Part 5

Bei Zhang and Tiantian Xie

1. **What did you learn from this project?**

We have learned the basic structure of CPU，Verilog language, Modelsim software, RTL simulation and how to play with the FPGA board. Also, after finished this project, we have a different understanding in how pipeline datapath works. We also developed my patience in doing endless debug, and tried to avoid the mistakes taken by careless thinking.

1. **What would you do differently next time?**

We should consider more about the hazards which might be happen if we do it again. We can add some forwarding component for branch instruction to increase the efficiency. However, this time, we just use bubble to avoid the three kinds of hazards in pipeline data path.

1. **What is your advice to someone who is going to work on a similar project?**

You’d better use Verilog instead of VHDL for coding. If you use VHDL, you might face some library setting in the last step when using Quartus. Also in industry, Verilog HDL is much more popular than VHDL.

The Quartus II has already been renewed, but the tutorial in the description of part5 hasn’t yet. Try to learn using this software by yourself. Don’t follow the steps in the tutorial! Otherwise, you will never get the result.

Tab.1 Assembly code and binary code of the final test program

|  |  |
| --- | --- |
| Test programs | |
| a=7; % a is saved in $5  b=1; %b is saved in $6  for a>0  {  b=b+1;  a=a-1;  }  if(a<b)  {c=1 ;} %c is saved in $8  d=a & b; %d is saved in $12  e=a | b; %e is saved in $13  f=b-a; %f is saved in $14 | |
| Assembly code | Binary code |
| sw $7, 0($7)  lw $5, 0($7)  add $6, $0, $1  halt  halt  halt  beq $5, $0, 6  halt  halt  halt  addi $6, $6, 1 %6 change to 8  addi $5, $5, -1 %5 change to 0  j 3  halt  slt $8,$5,$6 %8change to 1  and $12,$5,$6 %12 change to 0  or $13,$5,$6 %13 change to 8  sub $14,$6,$5 %14 change to 8 | 0001 0111 0111 0000  0000 0111 0101 0000  0010 0000 0001 0110  1111 0000 0000 0000  1111 0000 0000 0000  1111 0000 0000 0000  0111 0101 0000 0110  1111 0000 0000 0000  1111 0000 0000 0000  1111 0000 0000 0000  0011 0110 0110 0001  0011 0101 0101 1111  1101 0000 0000 0011  1111 0000 0000 0000  1001 0101 0110 1000  0101 0101 0110 1100  0110 0101 0110 1101  0100 0110 0101 1110 |