Computer Architecture - tutorial 2

Context, Objectives and Organization

The goals of the quantitative exercises in this tutorial, which covers Lecture 4 (pipelining and pipeline hazards), are: to provide examples that demonstrate the principles and performance implications of pipelining (E1), and to work through the scheduling of loops for the 5-stage the MIPS pipeline (E2). The exercises are extracted from the H&P book 3rd edition.

E1: H&P (2e) 3.4 pg. 217 and Example pg. 137, groups of 2 – 10 min

Problem

Consider an unpipelined machine with a 10ns clock time. For a particular workload, instructions take on average 4.4 clock cycles to execute. Now consider a pipelined version of this machine. Due to clock skew the machine adds 1ns of overhead to the clock. Assume that this overhead is fixed and each pipeline stage is balanced and takes 10ns (without the skew) in the five stage pipeline. Plot the speedup of the pipelined machine versus the unpipelined machine as the number of pipeline stages is increased from five stages to 20 stages, considering only the impact of the pipelining overhead and assuming that the work can be evenly divided as the stages are increased (which is not generally true). Also plot the "perfect" speedup that would be obtained if there was no overhead.

E2: groups of 3 – 20 min

Problem

Use the following code fragment:

```
loop: LD     R1,0(R2)
      DADDI  R1,R1,1
      SD     0(R2),R1
      DADDI  R2,R2,4
      DSUB   R4,R3,R2
      BNEZ   R4,loop
```

Assume that the initial value of R3 is R2+396. Throughout this exercise use the classic RISC five-stage integer pipeline in H&P (branch is resolved in the second stage) and assume all memory accesses take 1 clock cycle.

a. Show the timing of this instruction sequence for the RISC pipeline *without* any forwarding or bypassing hardware but assuming a register read and a write in the same clock cycle "forwards" through the register file. Assume that the branch is handled by flushing the

pipeline. If all memory references take 1 cycle, how many cycles does this loop take to execute?

b. Show the timing of this instruction sequence for the RISC pipeline with normal forwarding and bypassing hardware. Assume that the branch is handled by predicting it as not taken. If all memory references take 1 cycle, how many cycles does this loop take to execute?

c. Assume the RISC pipeline with a single-cycle delayed branch and normal forwarding and bypassing hardware. Schedule the instructions in the loop including the branch delay slot. You may reorder instructions and modify the individual instructions operands, but do not undertake other loop transformations that change the number or opcode of the instructions in the loops. Show a pipeline timing diagram and compute the number of cycles needed to execute the entire loop.

You may use the following pipeline representation diagrams to develop your solution:

a.

Inst.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

b.

Inst.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

c.

Inst.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Boris Grot. Thanks to Vijay Nagarajan, Nigel Topham and Marcelo Cintra.