



2022150221 何泽锋

1. C 2. A 3. D 4. B B A

5. $0x^{\text{F}}\text{EFFFFE}$ DE $0x\text{EADFEED}$ 0 $0x\text{BFCD}$

6. mfhi \$t1 mflb \$t2.

7. jar myfunc1 jr \$ra

8. $0x1004$ $0x10B4$

9. a) addi \$t0, \$t0, 0;
add \$s0, \$s1, \$s2; // g+h
sll \$t0, \$t0, 4; // 地址偏移
add \$t1, \$t0, \$s7; // B[4]的地址
lw \$t3, 0(\$t1); // B[4]
add \$s0, \$s0, \$t3; // g+h+B[4]

b) addi \$t0, \$t0, 1;
sll \$t0, \$t0, 4;
add \$t1, \$t0, \$t7;
lw \$t0, 0(\$t1); // B[4]
add \$t0, \$t0, \$t6; // A[B[4]]的地址
lw \$t1, 0(\$t0); // A[B[4]]
add \$s0, \$s1, \$t1; // f = g - A[B[4]]

c) addi \$t0, \$t0, 4;
add \$t0, \$t0, \$t7;
lw \$t1, 0(\$t0); // B[1]
add \$s0, \$s1, \$s2; // f = g+h
add \$s0, \$s0, \$t1; // f = g+h+B[1]

```

d) sll $t0, $s2, 2; // g*4
   add lw $t1, $t0, $s7; // 8*B[g]
   lw $t0, 0($t1); // B[g]
   addi $t0, $t0, 1; // B[g]+1
   add lw $t1, $t0, $s6; // 4*A[B[g]+1]
   lw $t0, 0($t1); // A[B[g]+1]
   addi $s0, $t0, 0; // f = A[B[g]+1]

```

第二章

```

2.1 add $s5, $s1, $s2;
    sli $s5, $s2, 20;
    lw $s7, 50($s2);
    j lw 2500;
    jal lw 2500;

```

2.2 I型: 31-26 25-21 20-16 15-0
 op rs rt immediate.

R型: 31-26 25-21 20-16 15-11 10-6 5-0
 op rs rt rd shamt func

J型: 31-26 25-0
 op address

```

2.3 add $s0, $0, $0; // b=0
Loop: addi $s0, $s0, 2; // b=b+2
      addi $t0, $t0, -1; // i i=i-1
      slt $t3, $0, $t0; // 0 < i
      bne $t3, $0, Loop; #

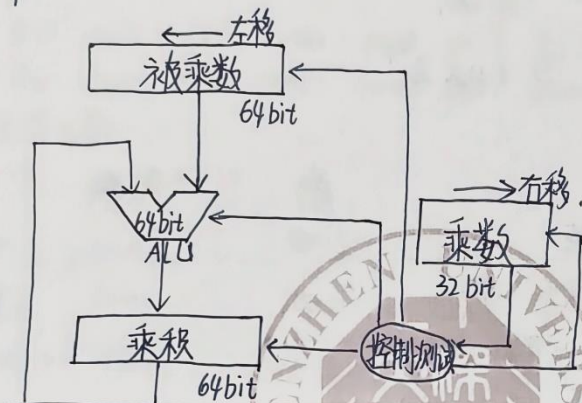
```



2022/5/22 何泽锋

第三章

3.1



轮次	步骤	乘数	被乘数	乘积
1	1a. $1 \Rightarrow \text{乘积} = \text{乘积} + \text{被乘数}$	10□□	0000 0101	0000 0101
	2. 被乘数左移	1011	0000 1010	0000 0101
	3. 乘数右移	0101	0000 1010	0000 0101
2	1a. $1 \Rightarrow \text{乘积} = \text{乘积} + \text{被乘数}$	010□	0000 1010	0000 1111
	2. 被乘数左移	0101	0001 0100	0000 1111
	3. 乘数右移	0010	0001 0100	0000 1111
3	1a. $0 \Rightarrow \text{无操作}$	0010	0001 0100	0000 1111
	2. 被乘数左移	0010	0010 1000	0000 1111
	3. 乘数右移	0001	0010 1000	0000 1111
4	1a. $1 \Rightarrow \text{乘积} = \text{乘积} + \text{被乘数}$	000□	0010 1000	0011 0111
	2. 被乘数左移	0001	0101 0000	0011 0111
	3. 乘数右移	0000	0101 0000	0011 0111

3.2 (d)

转换: $0.5_{10} = 1.0 \times 2^{-1}$

$-0.4375_{10} = -1.11 \times 2^{-2}$

对阶: $-1.11 \times 2^{-2} \Rightarrow -0.111 \times 2^{-1}$

尾数相加: $1.0 - 0.111 = 0.001$

规格化: $0.001 \times 2^{-1} \Rightarrow 1.0 \times 2^{-4}$

3.3

轮次	步骤	商	除数	余数
0	初始值	0000	0010	0000 0111
1	1. 余数 = 余数 - 除数	0000	0010	1110 0111
	2b. 余数 $\leq 0 \Rightarrow +$ 除数, 最低位 0		0010	0000 0111
	3. 余数左移, 最低位为 0		0010	0000 1110
2	1. 余数 = 余数 - 除数		0010	1110 1110
	2b. 余数 $\leq 0 \Rightarrow +$ 除数		0010	0000 1110
	3. 余数左移, 最低位为 0		0010	0001 1100
3	1. 余数 = 余数 - 除数		0010	1111 1100
	2b. 余数 $\leq 0 \Rightarrow +$ 除数		0010	0001 1100
	3. 余数左移, 最低位为 0		0010	0011 1000
4	1. 余数 = 余数 - 除数		0010	0001 1000
	2a. 余数 ≥ 0 , 余数左移, 最低位为 1		0010	0011 0001
				0011
5	1. 余数 = 余数 - 除数		0010	0001 0001
	2a. 余数 ≥ 0 , 余数左移, 最低位为 1		0010	0010 0011
				0011

因为多进行一次移位, 实际结束时先取余数 (高 4 位) 0001, 再进行移位, 得到 (低 4 位) 商 0011



2022150221 何泽锋

3.4

R_1 : 0000 0000 0000 0000 0001 0000 1000 1011

R_2 : 1000 0000 1000 0000 0001 0000 1000 1011

(1) 无符号数

R_1 : 0000108BH

R_2 : 8080108BH

(2) 带符号整数

R_1 : 0000108BH

R_2 : -7F7FEF75H

(3) 单精度浮点数

R_1 : 0.002116×2^{-126} H

R_2 : $-1.002116 \times 2^{-126}$ H

第四章

4.1 2) if (ID/EX.MemRead and
((ID/EX.RegisterRt = IF/ID.RegisterRs) or
(ID/EX.RegisterRt = IF/ID.RegisterRt)))
stall the pipeline.

3). PCWrite 和 IF/IDWrite 信号值为 0, 作用是阻止 PC 寄存器、IF/ID 寄存器的更新, 使得流水流不会往前推进, 达到阻塞效果。
EX 级使用的 WB/M/Ex 信号全为 0。

4.2

1) 无法优化指令序列

2) 使用延迟槽: $(6+1) \times 100 + 4 = 804$

假设 bne 没有优化. $(6+1) \times 100 + 4 = 704$.

开启分支预测: $8 \times 100 + 4 + 2 \times 2 = 808$.