System-on-Chip (SoC) Design

ECE382M.20, Fall 2021

Homework #3

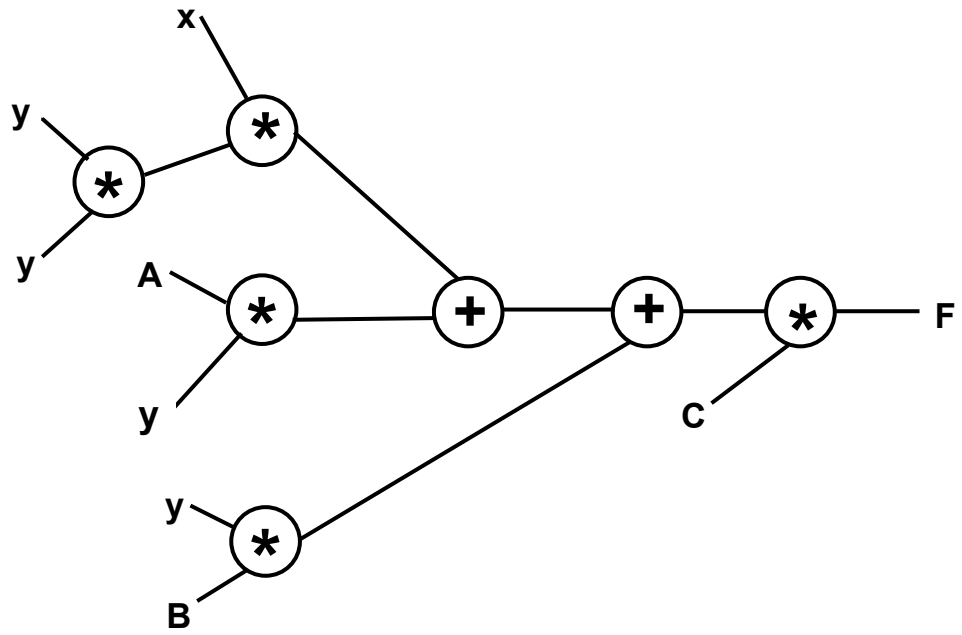
Assigned: October 14, 2021
Due: October 28, 2021

Instructions:

- Please submit your solutions via Gradescope. Submissions should include a single PDF with the writeup and single Zip or Tar archive for source code.
- You may discuss the problems with your classmates but make sure to submit your own independent and individual solutions.

Problem 1: High-Level Synthesis (100 points)

Consider the following dataflow graph (DFG) of a computation and an RTL resource library that contains 2-input adders with one cycle latency consuming *E* units of energy per addition and non-pipelined 2-input multipliers that require two cycles and 4*E* energy per multiplication.



(a) Apply behavioral optimizations to the DFG in order to minimize the tree height in cycles, i.e. the ASAP schedule length. How many cycles does it take to execute the computation assuming unlimited resources?

(b) For this sub-question only, assume that multipliers take 30 ns and adders take 10 ns. Considering multicycling and chaining, what is the fastest execution time possible for the original DFG? What is the cycle time? What is the minimum number of each resource required to achieve this time? Now image that cycles must take at least 20 ns. What is the new fastest possible execution time? Does the minimum number of resources change with this new constraint?

(c) Apply behavioral optimizations in order to minimize the energy consumption without increasing the minimal latency from (a) required for the overall computation. What is the energy required for the original DFG, your DFG from (a) and the energy-optimized DFG?

(d) Assuming a resource allocation of one adder and one multiplier, apply a list scheduling algorithm using operation mobility as priority to schedule the graph into a minimum number of cycles. Show the steps of the algorithm and the final schedule and latency obtained.

(e) Use the left-edge algorithm to determine a minimal set of registers and a corresponding register binding for your solution from (d).

(f) For your implementation in (e), draw a multiplexer-based realization of your final datapath and show the state machine of the controller driving the datapath computation.