

Homework 01

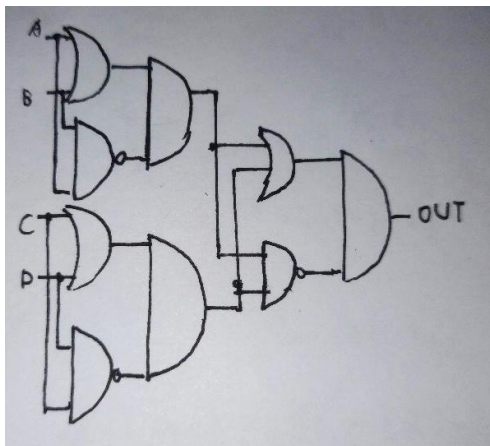
PB20000296 郑滕飞

1.

(a) $= (AB + BC + AC)(\bar{A} + \bar{B} + \bar{C})$, 由于 $A\bar{A} = B\bar{B} = C\bar{C} = 0$, 展开后只保留交叉项, 由此即为(b)的形式。

2.

实际电路为 $(A \wedge B) \wedge (C \wedge D)$, 当且仅当有奇数个 1 时输出 1。



3.

a.

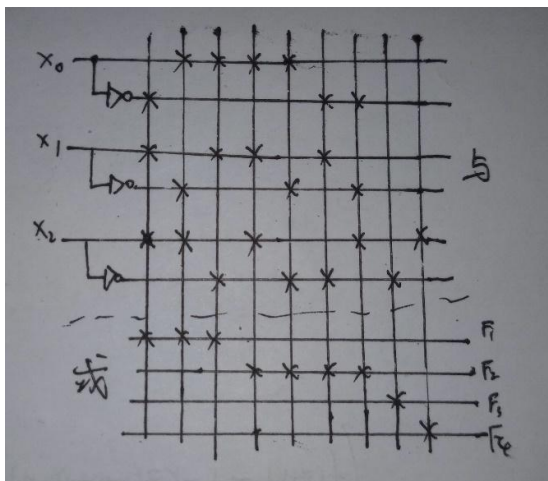
$$F_1 = x_2 x_1 \bar{x}_0 + x_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 x_0$$

$$F_2 = x_2 x_1 x_0 + \bar{x}_2 \cdot \bar{x}_1 x_0 + \bar{x}_2 x_1 \bar{x}_0 + x_2 \bar{x}_1 \cdot \bar{x}_0$$

$$F_3 = \bar{x}_2$$

$$F_4 = x_2$$

b.



4.

FUNC1: 2-1MUX, S 为 1 时输出为 I1, 否则为 I0。

FUNC2: 同步高电平复位为 0, 当控制信号为高电平时计数器每个时间周期加一, 否则减一。

5.

```
module adder(  
    input [15:0] LOAD,  
    input [3:0] In,  
    input Clk,  
    input Rst,  
    input Load,  
    output reg [15:0] Out  
);  
    always @(posedge Clk or posedge Rst) begin  
        if (Rst) Out <= 16'b0;  
        else if (Load) Out <= LOAD;  
        else Out <= Out + In;  
    end  
endmodule
```

Homework 02

PB20000296 郑腾飞

1.

(1)

处理器	时钟周期数	指令数
P1	3×10^{10}	2×10^{10}
P2	2.5×10^{10}	2.5×10^{10}
P3	4×10^{10}	1.8×10^{10}

(2)

设时钟频率变为 x ，则 $x \times (1 - 30\%) = 1 \times (1 + 20\%)$ ，解得 $x \approx 170\%$ 。

2.

(1)

单核：算术指令时间 1.28s，存取指令时间 7.68s，分支指令时间 0.64s，总时间 9.60s。

双核：算术指令时间 0.91s，存取指令时间 5.49s，分支指令时间 0.64s，总时间 7.04s。

四核：算术指令时间 0.46s，存取指令时间 2.74s，分支指令时间 0.64s，总时间 3.84s。

八核：算术指令时间 0.23s，存取指令时间 1.37s，分支指令时间 0.64s，总时间 2.24s。

双核加速比 1.36，四核加速比 2.5，八核加速比 4.29。

(2)

单核：算术指令时间 2.56s，存取指令时间 7.68s，分支指令时间 0.64s，总时间 10.88s。

双核：算术指令时间 1.83s，存取指令时间 5.49s，分支指令时间 0.64s，总时间 7.96s。

四核：算术指令时间 0.91s，存取指令时间 2.74s，分支指令时间 0.64s，总时间 4.29s。

八核：算术指令时间 0.46s，存取指令时间 1.37s，分支指令时间 0.64s，总时间 2.47s。

(3)

单核总时间比起四核增加 5.76s，存取指令时间需要为 1.92s，由此 CPI 需变为 3。

3.

(1)

P1 所需时间 1.125s，P2 所需时间 0.25s，因此 P1 时钟频率更高，但性能更低。

(2)

所需时间为 0.225s，P2 只能执行 9×10^8 条指令。

(3)

P1 为 4444，P2 为 4000，P1 的 MIPS 更高，但性能更低。

(4)

P1 为 1778，P2 为 1600。

4.

(1)

浮点操作时间减少 14s，总时间减少 5.6%。

(2)

总时间需减少 50s，整型操作总时间 55s，需减少 91%。

(3)

分支指令不足 50s，不可能。

Homework 03

PB20000296 郑滕飞

1.

(1)

addi x30, x10, 8 (I 型)

opcode:0010011 rs1:01010 rd:11110 imm:000000001000 func3:000

(2)

addi x31, x10, 0 (I 型)

opcode:0010011 rs1:01010 rd:11111 imm:000000000000 func3:000

(3)

sd x31, 0(x30) (S 型)

opcode:0100011 rs1:11110 rs2:11111 imm:000000000000 func3:011

(4)

ld x30, 0(x30) (I 型)

opcode:0000011 rs1:11110 rd:11110 imm:000000000000 func3:011

(5)

add x5, x30, x31 (R 型)

opcode:0110011 rs1:11110 rs2:11111 rd:00101 func7:0000000 func3:000

2.

64 位表示范围为 $-2^{63} \sim 2^{63} - 1$, 由此:

(1)

$$x_6 \geq 2^{63} - 128$$

(2)

$$x_6 \leq 128 - 2^{63}$$

(3)

$$x_6 \leq 127 - 2^{63}$$

3.

(1)

20 位偏移量, 范围 0xFFF00000-0x000FFFFE, 故 PC 范围 0x1FF00000-0x200FFFFE。

(2)

12 位偏移量, 范围 0xFFFF0000-0x00000FFE, 故 PC 范围 0x1FFFF000-0x20000FFE。

4.

(1)

```
int i;  
for (i = 0; i < 100; i++) result += MemArray[i];
```

(2)

```
addi x6, x10, 0  
addi x29, x10, 400  
LOOP: lw x7, 0(x6)  
add x5, x5, x7
```

```
addi x6, x6, 4  
blt x6, x29, LOOP
```

5.

(1)

0x1122334455667788 大端对齐排列 11,22,33,44,55,66,77,88,
读入 11, 寄存器 0000000000000011,
大端对齐存入 00,00,00,00,00,00,00,11,
故结果为 00。

(2)

0x1122334455667788 小端对齐排列 88,77,66,55,44,33,22,11,
读入 88, 寄存器 ffffffff88,
小端对齐存入 88,ff,ff,ff,ff,ff,ff,ff,
故结果为 88。

Homework 04

PB20000296 郑腾飞

1.

$$78.75 = 2^{-2} \times 315 = 2^6 \times (1.00111011)_2$$

由此可表示为 0 10000101 0011101100000000000000 16 进制为 0x429d8000

2.

(1)

只有 Load 与 Store 会发生数据访存，比例为 39%。

(2)

所有指令都要进行指令访存，比例为 100%。

(3)

除了 R-type 以外均使用符号扩展，比例为 77%。

3.

指令为 0000000 01100 01101 011 10100 0100011 即 sd x13, 20(x12)

(1)

ALUOp 为 00, ALU 控制线为 0010。

(2)

不发生跳转，故 PC 为 0xbfc00390，通路 1->3->12->1。

(3)

ALU 输入值为 Reg[x13] 与 20。

加法单元 3 的输入值为 0xbfc0038c 与 4。

加法单元 11 的输入值为 0xbfc0038c 与 40 (20 左移一位后的结果)。

4.

(1)

最长通路 1->2->4->7->9->14->4

延迟 40+235+160+45+230+45+15=770ps

(2)

由于 5 和 7 总延迟没有 4 高，最长通路 1->2->4->9->13->14->4

延迟 40+235+160+230+235+45+15=960ps

(3)

最长通路 1->2->4->9->13

延迟 40+235+160+230+235=900ps

(4)

最长通路 1->2->4->7->9->与门->12->1

延迟 40+235+160+45+230+10+45+15=780ps

(5)

最长通路 1->2->4->7->9->14->4

延迟 40+235+160+45+230+45+15=770ps

(6)

最小时钟周期 960ps

Homework 05

PB20000296 郑腾飞

1.

(1)

流水化：取最大值知为 350ps。

非流水化：250+350+150+300+200=1250ps。

(2)

流水化：执行时间为五级 350ps，从而为 1750ps。

非流水线化：由于五步骤都需要，为 1250ps。

(3)

拆分最大的一级才有意义，因此拆分 ID，新的最大值为 MEM 的 300ps，即为新周期。

(4)

35%的指令(Load/Store)需要用数据储存器，由此利用率约 35%。

(5)

65%的指令(ALU/Logic/Load)需要写入寄存器堆，由此利用率约 65%。

2.

(1)

	CC 1	CC 2	CC 3	CC 4	CC 5	CC 6
sd x29, 12(x16)	IF	ID	EX	MEM	WB	
ld x29, 8(x16)		IF	ID	EX	MEM	WB
sub x17, x15, x14			IF	ID	EX	MEM
beqz x17, label				IF	IF	IF
add x15, x11, x14						
sub x15, x30, x14						

如上方所示，由于结构冒险会在第 4、5 个时钟周期停顿

(2)

不能，每条指令都需访问储存器(就所给代码而言可以，即将 ld, sd 放在最后)。

(3)

不能。代码中的 NOP 仍然需要 IF，仍会产生结构冒险。

(4)

ld 与 sd 均会导致停顿，因此会使时钟周期增加 35%。

3.

(1)

不会。最耗时间的部分不可能减少。

(2)

有可能，如：

add x3, x2, x1

add x10, x0, x1

add x5, x4, x3

五级流水线时第一条与第三条指令有数据冒险，但四级时不存在冒险。

(3)

有可能。需要用 `addi` 与 `ld/sd` 才能实现原本效果，增加指令。

4.
(1)

	CC 1	CC 2	CC 3	CC 4	CC 5	CC 6	CC 7	CC 8	CC 9	CC 10	CC 11	CC 12	CC 13	CC 14	CC 15	CC 16
ld x10, 0(x13)	IF	ID	EX	MEM	WB											
ld x11, 8(x13)		IF	ID	EX	MEM	WB										
BUBBLE			ID													
add x12, x10, x11				IF	ID	EX		WB								
subi x13, x13, 16					IF	ID	EX		WB							
bnez x12, LOOP						IF	ID	EX								
ld x10, 0(x13)							IF	ID	EX	MEM	WB					
ld x11, 8(x13)								IF	ID	EX	MEM	WB				
BUBBLE									ID							
add x12, x10, x11									IF	ID	EX			WB		
subi x13, x13, 16										ID	EX				WB	
bnez x12, LOOP											IF	ID	EX			

(2)

上图中为空代表没有进行有用的操作，因此实质上不会五个流水线级都有用。

Homework 06

PB20000296 郑滕飞

1.

(1)

字地址	二进制	标签	索引	Hit/Miss
0x03	00000011	0000	0011	Miss
0xb4	10110100	1011	0100	Miss
0x2b	00101011	0010	1011	Miss
0x02	00000010	0000	0010	Miss
0xbf	10111111	1011	1111	Miss
0x58	01011000	0101	1000	Miss
0xbe	10111110	1011	1110	Miss
0x0e	00001110	0000	1110	Miss
0xb5	10110101	1011	0101	Miss
0x2c	00101100	0010	1100	Miss
0xba	10111010	1011	1010	Miss
0xfd	11111101	1111	1101	Miss

(2)

字地址	二进制	标签	索引	偏移	Hit/Miss
0x03	00000011	0000	001	1	Miss
0xb4	10110100	1011	010	0	Miss
0x2b	00101011	0010	101	1	Miss
0x02	00000010	0000	001	0	Hit
0xbf	10111111	1011	111	1	Miss
0x58	01011000	0101	100	0	Miss
0xbe	10111110	1011	111	0	Hit
0x0e	00001110	0000	111	0	Miss
0xb5	10110101	1011	010	1	Hit
0x2c	00101100	0010	110	0	Miss
0xba	10111010	1011	101	0	Miss
0xfd	11111101	1111	110	1	Miss

2.

(1)

$1000 \times 0.3\% + (250 + 100) \times 2\% = 10$ ，即每 1000 条指令预计产生 10 次读失效，即 2000 个时钟周期产生 $10 \times 64 = 640$ 个字节的读请求，读带宽至少为 0.32 字节每周期。所有写请求都更新缓存，因此 1000 条指令需要进行 100 次写入，每次写入 4 字节(即储存器位宽)，写带宽至少为 0.2 字节每周期。

(2)

读的情况与第一问一致，仍为 0.32 字节每周期。

对所有脏块，读写都会导致更换写回，因此 1000 条指令产生 $350 \times 2\% \times 30\% = 2.1$ 次写请求，写带宽至少为 0.0672 字节每周期。

3.

命中率为 $\frac{2000}{2000+50} \approx 97.56\%$, 未命中惩罚为 150ns。

未命中率 2.44%, 平均访问时间 $50 + 2.44\% \times 150 = 53.66$ 纳秒。

4.

(1)

100ns 在 2GHz 下为 200 个时钟周期。

仅 L1: $1.5 + 7\% \times 200 = 15.5$ 周期

直接 L2: $1.5 + 7\% \times 12 + 3.5\% \times 200 = 9.34$ 周期

组相联 L2: $1.5 + 7\% \times 28 + 1.5\% \times 200 = 6.46$ 周期

(2)

$1.5 + 7\% \times 12 + 3.5\% \times (50 + 13\% \times 200) = 5$ 周期

(3)

*题干有误, 4%应为全局失效率, 不然与失效率高的描述不符

$9.34 = 1.5 + 7\% \times 50 + (4 - 0.7x)\% \times 200$, 解得 $2 < x < 3$, 即添加三片(至少 4 片)可以满足要求。

Homework 07

PB20000296 郑滕飞

1.
(1)

地址	虚拟页	TLB 命中	页命中	缺页错误	TLB			
					有效位	上次访问	标签	物理页
0x123d	1	M	H	Fault	1	5	b	12
					1	2	7	4
					1	4	3	6
					1	0	1	13
0x08b3	0	M	H		1	0	0	5
					1	3	7	4
					1	5	3	6
					1	1	1	13
0x356c	3	H			1	1	0	5
					1	4	7	4
					1	0	3	6
					1	2	1	13
0x871b	8	M	H	Fault	1	2	0	5
					1	0	8	14
					1	1	3	6
					1	3	1	13
0xbec6	b	M	H		1	3	0	5
					1	1	8	14
					1	2	3	6
					1	0	b	12
0x3140	3	H			1	4	0	5
					1	2	8	14
					1	0	3	6
					1	1	b	12
0xc049	c	M	M	Fault	1	0	c	15
					1	3	8	14
					1	1	3	6
					1	2	b	12

(2)

优势：更多命中，更少缺页错误；缺点：碎片化加重。

地址	虚拟页	TLB 命中	页命中	缺页错误	TLB			
					有效位	上次访问	标签	物理页
0x123d	0	M	H		1	5	b	12
					1	2	7	4
					1	4	3	6
					1	0	0	5

0x08b3	0	H			1	6	b	12
					1	3	7	4
					1	5	3	6
					1	0	0	5
0x356c	0	H			1	7	b	12
					1	4	7	4
					1	6	3	6
					1	0	0	5
0x871b	2	M	H	Fault	1	0	2	13
					1	5	7	4
					1	7	3	6
					1	1	0	5
0xbec6	2	H			1	0	2	13
					1	6	7	4
					1	8	3	6
					1	2	0	5
0x3140	0	H			1	1	2	13
					1	7	7	4
					1	9	3	6
					1	0	0	5
0xc049	3	H			1	2	2	13
					1	8	7	4
					1	0	3	6
					1	1	0	5

(3)

地址	虚拟页	标签 /索引	TLB 命中	页命中 /错误	TLB(索引依次 0011)			
					有效位	上次访问	标签	物理页
0x123d	1	0/1	M	H/Fault	1	5	b	12
					1	2	7	4
					1	4	3	6
					1	0	0	13
0x08b3	0	0/0	M	H	1	0	0	5
					1	3	7	4
					1	5	3	6
					1	1	0	13
0x356c	3	1/1	M	H	1	1	0	5
					1	4	7	4
					1	0	1	6
					1	2	0	13
0x871b	8	4/0	M	H/Fault	1	2	0	5
					1	0	4	14
					1	1	1	6
					1	3	0	13

0xbbee6	b	5/1	M	H	1	3	0	5
					1	1	4	14
					1	2	1	6
					1	0	5	12
0x3140	3	1/1	H		1	4	0	5
					1	2	4	14
					1	0	1	6
					1	1	5	12
0xc049	c	6/0	M	M/Fault	1	0	6	15
					1	3	4	14
					1	1	1	6
					1	2	5	12

(4)

地址	虚拟页	标签/索引	TLB 命中	页命中/错误	TLB(索引依次 0123)			
					有效位	上次访问	标签	物理页
0x123d	1	0/1	M	H/Fault	1	5	b	12
					1	0	0	13
					1	4	3	6
					0	8	4	9
0x08b3	0	0/0	M	H	1	0	0	5
					1	1	0	13
					1	5	3	6
					0	9	4	9
0x356c	3	0/3	M	H	1	1	0	5
					1	2	0	13
					1	6	3	6
					1	0	0	6
0x871b	8	2/0	M	H/Fault	1	0	2	14
					1	3	0	13
					1	7	3	6
					1	1	0	6
0xbbee6	b	2/3	M	H	1	1	2	14
					1	4	0	13
					1	8	3	6
					1	0	2	12
0x3140	3	0/3	M	H	1	2	2	14
					1	5	0	13
					1	9	3	6
					1	0	0	6
0xc049	c	3/0	M	M/Fault	1	0	3	15
					1	6	0	13
					1	10	3	6
					1	1	0	6

(5)

没有 TLB 就只能先访问内存中的页表，再访问对应的页，需要两次访问，TLB 可以让第一次发生概率大大减小。

2.

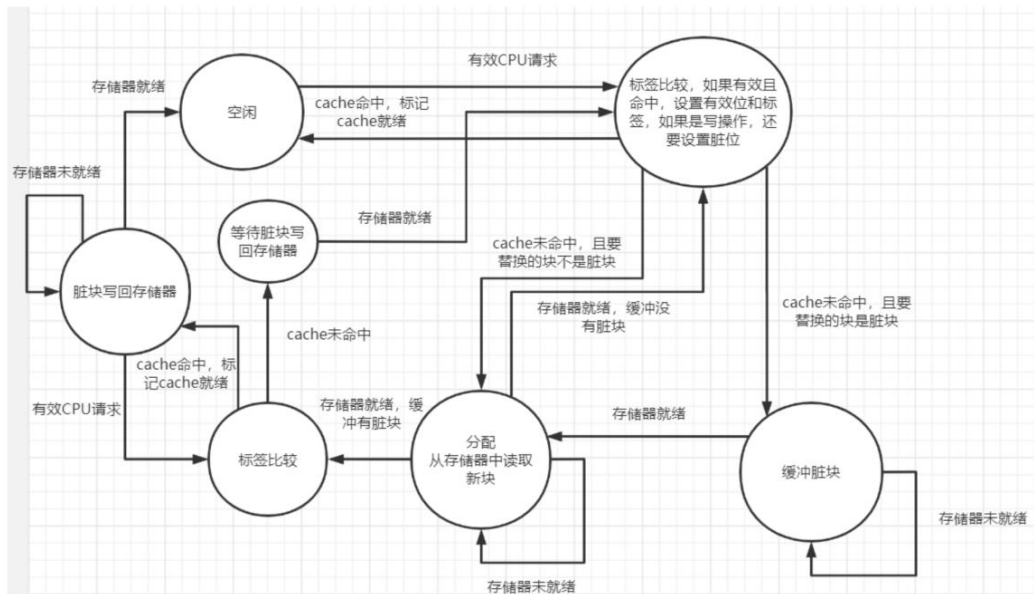
(1)

由于缓存与写缓冲区分开，可以直接读取缓存。

(2)

等待写缓冲区完成才能访问主存。

(3)



3.

(1)

01111100、1010101、00101110、11111111

(2)

1100100 第 6 位错; 1100111 第 7 位错; 1100000 第 3 位错; 1100001 第 4 位错

4.

每转一周平均时间 $60 \div 10000 \div 2 = 0.003s$

传输时间 $4 \div 20 \div 1024 \approx 0.0002s$

总时间 $6 + 3 + 0.2 + 0.2 = 9.4ms$