

```
1 # FILENAME: pipe1Demo.s BY: durant@msoe.edu      BEGAN: 22 April 2009
2 # $Id: pipe1Demo.s,v 1.3 2010/05/09 14:07:52 durant Exp durant $
3 # PROVIDES: CE2930 test program for the pipelined-cycle processor
4
5 # no need to implement slt* for the pipelined processor
6 # implemented instructions, arithmetic/logic,
7 # R-format: add addu and nor or sll srl sub subu
8 # I-format: addi addiu andi      ori
9 # branch,   I-format: beq bne
10 # transfer, I-format: lw sw
11
12 # demo procedure
13 # 1. assemble and test in a simulator (use MARS for correct branch offsets)
14 # 2. if bugs exist, correct and document on cover sheet, o.w. indicate no bugs exist
15 # 3. simulate in Quartus, including internal signals as needed
16 # 4. identify key results in the simulation and show to professor
17
18 # scoring if your demo doesn't work
19 # you can't demo on the due date: -10% first day, -5% each additional weekday
20
21      .text      # Executable code section
22 main:      # User program entry point
23
24      ori $t0,$zero,0x5AC3      # 3408 5ac3
25      andi $t1,$zero,0x5AC4     # 3009 5ac4 after if, ori is id
26      add $zero,$zero,$zero     # 0000 0020 after if, ori is ex
27      add $zero,$zero,$zero     # 0000 0020 after if, ori is mem
28      add $zero,$zero,$zero     # 0000 0020 after if, ori is wb
29      beq $t0,$zero,L1         # 1100 0002 branch will fail, predict not taken
30                                # is successful; andi is wb
31      bne $t0,$t1,L1           # 1509 0001 branch will succeed, need to flush
32      beq $zero,$zero,main     # 1000 fff8 should never reach this branch
33 # now, illustrate the data hazards which have not yet been resolved
34 L1:   add $t1,$t1,-1          # 2129 ffff add1, enters pipeline after successful bne flush
35      add $t1,$t1,-1          # 2129 ffff add2, add1 is id
36      add $t1,$t1,-1          # 2129 ffff add3, add1 is ex
37      add $t1,$t1,-1          # 2129 ffff add4, add1 is mem
38      add $t1,$t1,-1          # 2129 ffff add5, add1 is wb
39      add $t1,$t1,-1          # 2129 ffff add6, add2 is wb -- show that add1's
40                                # wb value is found by add6's id
41 END:   beq $zero,$zero,END    # 1000 ffff
42
43 # END OF PROGRAM
```