Scheduling and Pipelining

Due: February 25, 11:00 am

This is a written assignment, every student must hand in his or her homework. Bring your homework to class on February 4th. SCPD students may submit their homework by e-mail via scpd-distribution@lists.stanford.edu or give your homework to the courier.

1. Consider a loop with the following precedence constraints:

Assume the loop is executed \( n \) times on a machine with no resource constraints, where \( n \) is an even number.

(a) What is the lower bound on the initiation interval? (Hint: an initiation interval is always an integer).

(b) With the software pipe-lining algorithm described in class, find a schedule meeting the lower bound. What is the number of clocks it takes to finish executing the loop \( n \) times using the scheduling you have? What is the throughput as \( n \to \infty \)? (Hint: throughput = number of iterations per clock).

(c) By applying loop unrolling first and software pipe-lining afterwards, can you improve the throughput?

   If yes, describe briefly how you would apply loop unrolling and software pipe-lining to achieve a higher throughput. Again, what is the number of clocks it takes to finish executing the loop \( n \) times using the scheduling you have? What is the throughput as \( n \to \infty \)?

   If no, please explain why.
2. Consider a simple loop below

```c
for(i = 1; i < n; ++i) {
}
```

Assume LD and ST take 1 clock, ADD takes 2 clocks. The machine has two MEM units that can execute one LD and one ST, and one ALU unit that can execute ADD in one clock. The machine can autoincrement address registers.

(a) Draw the data dependence graph by showing three types of nodes, LD, ADD, and ST.

(b) What is the lower bound on the initiation interval?

3. Consider the following code, that is executed in one iteration of a loop

```c
LD R2 , 0(R9++)
LD R3 , 0(R8++)
MUL R4 , R2 , R3
ADD R5 , R4 , R4
DIV R6 , R4 , R5
ST 0(R7++) , R6
```

DIV takes 5 clock cycles. MUL takes 4 clock cycles. ADD takes 3 clock cycles. All other operations take 1 clock cycle. We have 3 DIV, MUL and ADD units each and we can perform 1 LD and 1 ST per clock cycle.

(a) What is the number of extra registers you need to remove any unnecessary dependences caused due to registers that can produce a tight code?

(b) How many times do we have to unroll the loop? Show the code generated.
4. Consider the following dependence graph for a single iteration of a loop, with resource constraints:

Instructions A, D and E take 2 clock cycles and B and C take 1 cycle each.

(a) What is the bound on the initiation interval $T$ according to the precedence and resource constraints for this program?

(b) What is the minimum initiation interval? Show a modulo reservation table for an optimal software pipelined schedule. Also show the code and schedule for an iteration in the source loop.

(c) Can the scheduling algorithm described in class produce the optimal schedule for this loop? Discuss whether and how the original algorithm should be improved.