仿真说明

目录

[仿真说明 1](#_Toc122706147)

[一、 Cpuclk 2](#_Toc122706148)

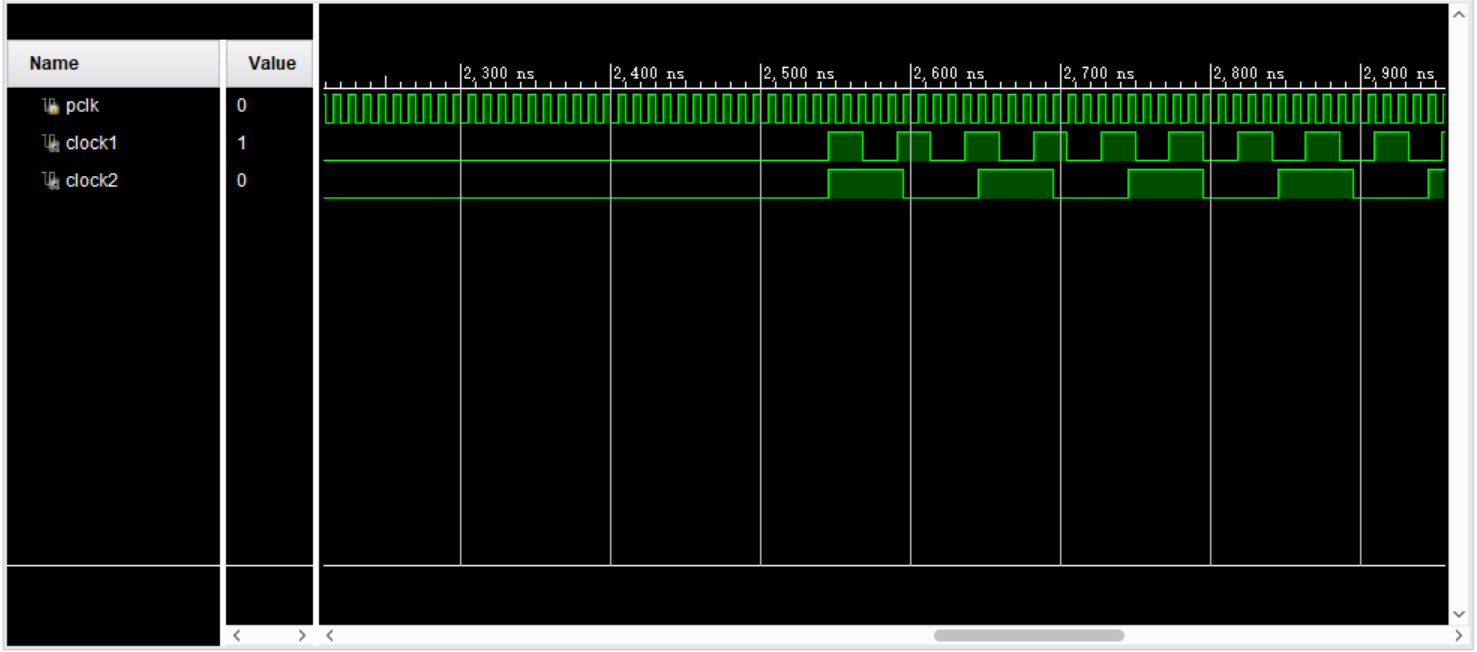
[二、 Ifetch32 2](#_Toc122706149)

[三、 Idecode32 3](#_Toc122706150)

[四、 Exe 4](#_Toc122706151)

[五、 Memory 5](#_Toc122706152)

1. Cpuclk
   1. 截图

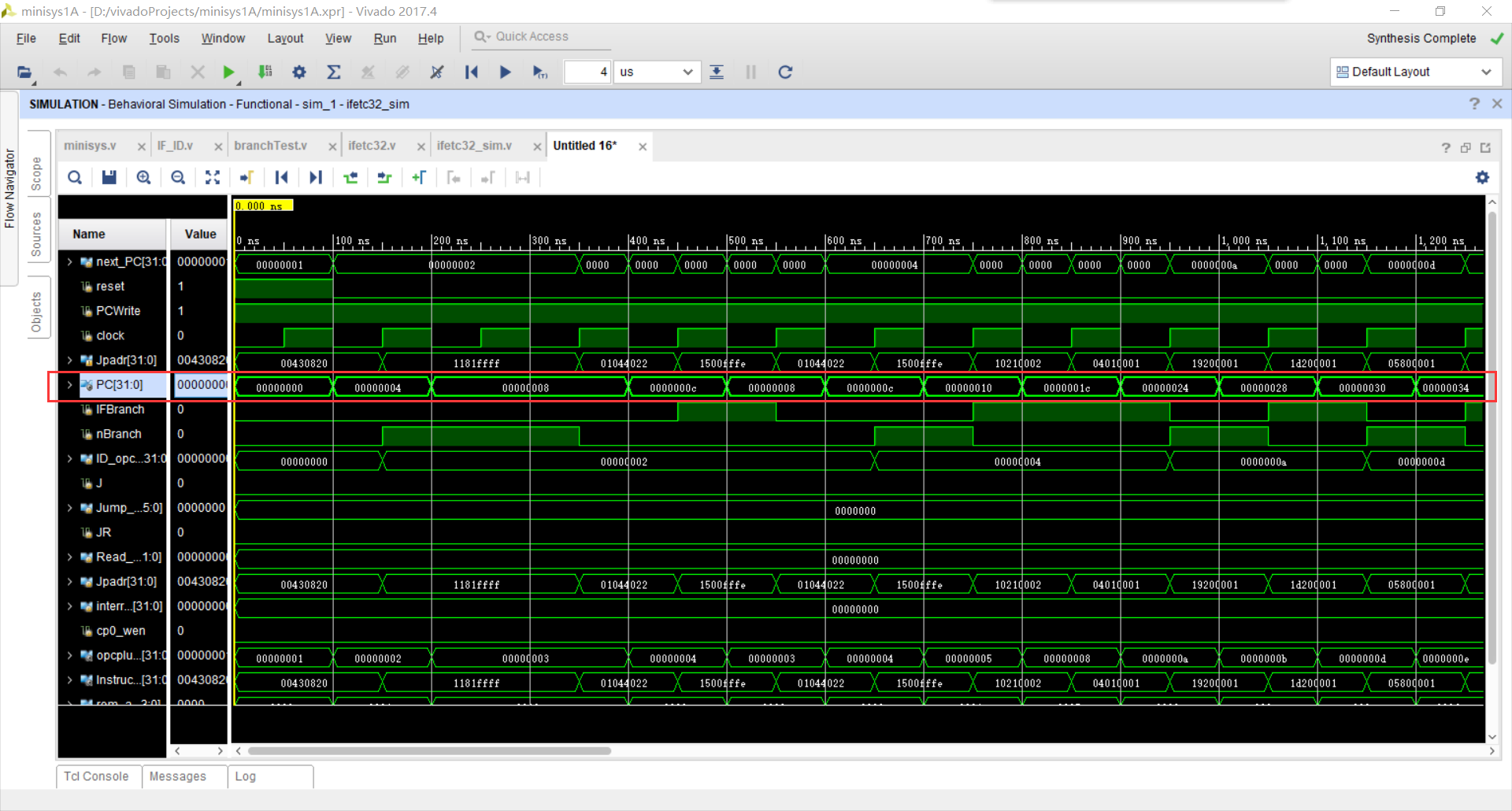


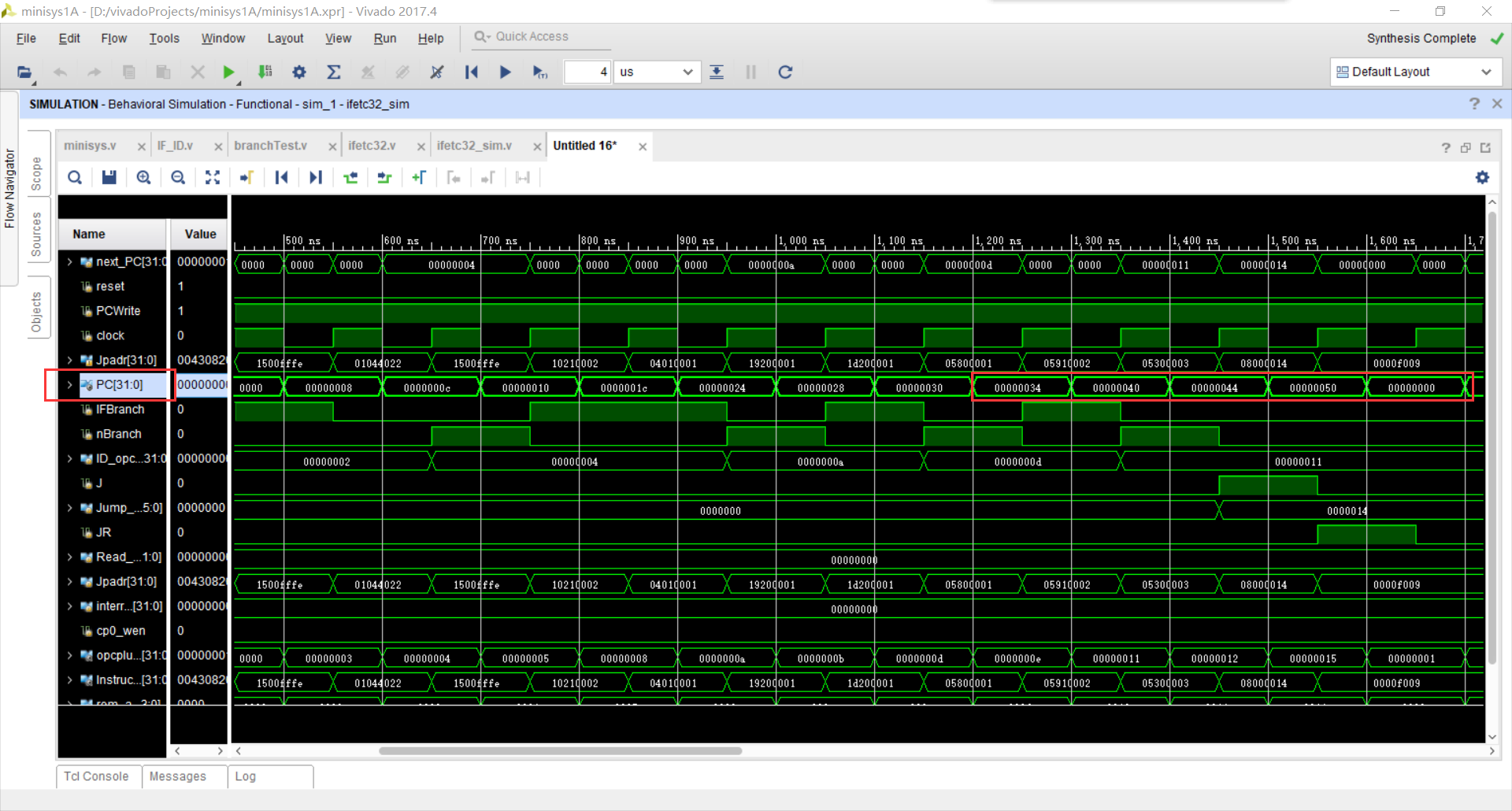
1. Ifetch32

2.1 Ifetch32-sim.asm

|  |
| --- |
| .DATA 0X0  .TEXT 0X0000  start:  add $1,$2,$3 # 00:$1=$2+$3=5  lop:  beq $12,$1,lop # 04:if $12=$1,jump lop,不分支  lop1:  sub $8,$8,$4 # 08:$8=$8-4  bne $8,$0,lop1 # 0c:if $8≠0,jump lop1  beq $1,$1,lop2 # 10:if $1=$1,jump lop2  nop # 14  nop # 18  lop2:  bgez $0,1 # 1c:if $zero≥0,jump PC+4+1<<2=24  jr $0 # 20:  blez $9,1 # 24:if $9<=0,jump PC+4+1<<2, 不分支  bgtz $9,1 # 28:if $9>0,jump PC+4+1<<2=30  j first # 2c:  bltz $12,1 # 30:if $12<0,jump PC+4+1<<2,不分支  bgezal $12,2 # 34:if $12≥0,jump PC+4+2<<2,PC=40  jal start # 38  nop # 3c  bltzal $9,3 # 40:if $9<0,jump PC+4+3<<2,不分支  j lop3 # 44:PC=50  nop # 48  nop # 4c  lop3:  jalr $30,$0 # 50:PC=0  first:  add $1,$3,$4 # 54:$1=$3+$4=7  #PC:  #00, 04, 08,0c,08,0c, 10,1c,24, 28,30, 34,40, 44, 50, 0  #00,00,04,04,08,0c,08,0c,08,10,1c,24,2c,28,30,38,34,40,50,44,48,50,54,0 |

2.2 截图



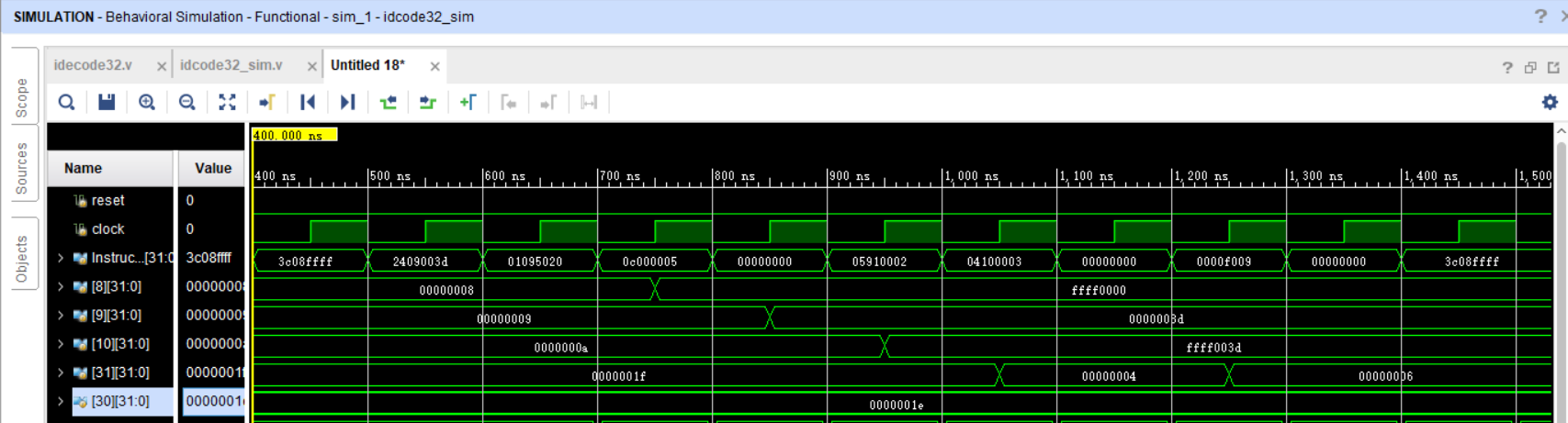


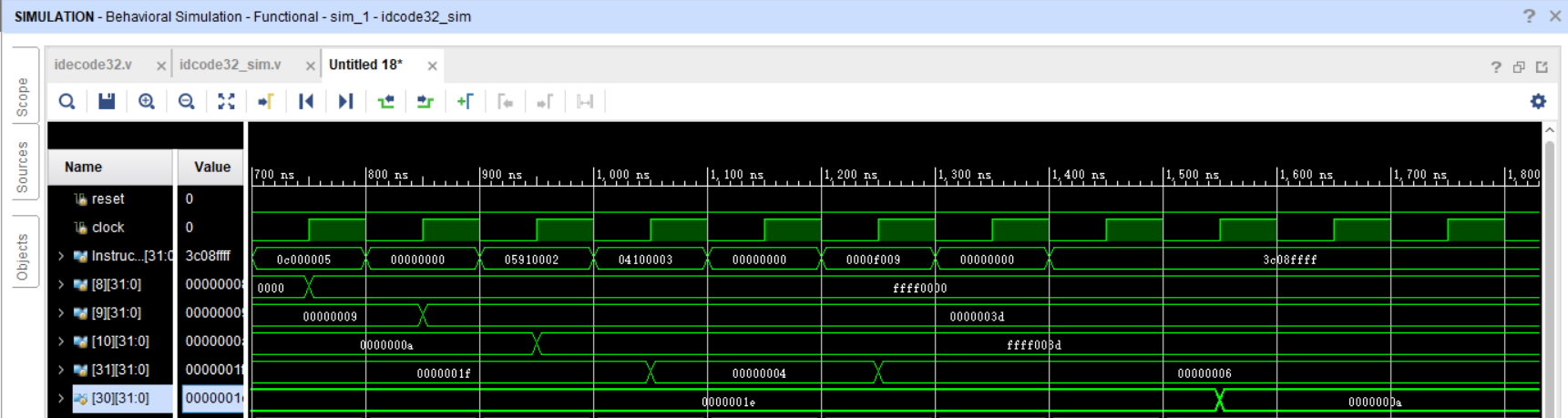
1. Idecode32

3.1 Idecode-sim.asm

|  |
| --- |
| .DATA 0X0  .TEXT 0X0000  start:  lui $8,0xffff # 00:$8=0xffff0000  addiu $9,$0,61 # 04:$9=61=111101=0x3d  add $10,$8,$9 # 08:$10=$8+$9=0xffff003d(负数)  jal lop1 # 0c:$31=(0c+4)>>2=10>>2=16>>2=4,PC=14  j start # 10  lop1:  bgezal $12,2 # 14:if $12≥0,$31=(14+4)>>2=18>>2=24>>2=6,jump PC+4+2<<2,PC=20  jal start # 18  nop # 1c  bltzal $0,3 # 20:if $9<0,jump PC+4+3<<2,不分支  lop2:  jalr $30,$0 # 24:$30=(24+4)>>2=28>>2=40>>2=10=a,PC=0  #PC:  #00,04,08,0c,14,20,24,0 |

3.2 截图



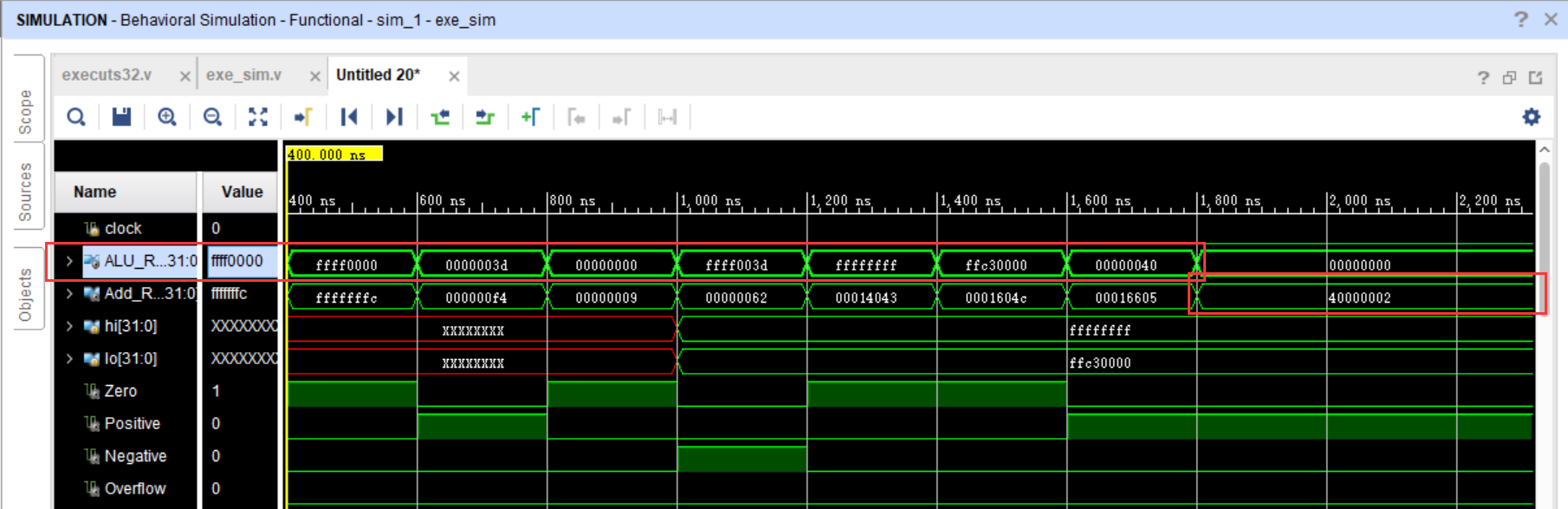


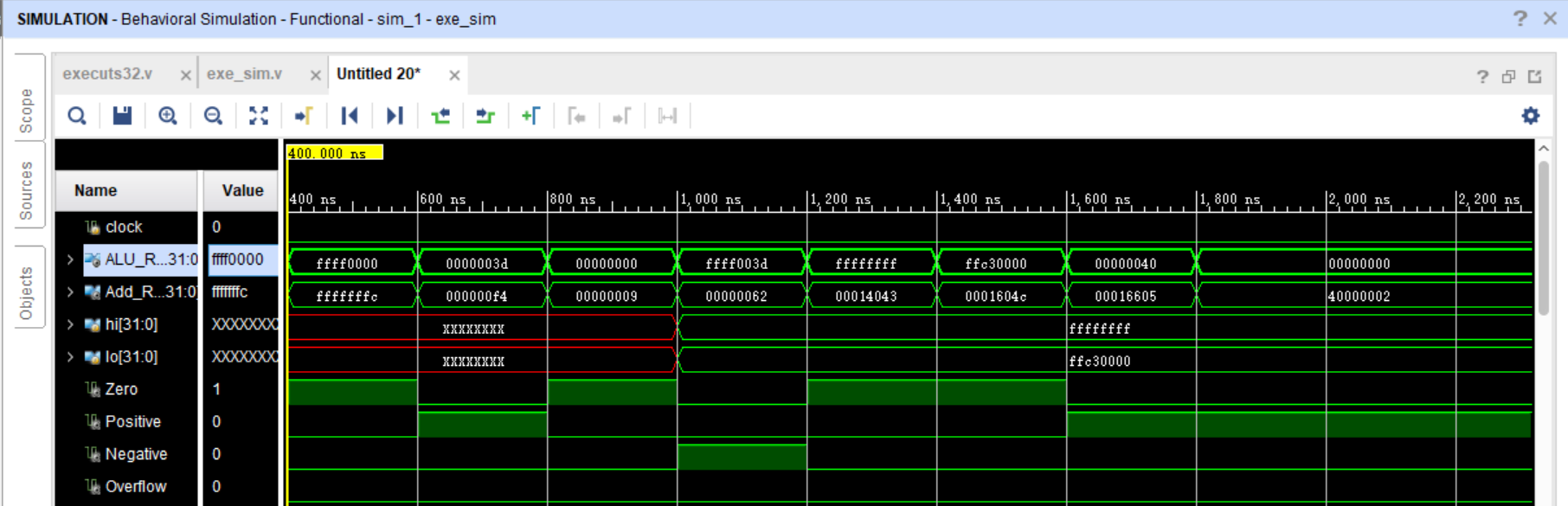
1. Exe

4.1 exe-sim.asm

|  |
| --- |
| .DATA 0X0  .TEXT 0X0000  start:  lui $8,0xffff # 00:$8=0xffff0000  addiu $9,$0,61 # 04:$9=61=111101=0x0000003d  andi $10,$5,2 # 08:$10=$5&2=0  mult $8,$9 # 0c:有符号乘法,HI,LO=$8\*$9=0xffffffff,0xffc30000  mfhi $10 # 10:$10=0xffffffff  mflo $11 # 14:$11=0xffc30000  sll $11,$1,6 # 18:$11=1000000=0x40  lop:  beq $12,$1,lop # 1c:if $12=$1,jump lop,不分支  lop1:  nop # 20 |

4.2 截图





1. Memory

5.1 ram-sim.asm

|  |
| --- |
| .DATA 0X0  .TEXT 0X0000  start:  lb $10,4($0) #00:$10=Memory[4/4]=0xffffffaa,低八位符号扩展  lbu $10,4($0) #04:$10=Memory[4/4]=0x000000aa,低八位零扩展  lh $10,12($4) #08:$10=Memory[16/4]=0xffffaaaa,低16位符号扩展  lhu $10,12($4) #0c:$10=Memory[16/4]=0x0000aaaa,低16位零扩展  sb $7,16($0) #10:Memory[16/4]的第一个字节=($7)7..0  lw $10,16($0) #14:$10=0x0aaaaa07 |

5.2 ram.coe

|  |  |  |  |
| --- | --- | --- | --- |
| ram3 | ram2 | ram1 | ram0 |
| memory\_initialization\_radix = 16;  memory\_initialization\_vector =  00,  00,  00,  00,  0a,  ff; | memory\_initialization\_radix = 16;  memory\_initialization\_vector =  00,  00,  00,  3f,  aa,  ff; | memory\_initialization\_radix = 16;  memory\_initialization\_vector =  00,  00,  01,  ef,  aa,  ff; | memory\_initialization\_radix = 16;  memory\_initialization\_vector =  55,  aa,  01,  0f,  aa,  ff; |

5.3 仿真截图

