Tag	PPN	Valid	Tag	PPN	Valid	Tag	PPN	Valid
0	03	—	0	09	0D	1	00	—	0	07	02	1
1	03	2D	1	02	—	0	04	—	0	0A	—	0
2	02	—	0	08	—	0	06	—	0	03	—	0
3	07	—	0	03	0D	1	0A	34	1	02	—	0

(a) TLB: 四组，16 个条目，四路组相联

VPN	PPN	Valid	VPN	PPN	Valid	Idx	Tag	Valid	Blk 0	Blk 1	Blk 2	Blk 3
00	28	1	08	13	1	0	19	1	99	11	23	11
01	—	0	09	17	1	1	15	0	—	—	—	—
02	33	1	0A	09	1	2	1B	1	00	02	04	08
03	02	1	0B	—	0	3	36	0	—	—	—	—
04	—	0	0C	—	0	4	32	1	43	6D	8F	09
05	16	1	0D	2D	1	5	0D	1	36	72	F0	1D
06	—	0	0E	11	1	6	31	0	—	—	—	—
07	—	0	0F	0D	1	7	16	1	11	C2	DF	03
						8	24	1	3A	00	51	89
						9	2D	0	—	—	—	—
						A	2D	1	93	15	DA	3B
						B	0B	0	—	—	—	—
						C	12	0	—	—	—	—
						D	16	1	04	96	34	15
						E	13	1	83	77	1B	D3
						F	14	0	—	—	—	—

(b) 页表前 16 个页表项 PTE

(c) L1 d-cache: 16 个组，四字节的块，直接映射

(1) 请问虚拟地址中虚拟页号(VPN)字段占 8 位，页内偏移(VPO)字段占 6 位，TLB 索引 (TLBI) 字段占 2 位，TLB 标记 (TLBT) 字段占 6 位。

(2) 物理地址中物理页号 (PPN) 字段占 6 位，cache 行索引字段占 4 位。

(3) 当 CPU 访问 0x3D4 处字节时，虚拟地址转换为物理地址时 PPN 的值为 0D，cache 行地址为 5，最终数据值为 36 (全部填写十六进制)。

(4) 当 CPU 访问 0x15F 处字节时，虚拟地址转换为物理地址时 PPN 的值为 16，cache 行地址为 7，最终数据值为 03 (全部填写十六进制)。

(最后一问题目错了，直接给分，其他每空一分)

五、(15 分) 下图为 MIPS32 指令格式，完成下列各问：

指令格式类型	MIPS 指令字长 32 位						说明
指令格式类型	字段名称						说明
指令格式类型	6	5	5	5	5	6	MIPS 指令字长为 32 位
R 型	OP	rs	rt	rd	shamt	funct	算术类指令，funct 为运算操作码
I 型	OP	rs	rt	地址/立即数			数据传输、分支、立即数指令
J 型	OP	目标地址					跳转指令

1) 已知 OP 为零时为 R 型指令，请问该指令格式最大能支持多少条指令，其中 R 型指令最多多少条？I 型和 J 型指令总和最多多少条？请写出理由。(4 分)

64,63

2) 指令是用户与硬件之间的接口，从本题给出的指令格式看，汇编级程序员可使用的源寄存器个数最多为多少？(3 分)

32

3) MIPS 通用寄存器中 0 号寄存器的意义是什么，MIPS CPU 为何要设计零号寄存器？(4 分)

恒零，便于置 0 操作，或者实现 MOV 指令，减少指令

4) beq 指令执行阶段的操作为 $PC \leftarrow PC + 4 + IR[15:0] \ll 2$ ，即将指令字中低 16 位左移 2 位后与 PC+4 的值相加生成 32 位地址，请问这里为什么要左移 2 位？(4 分)

指令对齐

六、(15 分) 某计算机的 CPU 主频为 2GHz, CPI 为 0.5, 总线带宽为 400MB/s。该计算机现有键盘和网卡两个设备, 键盘以 32 位的字为单位进行数据传输, 平均数据传输率为 1MB/s; 网卡以 2KB 的块大小与主机交换数据, 平均数据传输率为 400MB/s。

(1) 如果计算机对键盘采用程序查询方式进行数据输入, 查询操作需要 100 个时钟周期, 求 CPU 为 I/O 查询所花费的时间占整个 CPU 时间的百分比, 假定进行足够的查询以避免数据丢失。(4 分)

键盘采用程序查询方式, 每秒进行查询的次数为: $1\text{MB} / 4\text{B} = 250\text{K}$, 而查询 250K 次需要的时钟周期数为: $250\text{K} \times 100\text{T} = 25000\text{KT}$, 则可算出 CPU 为 I/O 查询所花费的时间比率为: $25000\text{K} / 2 \times 10^9 = 1.25\%$

(2) 如果计算机对键盘采用中断方式进行数据输入, 对应的中断服务程序包含 20 条指令, 中断服务的其他开销相当于 2 条指令的执行时间, 求 CPU 为该设备传输数据花费的时间占整个 CPU 时间的百分比。(4 分)

每秒 CPU 用于键盘 I/O 中断的次数为: $1\text{MB/s} / 4\text{B} = 250\text{K}$, 每次所需的时钟周期数为: $(20 + 2) \times 0.5 = 11$, 每秒用于中断方式数据传送的时钟周期数为: $250\text{K} \times 11 = 2.75\text{M}$, 因此占整个 CPU 时间的百分比为: $2.75\text{M} / 2 \times 10^9 = 0.1375\%$ 。

(3) 如果计算机对网卡采用 DMA 方式进行数据输入输出, DMA 预处理和后处理的总开销为 500 个时钟周期, 求 CPU 为该设备传输数据花费的时间占整个 CPU 时间的百分比。(4 分)

每秒 CPU 用于 DMA 的次数为: $400\text{MB/s} / 2\text{KB} = 200\text{K}$, 每次所需的时钟周期数为 500, 每秒用于 DMA 方式数据传送的时钟周期数为: $200\text{K} \times 500 = 100\text{M}$, 因此占整个 CPU 时间的百分比为: $100\text{M} / 2 \times 10^9 = 5\%$ 。

(5) 如果键盘采用中断方式, 网卡采用 DMA 方式。键盘和网卡能否同时工作, 为什么? 如果能够同时工作, 键盘和网卡谁的优先级高, 为什么? 如果不能同时工作, 则应该采取什么措施使之同时工作。(3 分)

键盘和网卡不能同时工作, 虽然键盘和网卡在进行 I/O 操作时所占用的整个 CPU 时间的百分比都非常小, 从 CPU 的角度来看是能够同时工作的; 但是该计算机系统的总线带宽只有 400MB/s, 仅能够满足网卡对总线带宽的需求, 因此不能同时工作。采取的措施是提高总线带宽。

七、(20 分) 某计算机字长 32 位，支持下表中的五条 MIPS32 指令，CPU 内部采用单总线结构，具体数据通路如图所示。除多路选择器选择控制信号外，图中所有控制信号为 1 时表示有效、为 0 时表示无效，控制信号功能说明见表。

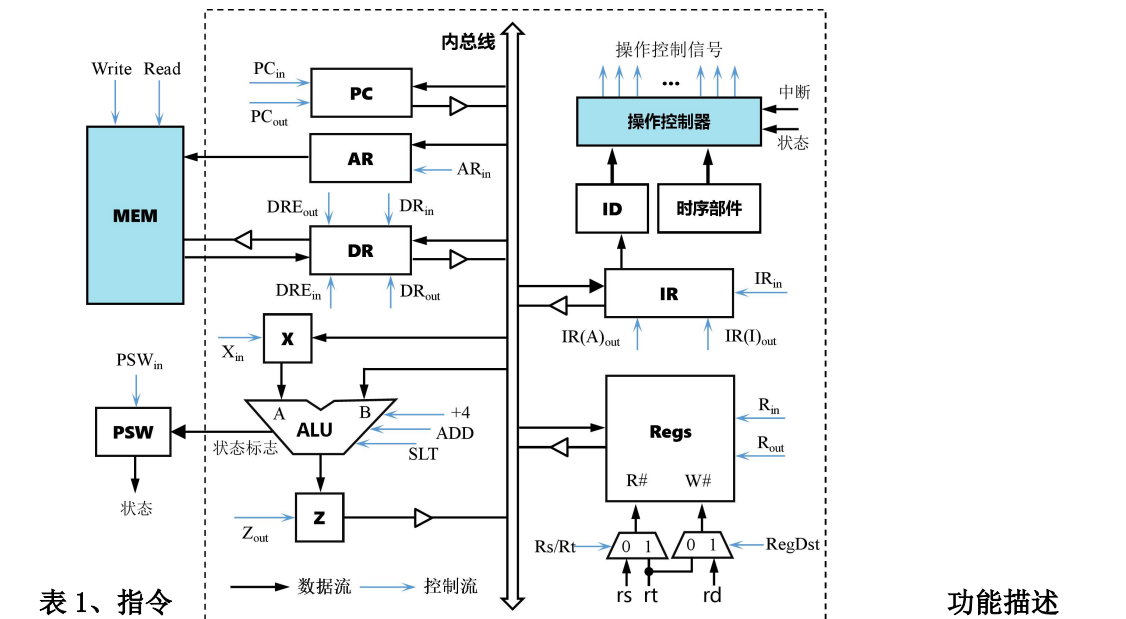


表 1、指令

#	指令	汇编代码	指令类型	RTL 功能说明
1	lw	lw rt,imm(rs)	I 型	$R[rd] \leftarrow M[R[rs] + \text{SignExt}(imm)]$
2	sw	sw rt,imm(rs)	I 型	$M[R[rs] + \text{SignExt}(imm)] \leftarrow R[rt]$
3	beq	beq rs,rt,imm	I 型	if($R[rs] == R[rt]$) $PC \leftarrow PC + 4 + \text{SignExt}(imm) \ll 2$
4	addi	addi rt,rs,imm	I 型	$R[rt] \leftarrow R[rs] + \text{SignExt}(imm)$
5	slt	slt rd,rs,rt	R 型	If ($rs < rt$) $R[rd] \leftarrow 1$ else $R[rd] \leftarrow 0$

表 2、控制信号功能描述

#	控制信号	功能说明
1	PCin	控制 PC 接收来自内总线的数据，需配合时钟控制
2	PCout	控制 PC 向内总线输出数据
3	ARin	控制 AR 接收来自内总线的数据，需配合时钟控制
4	DRin	控制 DR 接收来自内总线的数据，需配合时钟控制
5	DRout	控制 DR 向内总线输出数据
6	DREin	控制 DR 接收从主存读出的数据，需配合时钟控制
7	DREout	控制 DR 向主存输出数据，以便最后将该数据写入主存
8	Xin	控制暂存寄存器 X 接收来自内总线的数据，需配合时钟控制
9	+4	将 ALU A 端口的数据加 4 输出
10	ADD	控制 ALU 执行加法，实现 A 端口和 B 端口的两数相加
11	SLT	控制 ALU 执行 SLT 小于置位运算
12	PSWin	控制状态寄存器 PSW 接收 ALU 的运算状态，需配合时钟控制
13	Zout	控制 Z 向内总线输出数据
14	IRin	控制 IR 接收来自内总线的指令，需配合时钟控制
15	IR(A)out	控制 IR 中的分支目标地址输出到内总线，指令字中的立即数要转换成目标地址需要相应逻辑
16	IR(I)out	控制 IR 中的立即数输出到内部总线，指令字中的立即数符号扩展为 32 位才能输出
17	Write	存储器写命令，需配合时钟控制
18	Read	存储器读命令

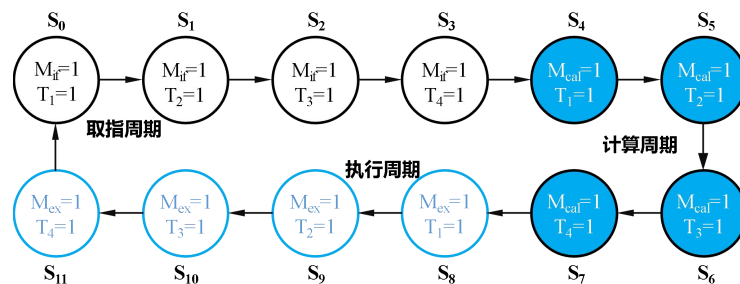
#	控制信号	功能说明
19	Rin	控制寄存器堆接收来自内总线的数据，写入 W#端口对应的寄存器中，需配合时钟控制
20	Rout	控制寄存器堆输出指定编号 R#寄存器的数据，该寄存器组为单端口输出
21	Rs/Rt	控制多路选择器选择送入 R#的寄存器编号，为 0 时送入指令字中 rs 字段，为 1 时送入 rt
22	RegDst	控制多路选择器选择送入 W#的寄存器编号，为 0 时送入指令字中的 rt 字段，为 1 时送入 rd

当 ALU 两操作数相等时，equal 状态信号输出为 1，否则为零，equal 信号与运算无关。

- 1) 图中 Z 部件是什么部件，其主要作用是什么，该部件在该结构中是否是必须的？（3 分）

Z 是运算结果缓冲器，在时钟配合下缓存运算结果，是必须的，否则运算结果直接输出到总线引起数据冲突。

- 2) 如果采用三级时序设计硬布线控制器，机器周期数，机器周期节拍数均固定，请给出三级时序发生器的状态机。（4 分）



- 3) 采用微程序设计操作控制器，取指微程序和 sw 指令微程序已部分给出，请用 16 进制完成如下填空。（9 分）

		PC _{out}	DR _{out}	Z _{out}	R _{out}	IR(A) _{out}	DRE _{out}	PC _{in}	AR _{in}	DRE _{in}	DR _{in}	X _{in}	R _{in}	IR _{in}	PSW _{in}	Rs/Rt	RegDst	ADD	+4	SLT	Read	Write	P _{IR}	P _{equal}		
功能	微地址	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	P0	P1	下址 (10 进制)
取指	0	1								1			1													1
	1																			1						2
	2			1					1		1															3
	3		1												1									1		X
sw	9				1							1														
	10					1													1							
	11			1						1																
	12				1							1					1									
	13							1															1			

第 2 条微指令控制字段为 85002 、 P 字段为 0 、 下址为 3

第 3 条微指令控制字段为 100100 、 P 字段为 2

第 12 条微指令控制字段为 40840

第 13 条微指令控制字段为 8001 、 P 字段为 0 、 下址为 0

- 4) 第 (2) 问如果需要在支持中断，状态图需要进行什么修改，第 (3) 问如需要在支持中断，P 字段是否需要修改，为什么？（4 分）

S₁₁ 状态需要进行中断判断，如果有中断请求需要进入中断响应周期，否则进入 S₀。

需要增加 P_{end} 表示最后是微程序的最后一条微指令，方便进行中断判断。