

University of Pennsylvania
Department of Electrical and Systems Engineering
Electronic Design Automation

ESE535, Spring 2008

Assignment #6B

Wednesday, April 9th

Due: Monday, April 21st, beginning of class.

Resources You are free to use any books, articles, notes, or papers as references. Provide citations in your writeup as appropriate.

Collaboration Please work independently on this assignment.

Writeup Writeup should be in an electronically readable format (HTML or PDF preferred—I do not want to decipher handwriting or hand-drawn figures). State any assumptions you need to make.

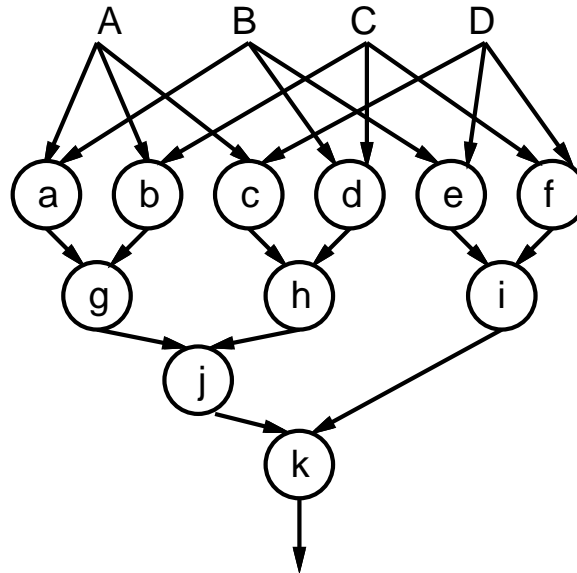
Problems

1. [2pts] Consider the following C function:

```
int f(int a, int b, int c, int d)
{
    int t;
    t=(a+b)+(c+d);
    t=t>>2;
    a=a-t;
    b=b-t;
    c=c-t;
    d=d-t;
    if (a>b) { t=a; a=b; b=t; }
    if (c>d) { t=c; c=d; d=t; }
    if (b>d) { t=b; b=d; d=t; }
    if (a>c) { t=a; a=c; c=t; }
    return(d-a);
}
```

- (a) Create the dataflow graph for this function.
- (b) What is the critical path length for the resulting dataflow graph?
- (c) Schedule the dataflow graph onto 3 ALUs. Assume the ALU has an operation:
 $res \leftarrow \text{mux}(\text{binary_condition}, A, B)$

2. [2pts] Consider the following dataflow graph.



Provide two schedules of the graph onto a set of 2 Functional Units. Both should minimize the schedule length. A, B, C, D start as live state (so the minimum storage required is 4 memory locations); each of these initial 4 memory locations can be reused when the input is no longer needed (*i.e.*, is no longer live). As part of your schedule, list the set of variables which are live following each time step.

- (a) One schedule should attempt to minimize the number of necessary memory locations needed to store intermediate values during the calculation.
 - (b) One schedule should attempt to maximize the number of live memory locations needed to store intermediate values during the calculation.
3. [4pts] Develop a scheduling algorithm to minimize the amount of memory needed to hold live values during execution on a given (fixed) number of functional units (*i.e.*, a general algorithm for solving problems like the previous one).
- (a) Provide your algorithm.
 - (b) How does your algorithm relate to (any of) the algorithms and approaches described in the course?
 - (c) Characterize the runtime of your algorithm? (*i.e.*, state the asymptotic complexity)
 - (d) Describe its optimality.
 - (e) Use your algorithm to schedule the graph in the previous problem; as before, show the set of variables live following each time step.

4. [2pts] Consider the following two-sided channel routing problem:

top	1	nc	nc	3	2	1	3	4	1	2	3	nc	4	5
bottom	nc	2	1	3	4	2	5	5	nc	4	3	4	1	nc

nc = no connection

Assume you have two layers of metal for routing.

- (a) Draw the vertical constrain graph (VCG)
- (b) Resolve any conflicts which the VCG highlights; show the revised problem.
- (c) Route the channel, minimizing the number of tracks; show the routed track.