

实验二 模型机组合部件的实现（一）

班级 计科 2003 姓名 袁鹏 学号 202008010321

一、实验目的

1. 了解简易模型机的内部结构和工作原理。
2. 熟悉译码器、运算器的工作原理。
3. 分析模型机的功能，设计指令译码器。
4. 分析模型机的功能，设计 ALU。

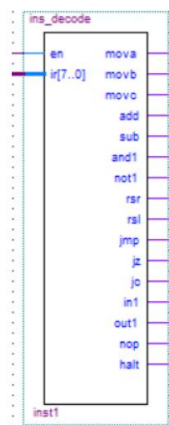
二、实验内容

1. 指令译码器

指令译码器是根据指令系统表中的指令编码，对输入的 8 位指令进行解析，判定是哪条指令，则对应指令的输出为 1，否则输出为 0。

表 1 指令系统表

汇编符号	功能	编码
MOV R1, R2	$(R2) \rightarrow R1$	1100 R1 R2
MOV M, R2	$(R2) \rightarrow (C)$	1100 11 R2
MOV R1, M	$((C)) \rightarrow R1$	1100 R1 11
ADD R1, R2	$(R1) + (R2) \rightarrow R1$	1001 R1 R2
SUB R1, R2	$(R1) - (R2) \rightarrow R1$	0110 R1 R2
AND R1, R2	$(R1) \& (R2) \rightarrow R1$	1011 R1 R2
NOT R1	$\neg (R1) \rightarrow R1$	0101 R1 XX
RSR R1	$(R1)$ 循环右移一位 $\rightarrow R1$	1010 R1 00
RSL R1	$(R1)$ 循环左移一位 $\rightarrow R1$	1010 R1 11
JMP add	$add \rightarrow PC$	0011 00 00, address
JZ add	结果为 0 时 $add \rightarrow PC$	0011 00 01, address
JC add	结果有进位时 $add \rightarrow PC$	0011 00 10, address
IN R1	(开关 7-0) $\rightarrow R1$	0010 R1 XX
OUT R1	$(R1) \rightarrow$ 发光二极管 7-0	0100 R1 XX
NOP	$(PC) + 1 \rightarrow PC$	0111 00 00
HALT	停机	1000 00 00



指令译码器的输入输出引脚如上图所示。en 为使能信号，ir[7..0]是 8 位指令编码，输出是对应的 16 条指令。引脚之间的相互关系如下表所示：

表 2 指令译码器引脚关系

en	ir[7..0]	16 个输出信号
1	8 位的指令编码	指令编码对应的指令输出为 1，其它输出为 0
0	8 位的指令编码	不管 ir 为何值，16 个输出全为 0

2. ALU

算术逻辑运算类指令：

ADD R1, R2

SUB R1, R2

AND R1, R2

NOT R1

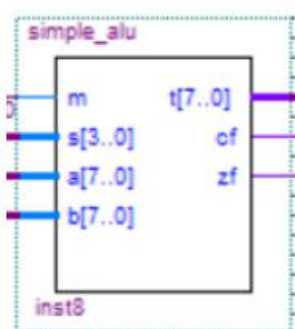
这类指令的执行过程为：

由 R2 的编码通过 RAA1、RAA0 从通用寄存器组 S 口读出 R2 的内容，由 R1 的编码通过 RWBA1、RWBA0 从通用寄存器组 D 口读出 R1 的内容，在 S3~S0 和 M 的控制下，实现运算，经移位逻辑送入总线 BUS；由 /WE 控制和 R1 的编码选择 RWBA1、RWBA0，将 BUS 上的数据写入通用寄存器 R1。其中 ADD 和 SUB 指令影响状态位 Cf 和 Zf。

指令具体功能如下：

汇编符号	功能	编码
ADD R1, R2	$(R1) + (R2) \rightarrow R1$	1001 R1 R2
SUB R1, R2	$(R1) - (R2) \rightarrow R1$	0110 R1 R2
AND R1, R2	$(R1) \& (R2) \rightarrow R1$	1011 R1 R2
NOT R1	$\neg (R1) \rightarrow R1$	0101 R1 XX

ALU 除了要完成 ADD、SUB、AND、NOT 运算外，还需在 MOVA、MOVB、RSR、RSL 和 OUT 五条指令执行时，提供将数据传送至总线的通路。ALU 模块的输入输出引脚如下图所示：



其中 m 和 $s[3..0]$ 是控制信号，控制 $a[7..0]$ 和 $b[7..0]$ 输入的数据进行什么操作，并将产生的结果输出到 $t[7..0]$ 、 cf 和 zf 。各引脚间的相互关系如下表所示：

表 3 ALU 引脚关系

m	$s[3..0]$	$t[7..0]$	cf	zf
1	1001	$t=a+b$	有进位, $cf=1$ 无进位, $cf=0$	和为零, $zf=1$ 和不为零, $zf=0$
1	0110	$t=b-a$	有借位, $cf=1$ 无借位, $cf=0$	差为零, $zf=1$ 差不为零, $zf=0$
1	1011	$t=a\&b$	不影响	不影响
1	0101	$t=b$ (注: b 相反)	不影响	不影响
1	1010 或 0100	$t=b$	不影响	不影响
0	1100	$t=a$	不影响	不影响

三、实验过程

1、指令译码器

A) 创建工程（选择的芯片为 family=Cyclone II; name=EP2C5T144C8）

将工程名命名为 decode，存放在名字为 decode 的文件夹中

B) 编写源代码

```

1 module decode(en,ir,mova,movb,movc,add,sub,andl,notl,rsr,rs1,jmp,jz,jc,inl,outl,nop,halt);
2 input en;
3 input [7:0]ir;
4 output reg mova,movb,movc,add,sub,andl,notl,rsr,rs1,jmp,jz,jc,inl,outl,nop,halt;
5 always@ (en,ir)
6 begin
7 if(en==0)
8 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
9 else if(ir>=8'b1100_0000)
10
11 if(ir[0]==1 && ir[1]==1)
12 begin mova=0;movb=0;movc=1;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
13 else if(ir[2]==1 && ir[3]==1)
14 begin mova=0;movb=1;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
15 else
16 begin mova=1;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
17
18 else if(ir>=8'b1001_0000 && ir<=8'b1001_1111)
19 begin mova=0;movb=0;movc=0;add=1;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
20 else if(ir>=8'b0110_0000 && ir<=8'b0110_1111)
21 begin mova=0;movb=0;movc=0;add=0;sub=1;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
22 else if(ir>=8'b1011_0000 && ir<=8'b1011_1111)
23 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=1;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
24 else if(ir>=8'b0101_0000 && ir<=8'b0101_1111)
25 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=1;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
26 else if(ir>=8'b1010_0000 && ir<=8'b1010_1111)
27 if(ir[0]==0 && ir[1]==0)
28 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=1;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
29 else
30 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=1;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
31 else if(ir==8'b0011_0000)

```

实验一 译码器的实现

```
32 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=1;jz=0;jc=0;inl=0;outl=0;nop=0;halt=0;end
33 else if(ir==8'b0011_0001)
34 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=1;jc=0;inl=0;outl=0;nop=0;halt=0;end
35 else if(ir==8'b0011_0010)
36 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=1;inl=0;outl=0;nop=0;halt=0;end
37 else if(ir>=8'b0010_0000 && ir<=8'b0010_1111)
38 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=1;outl=0;nop=0;halt=0;end
39 else if(ir>=8'b0100_0000 && ir<=8'b0100_1111)
40 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=1;nop=0;halt=0;end
41 else if(ir==8'b0111_0000)
42 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=1;halt=0;end
43 else
44 begin mova=0;movb=0;movc=0;add=0;sub=0;andl=0;notl=0;rsr=0;rs1=0;jmp=0;jz=0;jc=0;inl=0;outl=0;nop=0;halt=1;end
45 end
46 endmodule
```

C) 编译与调试（包含编译调试过程中的错误、警告信息以及资源消耗）

Flow Status	Successful - Fri Dec 03 21:46:48 2021
Quartus II Version	9.0 Build 184 04/29/2009 SP 1 SJ Web Edition
Revision Name	decode
Top-level Entity Name	decode
Family	Stratix II
Met-timing requirements	Yes
Logic utilization	< 1 %
Combinational ALUTs	20 / 12,480 (< 1 %)
Dedicated logic registers	0 / 12,480 (0 %)
Total registers	0
Total pins	25 / 343 (7 %)
Total virtual pins	0
Total block memory bits	0 / 419,328 (0 %)
DSP block 9-bit elements	0 / 96 (0 %)
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 2 (0 %)
Device	EP2K10K10-10
Fitting Models	Final

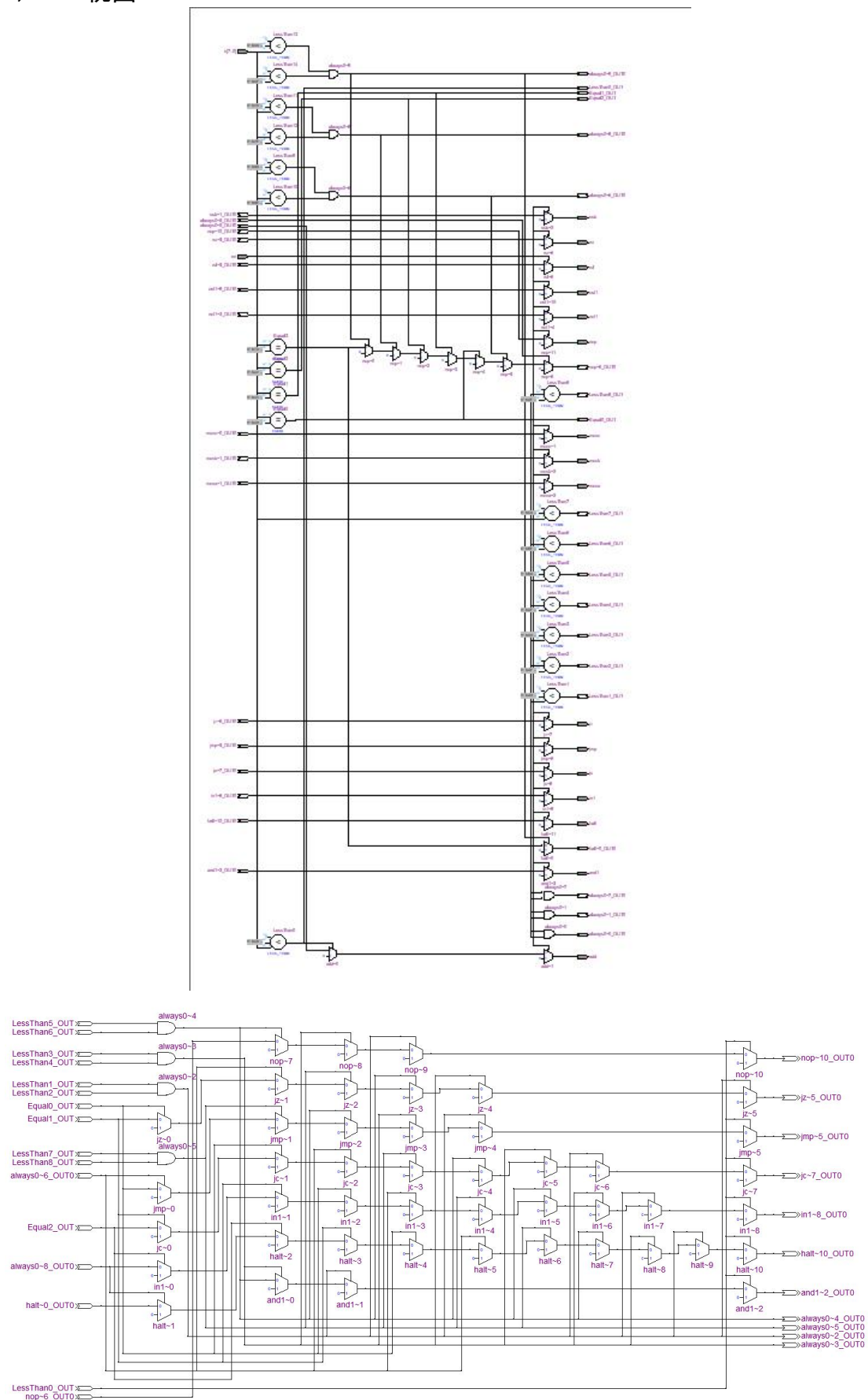
资源消耗

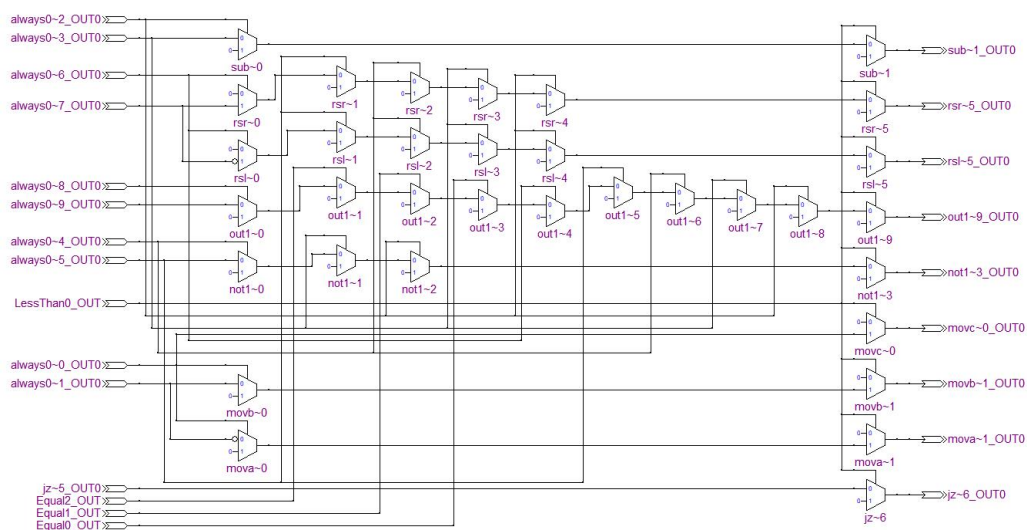
Type	Message
Warning	Warning: Feature LogicLock is not available with your current license
Warning	Warning: No exact pin location assignment(s) for 25 pins of 25 total pins
Warning	Warning: Some pins have incomplete I/O assignments. Refer to the I/O Assignment Warnings report for details
Warning	Warning: Found 16 output pins without output pin load capacitance assignment
Warning	Warning: The Reserve All Unused Pins setting has not been specified, and will default to 'As output driving ground'.

System (28) | Processing (50) | Extra Info | Info (45) | **Warning (5)** | Critical Warning | Error | Suppressed (6) | Flag

警告信息

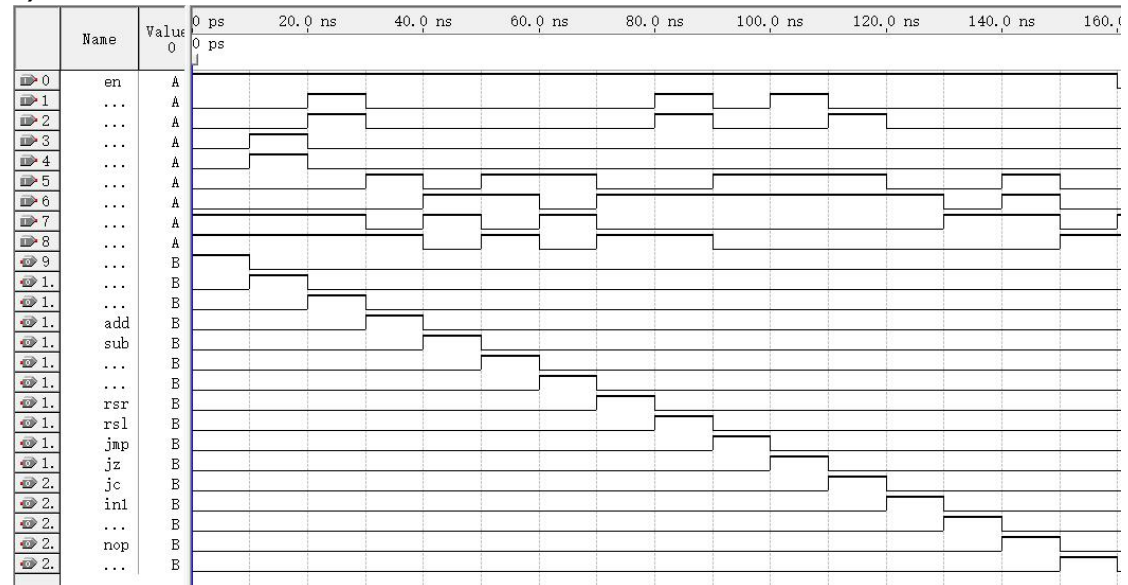
D) RTL 视图



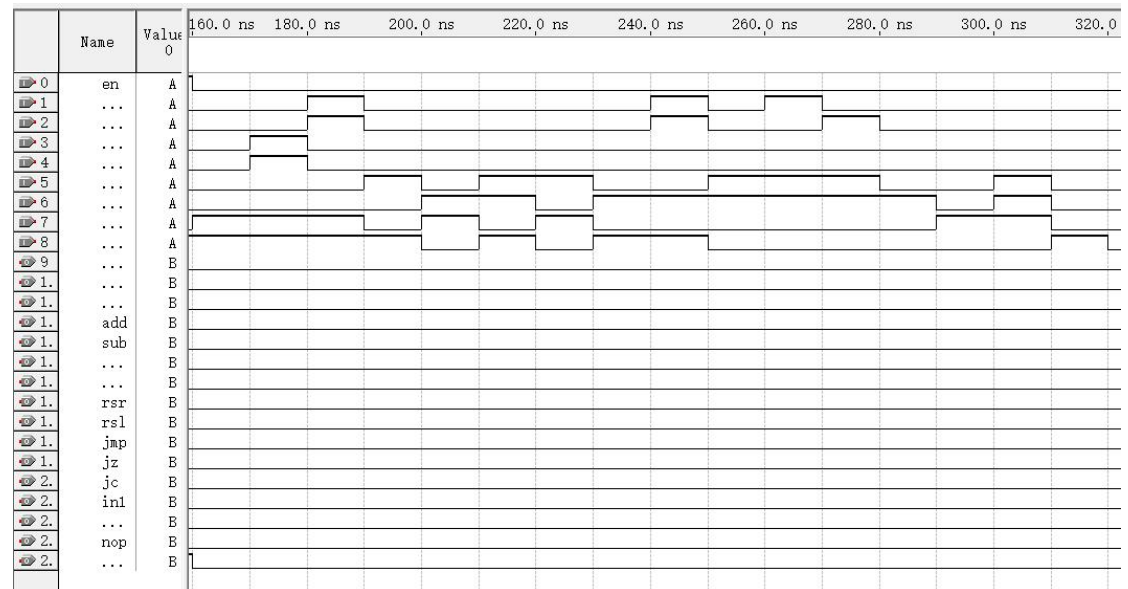


视图分析及结论：由于每一种输入下，所有的输出端口都有确定赋值，故 RTL 视图中没有出现锁存器。从视图中可以看到存在着大量的选择器，这是因为代码中存在大量的条件判断语句，需要对信号进行选择后输出。

E) 功能仿真波形



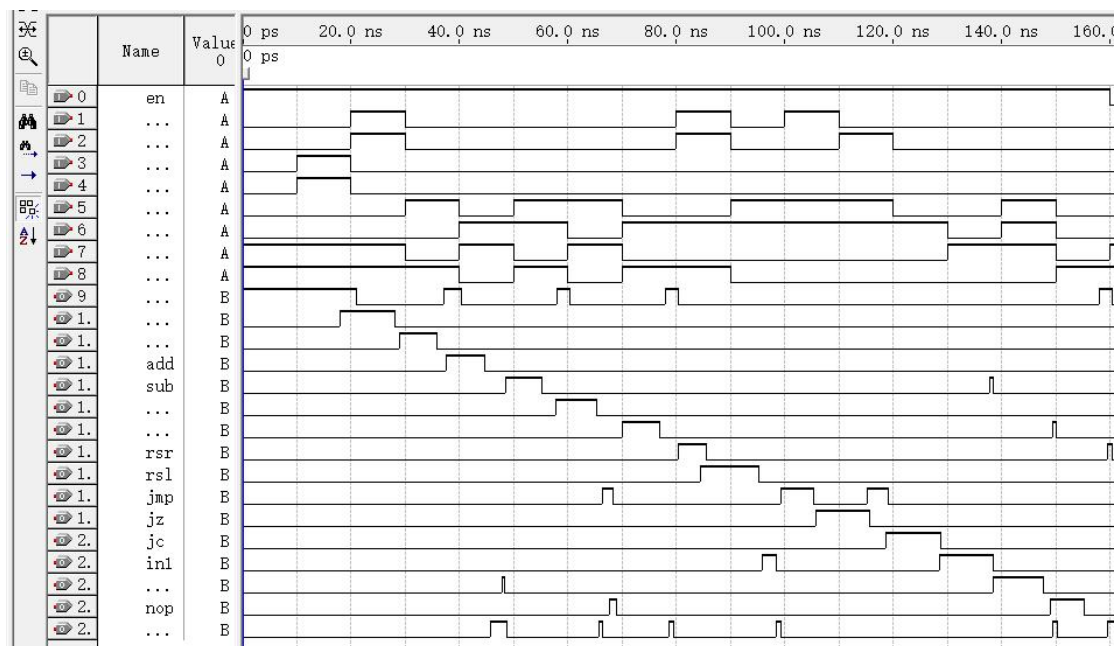
使能信号为 1



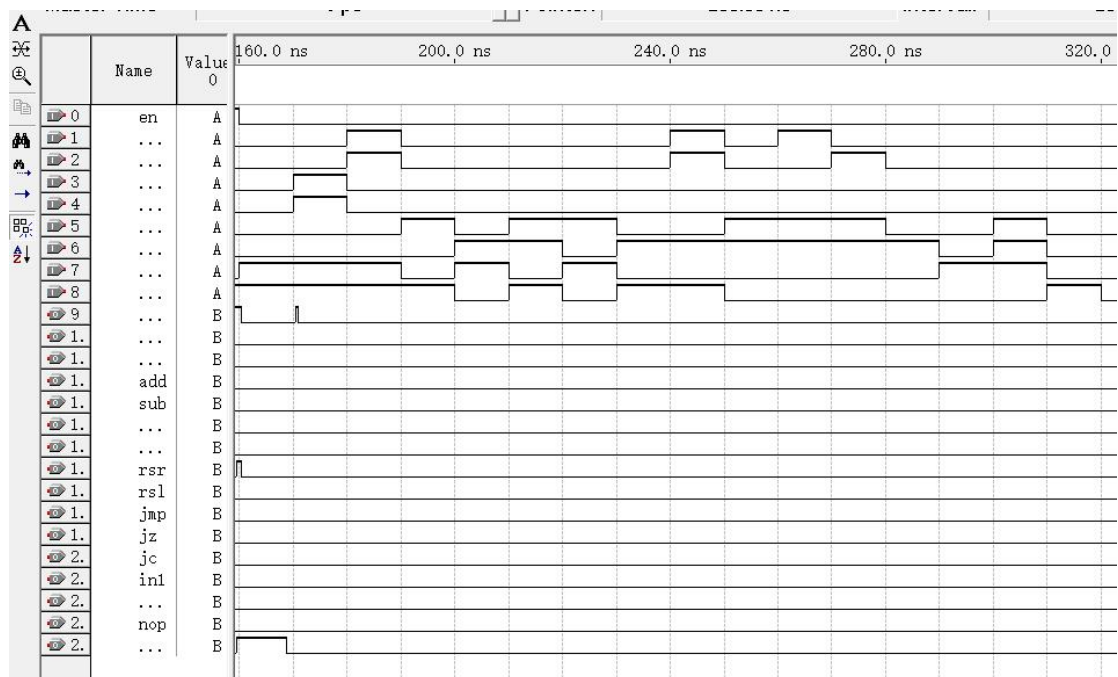
使能信号为 0

结果分析及结论：使能信号为 1 时，将指令表中的输入按顺序输入到电路中，可以观察到输出符合预期。使能信号为 0 时，不论输入组合是什么，输出均全为 0，同样符合预期。故功能仿真符合预期。

F) 时序仿真波形



使能信号为 1



使能信号为 0

结果分析及结论：时序仿真的波形总体趋势上和功能仿真相似。但是时序仿真的输出存在约 10ns 的延时，而且毛刺现象十分明显。这是由于存在延时而引起的竞争与冒险。但是只有当使能为 1 的时候才存在毛刺现象，当使能为 0 时，所有端口的输出为平坦的低电平，不再有毛刺现象。

G) 时序分析

Timing Analyzer Summary

	Type	Slack	Required Time	Actual Time	From	To	From Clock	To Clock	Failed Paths	
1	Worst-case tpd	N/A	None	11.019 ns	en	movb	--	--	0	
2	Total number of failed paths								0	

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
1	N/A	None	11.019 ns	en	movb
2	N/A	None	10.894 ns	ir[3]	movb
3	N/A	None	10.434 ns	ir[2]	movb
4	N/A	None	10.407 ns	ir[6]	movb
5	N/A	None	10.402 ns	ir[5]	rsr
6	N/A	None	10.323 ns	ir[1]	movb
7	N/A	None	10.200 ns	ir[5]	halt
8	N/A	None	10.128 ns	ir[5]	not1
9	N/A	None	10.124 ns	ir[3]	halt
10	N/A	None	10.082 ns	ir[5]	jmp
11	N/A	None	10.006 ns	ir[3]	jmp
12	N/A	None	9.929 ns	en	movc
13	N/A	None	9.912 ns	en	not1
14	N/A	None	9.834 ns	ir[5]	jc
15	N/A	None	9.758 ns	ir[3]	jc
16	N/A	None	9.664 ns	ir[2]	halt
17	N/A	None	9.662 ns	ir[5]	jz
18	N/A	None	9.586 ns	ir[3]	jz
19	N/A	None	9.546 ns	ir[2]	jmp
20	N/A	None	9.545 ns	ir[6]	rsr
21	N/A	None	9.496 ns	ir[4]	not1
22	N/A	None	9.482 ns	ir[6]	not1
23	N/A	None	9.429 ns	ir[4]	rsr
24	N/A	None	9.343 ns	ir[4]	halt
25	N/A	None	9.317 ns	ir[6]	movc
26	N/A	None	9.317 ns	en	movb
27	N/A	None	9.298 ns	ir[2]	jc
28	N/A	None	9.280 ns	ir[1]	halt
29	N/A	None	9.225 ns	ir[4]	jmp
30	N/A	None	9.218 ns	ir[5]	rs1
31	N/A	None	9.214 ns	ir[6]	halt
32	N/A	None	9.196 ns	en	rsr
33	N/A	None	9.126 ns	ir[2]	jz
34	N/A	None	9.116 ns	ir[5]	in1

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
35	N/A	None	9.039 ns	ir[1]	jmp
36	N/A	None	9.009 ns	ir[5]	nop
37	N/A	None	8.988 ns	en	halt
38	N/A	None	8.988 ns	ir[1]	rsr
39	N/A	None	8.977 ns	ir[4]	jc
40	N/A	None	8.961 ns	ir[1]	movc
41	N/A	None	8.958 ns	en	jz
42	N/A	None	8.956 ns	en	jmp
43	N/A	None	8.933 ns	ir[3]	nop
44	N/A	None	8.922 ns	en	jc
45	N/A	None	8.900 ns	en	in1
46	N/A	None	8.805 ns	ir[4]	jz
47	N/A	None	8.758 ns	ir[1]	jz
48	N/A	None	8.705 ns	ir[6]	movb
49	N/A	None	8.687 ns	ir[1]	jc
50	N/A	None	8.684 ns	ir[6]	jz
51	N/A	None	8.648 ns	ir[6]	jc
52	N/A	None	8.497 ns	ir[6]	in1
53	N/A	None	8.484 ns	ir[5]	and1
54	N/A	None	8.473 ns	ir[2]	nop
55	N/A	None	8.472 ns	ir[4]	in1
56	N/A	None	8.443 ns	ir[5]	sub
57	N/A	None	8.404 ns	ir[5]	out1
58	N/A	None	8.384 ns	ir[1]	movb
59	N/A	None	8.361 ns	ir[6]	rs1
60	N/A	None	8.334 ns	ir[5]	add
61	N/A	None	8.298 ns	ir[6]	jmp
62	N/A	None	8.269 ns	en	and1
63	N/A	None	8.245 ns	ir[4]	rs1
64	N/A	None	8.226 ns	en	sub
65	N/A	None	8.187 ns	en	out1
66	N/A	None	8.152 ns	ir[4]	nop
67	N/A	None	8.123 ns	ir[1]	rs1
68	N/A	None	8.040 ns	en	rs1

tpd									
	Slack	Required P2P Time	Actual P2P Time	From	To				
69	N/A	None	8.013 ns	ir[7]	movb				
70	N/A	None	7.992 ns	ir[3]	movb				
71	N/A	None	7.989 ns	ir[2]	movb				
72	N/A	None	7.869 ns	ir[6]	andi				
73	N/A	None	7.854 ns	ir[4]	andi				
74	N/A	None	7.833 ns	ir[4]	sub				
75	N/A	None	7.794 ns	ir[4]	out1				
76	N/A	None	7.760 ns	ir[1]	nop				
77	N/A	None	7.751 ns	ir[6]	sub				
78	N/A	None	7.715 ns	ir[6]	out1				
79	N/A	None	7.712 ns	ir[6]	nop				
80	N/A	None	7.606 ns	en	nop				
81	N/A	None	7.477 ns	ir[6]	add				
82	N/A	None	7.376 ns	en	add				
83	N/A	None	7.327 ns	ir[4]	add				
84	N/A	None	7.092 ns	ir[0]	movb				
85	N/A	None	6.923 ns	ir[7]	movc				
86	N/A	None	6.917 ns	ir[7]	not1				
87	N/A	None	6.477 ns	ir[7]	rsr				
88	N/A	None	6.426 ns	ir[7]	halt				
89	N/A	None	6.411 ns	ir[0]	halt				
90	N/A	None	6.311 ns	ir[7]	movb				
91	N/A	None	6.308 ns	ir[7]	jmp				
92	N/A	None	6.060 ns	ir[7]	jc				
93	N/A	None	5.894 ns	ir[7]	in1				
94	N/A	None	5.888 ns	ir[7]	jz				
95	N/A	None	5.756 ns	ir[0]	movc				
96	N/A	None	5.652 ns	ir[0]	jz				
97	N/A	None	5.517 ns	ir[0]	rsr	103	N/A	None	5.220 ns
98	N/A	None	5.472 ns	ir[0]	jc	104	N/A	None	5.181 ns
99	N/A	None	5.301 ns	ir[0]	jmp	105	N/A	None	5.144 ns
100	N/A	None	5.293 ns	ir[7]	rsl	106	N/A	None	4.909 ns
101	N/A	None	5.274 ns	ir[7]	andi	107	N/A	None	4.616 ns
102	N/A	None	5.235 ns	ir[7]	nop	108	N/A	None	4.386 ns

结果分析及结论：

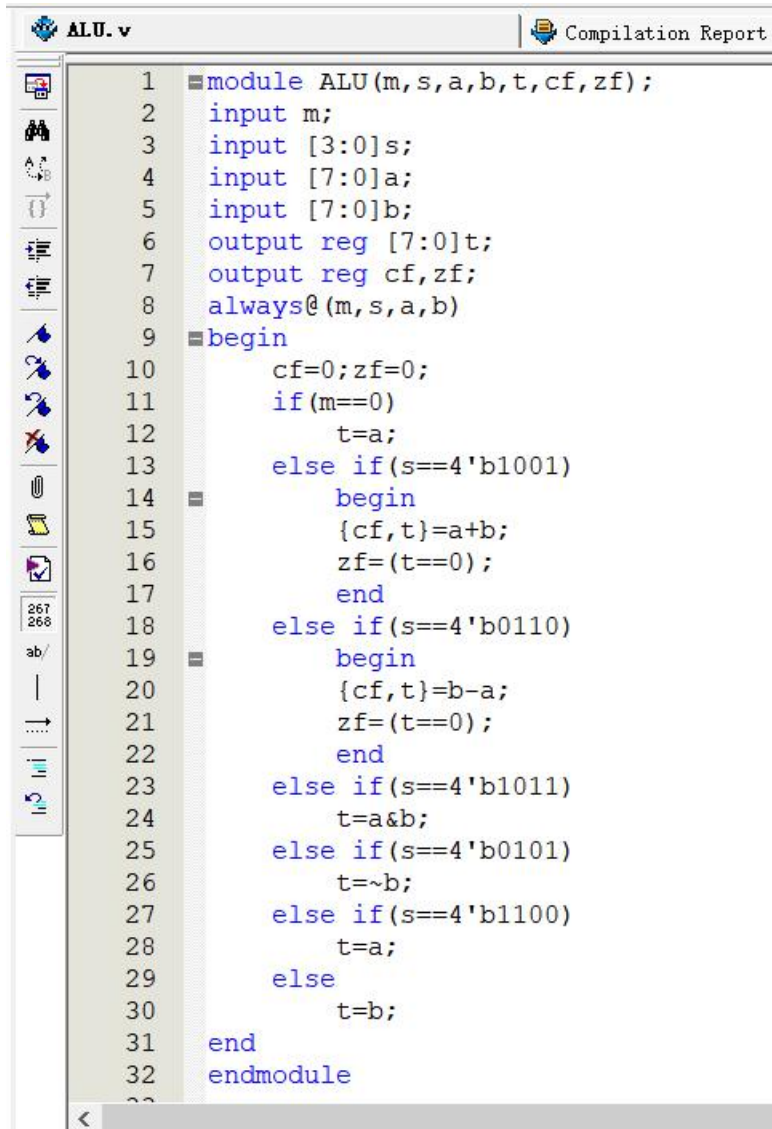
延时范围为 4.386ns-11.109ns，且集中在 7ns-10ns 之间，与时序仿真波形图观察到的结果基本符合。不同的输出端口的延时存在的差异，在时序仿真波形图中各个输出端口输出高电平的时长也存在差异。

2、算术逻辑单元 ALU

A) 创建工程（选择的芯片为 family=FLEX10K；name=EPF10K20T1144-4）

将工程名命名为 ALU，存放在名字为 ALU 的文件夹中






B) 编写源代码



```
1 module ALU(m,s,a,b,t,cf,zf);
2   input m;
3   input [3:0]s;
4   input [7:0]a;
5   input [7:0]b;
6   output reg [7:0]t;
7   output reg cf,zf;
8   always@(m,s,a,b)
9   begin
10      cf=0;zf=0;
11      if(m==0)
12          t=a;
13      else if(s==4'b1001)
14          begin
15              {cf,t}=a+b;
16              zf=(t==0);
17          end
18      else if(s==4'b0110)
19          begin
20              {cf,t}=b-a;
21              zf=(t==0);
22          end
23      else if(s==4'b1011)
24          t=a&b;
25      else if(s==4'b0101)
26          t=~b;
27      else if(s==4'b1100)
28          t=a;
29      else
30          t=b;
31  end
32 endmodule
```

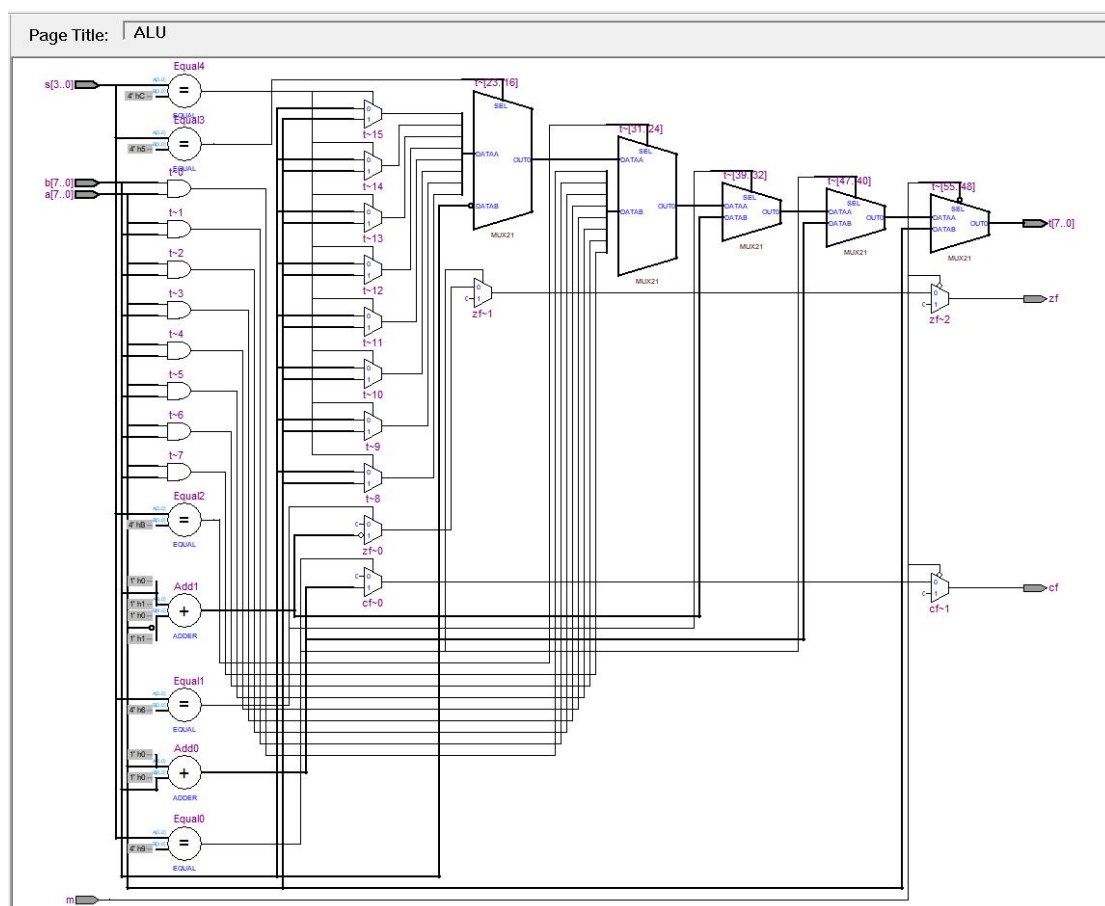
C) 编译与调试（包含编译调试过程中的错误、警告信息以及资源消耗）

```
Flow Status                Successful - Fri Dec 03 22:56:29 2021
Quartus II Version         9.0 Build 184 04/29/2009 SP 1 SJ Web Edition
Revision Name              ALU
Top-level Entity Name      ALU
Family                     Stratix II
Met timing requirements     Yes
Logic utilization          < 1 %
    Combinational ALUTs    47 / 12,480 ( < 1 % )
    Dedicated logic registers 0 / 12,480 ( 0 % )
Total registers            0
Total pins                 31 / 343 ( 9 % )
Total virtual pins         0
Total block memory bits    0 / 419,328 ( 0 % )
DSP block 9-bit elements   0 / 96 ( 0 % )
Total PLLs                 0 / 6 ( 0 % )
Total DLLs                 0 / 2 ( 0 % )
Device                     EP2S15F484C3
Timing Models              Final
```

Type	Message
	Warning: Feature LogicLock is not available with your current license
	Warning: No exact pin location assignment(s) for 31 pins of 31 total pins
	Warning: Some pins have incomplete I/O assignments. Refer to the I/O Assignment Warnings report for details
	Warning: Found 10 output pins without output pin load capacitance assignment
	Warning: The Reserve All Unused Pins setting has not been specified, and will default to 'As output driving ground'.

System (14) Processing (50) Extra Info Info (45) **Warning (5)** Critical Warning Error Suppressed (6) Flag

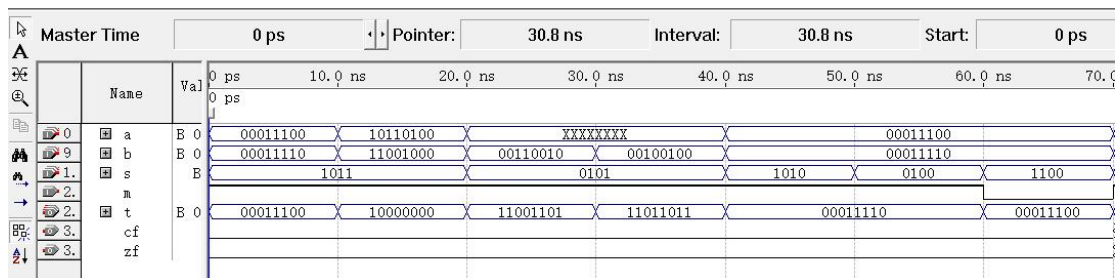
D) RTL 视图



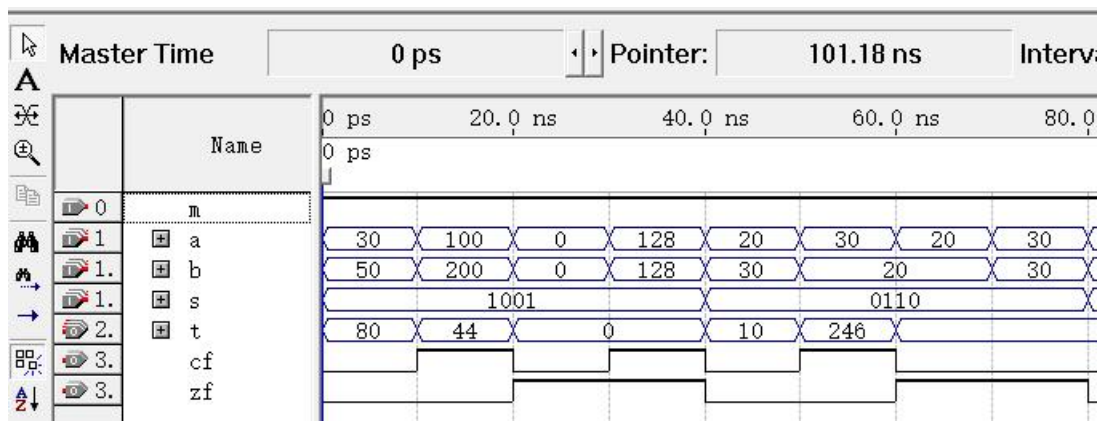
结果分析及结论：

除了 ADD 和 SUB 操作要对 cf 和 zf 进行赋值外，其余操作也要对 cf 和 zf 进行赋值，以确保组合电路中没有锁存器。观察 RTL 视图可知，没有存在锁存器，故这是符合要求的。电路中用八个与门实现了与运算，加法器实现了加运算，通过选择器选择对应的结果输出。

E) 功能仿真波形



与运算、取反、传输通道的仿真波形

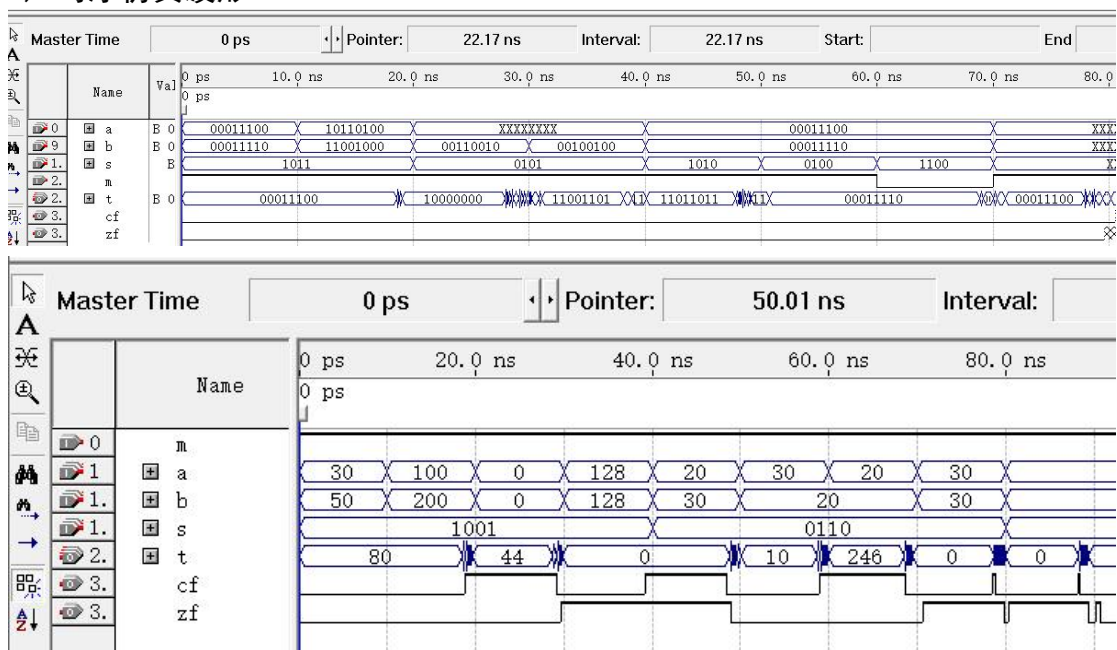


加法、减法的仿真波形

结果分析及结论：

当 cf 和 zf 没有输出时，为了避免产生锁存器，默认 cf 和 zf 输出 0。由于存储 cf 和 zf 的寄存器使能信号处于非使能状态，故不用担心输出的 cf 和 zf 被写入进寄存器。输入 a 是源寄存器口，b 是目的寄存器口，执行取反操作时，目的寄存器口输入的数据有效，即对 b 取反。进行减法操作是目的寄存器的值减去源寄存器的值，即 $b-a$ 。

F) 时序仿真波形



结果分析及结论：

时序仿真输出的延时约 10ns。可以从波形中看到，在输入改变的时候，输出会存在一段噪声，随后输出正确的值。不同的操作对应的噪声时长也不同。但总体而言，输出的值只有部分时长被噪声掩盖，正确的输出仍占主导地位。

G) 时序分析

Timing Analyzer Summary									
	Type	Slack	Required Time	Actual Time	From	To	From Clock	To Clock	Failed Paths
1	Worst-case tpd	N/A	None	11.989 ns	b[6]	t[6]	--	--	0
2	Total number of failed paths								0

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
1	N/A	None	11.989 ns	b[6]	t[6]
2	N/A	None	11.833 ns	m	t[6]
3	N/A	None	11.825 ns	s[3]	t[6]
4	N/A	None	11.764 ns	s[1]	t[6]
5	N/A	None	11.600 ns	b[6]	cf
6	N/A	None	11.564 ns	b[4]	t[6]
7	N/A	None	11.561 ns	s[0]	t[6]
8	N/A	None	11.287 ns	b[3]	t[6]
9	N/A	None	11.194 ns	b[1]	t[6]
10	N/A	None	11.162 ns	s[3]	t[1]
11	N/A	None	11.144 ns	b[3]	cf
12	N/A	None	11.132 ns	b[2]	t[6]
13	N/A	None	11.123 ns	a[0]	t[6]
14	N/A	None	11.107 ns	b[4]	cf
15	N/A	None	11.100 ns	s[1]	t[1]
16	N/A	None	11.092 ns	b[6]	zf
17	N/A	None	11.063 ns	a[5]	t[6]
18	N/A	None	11.052 ns	m	t[1]
19	N/A	None	11.042 ns	a[1]	t[6]
20	N/A	None	11.039 ns	b[0]	t[6]
21	N/A	None	10.996 ns	a[3]	t[6]
22	N/A	None	10.989 ns	b[2]	cf
23	N/A	None	10.939 ns	a[4]	t[6]
24	N/A	None	10.924 ns	s[0]	t[1]
25	N/A	None	10.902 ns	b[5]	t[6]
26	N/A	None	10.874 ns	a[2]	t[6]
27	N/A	None	10.857 ns	b[1]	cf
28	N/A	None	10.857 ns	s[2]	t[6]
29	N/A	None	10.808 ns	a[3]	cf
30	N/A	None	10.796 ns	a[4]	cf
31	N/A	None	10.763 ns	b[7]	cf
32	N/A	None	10.724 ns	m	t[0]
33	N/A	None	10.716 ns	s[3]	t[0]
34	N/A	None	10.676 ns	a[5]	cf

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
35	N/A	None	10.655 ns	s[1]	t[0]
36	N/A	None	10.653 ns	b[4]	zf
37	N/A	None	10.644 ns	a[2]	cf
38	N/A	None	10.604 ns	a[6]	t[6]
39	N/A	None	10.592 ns	a[0]	cf
40	N/A	None	10.542 ns	b[5]	cf
41	N/A	None	10.539 ns	a[7]	cf
42	N/A	None	10.489 ns	a[6]	cf
43	N/A	None	10.457 ns	a[1]	cf
44	N/A	None	10.452 ns	s[0]	t[0]
45	N/A	None	10.443 ns	a[0]	t[0]
46	N/A	None	10.425 ns	b[4]	t[4]
47	N/A	None	10.406 ns	b[0]	cf
48	N/A	None	10.392 ns	s[3]	cf
49	N/A	None	10.378 ns	s[3]	t[4]
50	N/A	None	10.332 ns	a[7]	zf
51	N/A	None	10.326 ns	s[3]	t[3]
52	N/A	None	10.324 ns	m	t[3]
53	N/A	None	10.319 ns	m	t[4]
54	N/A	None	10.302 ns	s[1]	cf
55	N/A	None	10.284 ns	s[1]	t[4]
56	N/A	None	10.283 ns	b[1]	zf
57	N/A	None	10.264 ns	b[0]	t[0]
58	N/A	None	10.258 ns	b[1]	t[1]
59	N/A	None	10.255 ns	s[1]	t[3]
60	N/A	None	10.253 ns	a[0]	t[1]
61	N/A	None	10.250 ns	b[3]	t[4]
62	N/A	None	10.212 ns	a[0]	zf
63	N/A	None	10.207 ns	b[3]	zf
64	N/A	None	10.206 ns	s[2]	t[1]
65	N/A	None	10.169 ns	b[0]	t[1]
66	N/A	None	10.146 ns	b[6]	t[7]
67	N/A	None	10.132 ns	a[5]	zf
68	N/A	None	10.131 ns	a[1]	zf

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
69	N/A	None	10.131 ns	m	t[2]
70	N/A	None	10.128 ns	b[0]	zf
71	N/A	None	10.127 ns	b[7]	zf
72	N/A	None	10.123 ns	s[3]	t[2]
73	N/A	None	10.112 ns	s[0]	cf
74	N/A	None	10.097 ns	b[2]	zf
75	N/A	None	10.095 ns	b[2]	t[4]
76	N/A	None	10.087 ns	a[1]	t[1]
77	N/A	None	10.085 ns	a[3]	zf
78	N/A	None	10.081 ns	s[0]	t[4]
79	N/A	None	10.069 ns	b[1]	t[4]
80	N/A	None	10.062 ns	s[1]	t[2]
81	N/A	None	10.052 ns	s[0]	t[3]
82	N/A	None	9.998 ns	a[0]	t[4]
83	N/A	None	9.977 ns	b[5]	zf
84	N/A	None	9.963 ns	a[2]	zf
85	N/A	None	9.951 ns	b[1]	t[3]
86	N/A	None	9.917 ns	a[1]	t[4]
87	N/A	None	9.914 ns	b[0]	t[4]
88	N/A	None	9.914 ns	a[3]	t[4]
89	N/A	None	9.888 ns	a[4]	t[4]
90	N/A	None	9.880 ns	a[0]	t[3]
91	N/A	None	9.859 ns	b[3]	t[3]
92	N/A	None	9.859 ns	s[0]	t[2]
93	N/A	None	9.837 ns	s[3]	t[7]
94	N/A	None	9.803 ns	m	t[7]
95	N/A	None	9.799 ns	a[1]	t[3]
96	N/A	None	9.796 ns	b[0]	t[3]
97	N/A	None	9.779 ns	b[2]	t[3]
98	N/A	None	9.776 ns	a[4]	zf
99	N/A	None	9.750 ns	a[2]	t[4]
100	N/A	None	9.748 ns	s[2]	t[0]
101	N/A	None	9.734 ns	s[1]	t[7]
102	N/A	None	9.731 ns	a[3]	t[3]

tpd					
	Slack	Required P2P Time	Actual P2P Time	From	To
103	N/A	None	9.703 ns	m	cf
104	N/A	None	9.685 ns	b[4]	t[7]
105	N/A	None	9.653 ns	a[2]	t[3]
106	N/A	None	9.612 ns	a[6]	zf
107	N/A	None	9.586 ns	s[3]	zf
108	N/A	None	9.572 ns	s[0]	t[7]
109	N/A	None	9.570 ns	s[3]	t[5]
110	N/A	None	9.536 ns	m	t[5]
111	N/A	None	9.513 ns	b[3]	t[7]
112	N/A	None	9.467 ns	s[1]	t[5]
113	N/A	None	9.392 ns	s[2]	cf
114	N/A	None	9.376 ns	s[2]	t[4]
115	N/A	None	9.365 ns	a[7]	t[7]
116	N/A	None	9.358 ns	b[2]	t[7]
117	N/A	None	9.348 ns	s[2]	t[3]
118	N/A	None	9.337 ns	b[1]	t[2]
119	N/A	None	9.319 ns	m	zf
120	N/A	None	9.315 ns	b[1]	t[7]
121	N/A	None	9.305 ns	s[0]	t[5]
122	N/A	None	9.266 ns	s[1]	zf
123	N/A	None	9.247 ns	b[4]	t[5]
124	N/A	None	9.246 ns	a[0]	t[2]
125	N/A	None	9.244 ns	a[0]	t[7]
126	N/A	None	9.179 ns	a[1]	t[2]
127	N/A	None	9.177 ns	a[3]	t[7]
128	N/A	None	9.171 ns	s[0]	zf
129	N/A	None	9.165 ns	a[4]	t[7]
130	N/A	None	9.164 ns	a[5]	t[7]
131	N/A	None	9.163 ns	a[1]	t[7]
132	N/A	None	9.162 ns	b[0]	t[2]
133	N/A	None	9.160 ns	b[0]	t[7]
134	N/A	None	9.155 ns	s[2]	t[2]
135	N/A	None	9.145 ns	b[7]	t[7]
136	N/A	None	9.096 ns	b[2]	t[2]

137	N/A	None	9.013 ns	a[2]	t[7]
138	N/A	None	9.009 ns	b[5]	t[7]
139	N/A	None	8.983 ns	a[2]	t[2]
140	N/A	None	8.898 ns	b[3]	t[5]
141	N/A	None	8.867 ns	s[2]	t[7]
142	N/A	None	8.858 ns	a[6]	t[7]
143	N/A	None	8.855 ns	b[1]	t[5]
144	N/A	None	8.784 ns	a[0]	t[5]
145	N/A	None	8.743 ns	b[2]	t[5]

146	N/A	None	8.703 ns	a[1]	t[5]
147	N/A	None	8.700 ns	b[0]	t[5]
148	N/A	None	8.688 ns	a[5]	t[5]
149	N/A	None	8.657 ns	a[3]	t[5]
150	N/A	None	8.600 ns	s[2]	t[5]
151	N/A	None	8.550 ns	a[4]	t[5]
152	N/A	None	8.535 ns	a[2]	t[5]
153	N/A	None	8.514 ns	b[5]	t[5]
154	N/A	None	8.445 ns	s[2]	zf

结果分析及结论：延时非常集中，绝大部分都分布在 9-11ns 之间。延时分布集中有利于减轻冒险和竞争导致的毛刺时长，让有效的输出占比更大。

四、思考题

1. 指令译码器必须要 16 个输出吗？可否将一些输出合并，哪些可以合并，为什么？

控制信号对应的指令编码如下：

LD PC=JMP+JZ·ZF+JC·CF

IN PC=SM'+JZ·ZF'+JC·CF'

MADD0=MOVC

MADD1=MOVB

DL=SM'+MOVC+JMP+JZ·ZF+JC·CF

XL=MOVB

LD IR=SM'

/WE=(MOVA+MOVC+ADD+SUB+AND+NOT+IN+RSR+RSL)'

M= ADD+SUB+AND+NOT+RSR+RSL

F->BUS=MOVA+MOVB+ADD+SUB+AND+NOT+OUT

FL->BUS=RSL

FR->BUS=RSR

CF_EN=ADD+SUB+RSR+RSL

ZF_EN=ADD+SUB

可以观察到，ADD 和 SUB 总是同时出现，AND 和 NOT 总是同时出现，故可以将他们看做两个整体，也就是将四个输出合并成两个输出。

2 . ALU 中的 S[3..0]控制信号是来自哪里或者说与什么信息相同？

ALU 的 S[3..0]来自指令的前 4 位，与指令寄存器存储的指令前 4 位相同。

3、为何 S[3..0]等于 1100 时将输入 a 传给 t，S[3..0]等于 1010 或 0100 时将输入 b 传给 t？

a 是源寄存器口输入的数据，b 是目的寄存器口输入的数据。

S[3..0]=1100 时,执行的是 MOV 类操作，属于传输类指令，数据从源寄存器口输入 ALU，经过传输后送至目的端。属于源端和目的端不同的指令，故将输入 a 传给 t。

S[3..0]=1010 时，执行的是移位操作，S[3..0]=0100 时，执行的是 OUT 操作。其中移位操作的源端和目的端为同一个寄存器，故传输目的寄存器口输入的数据。OUT 操作则只负责将输入送至总线，由输出设备接收后输出，ALU 也只传输目的寄存器口输入的数据。

也可以从编码的角度来看待这个问题，1100 的编码中，处于源端的地址（倒数第 1,2 位）是有效的，故 ALU 传输源端，即 a 的数据。1010 和 0100 的源端不是有效地址，只有目的地的地址有效（倒数第 3,4 位），故 ALU 传输目的端，即 b 的数据。

五、实验总结、必得体会及建议

从需要掌握的理论、遇到的困难、解决的办法以及经验教训等方面进行总结。

需要掌握的理论：

模型机的内部结构和指令的执行过程

指令译码器的原理

ALU 的实现方法

遇到的困难：

Verilog 仿真操作遇到困难

解决的办法：

仔细阅读 quartus-II 使用手册，上网查阅相关资料。

经验教训：

边做实验边记录要点，将软件如何使用的、原理如何分析的、困难如何解决的逐一记录下来，当下次遇到类似的困难时，及时查阅之前的记录，有助于快速解决问题，达到事半功倍的效果。