Homework #10

This homework is due on **Friday May 22nd by 2pm**. Homework will be accepted in the EECS150 box outside 125 Cory Hall. Late homework will be penalized by 50%. No late homework will be accepted after the solution is posted.

1. For the given circuit, assume that $T_{\text{SHIFTER}}=10\text{ns}$, $T_{\text{ADDER}}=12\text{ns}$, $T_{\text{FF (setup and clock to Q)}}=2\text{ns}$.

   ![Circuit Diagram]

   a) What is the *unpipelined* throughput for the circuit?
   b) Draw a new version with two pipeline stages and maximum throughput.
   c) What is the new *throughput*? What is the new *latency*?

2. The following RTL describes one iteration of a looped computation performed on a simple processor comprising a data-path and a controller. The computation takes three constant inputs A, B, and C, and produces one output per iteration a bus labeled $Z$. All other “variables” in the RTL are registers:

   \[
   X_1 \leftarrow A, \quad X_2 \leftarrow Y_2; \\
   Y \leftarrow X_1 \times X_2, \quad X_1 \leftarrow B, \quad X_2 \leftarrow Y_1; \\
   T \leftarrow Y, \quad Y \leftarrow X_1 \times X_2; \\
   X_1 \leftarrow C, \quad X_2 \leftarrow Y; \\
   Y \leftarrow X_1 + X_2; \\
   X_1 \leftarrow T, \quad X_2 \leftarrow Y; \\
   Y \leftarrow X_1 + X_2; \\
   Y_2 \leftarrow Y_1, \quad Y_1 \leftarrow Y, \quad Z \leftarrow Y;
   \]

a) Recall that an abstract computation-graph is a graph that describes a computation. A well-formed abstract computation-graph minimizes registers and exposes parallel operations. Therefore, operators that are shared in the circuit should be separate in the graph. Furthermore, the only registers in the graph are those needed for holding feedback values. Draw the abstract computation graph for the above computation.

b) Based on your answer for part a) draw a graph for a computation that completes 4 loop iterations in parallel, producing 4 Z values per cycle.

3. For the 8-bit processor presented in class, write out the detailed micro-code for the add and ldb instructions. Do this by filling in a table with one column for each control bit needed in by the data-path, plus one to control instruction dispatch. Fill in the table with 1s and 0s, one row per cycle.

4. The 8-bit processor presented in class has a data-path and RTL description designed to minimize the cycle time at the expense of many cycles per instruction.

a) Modify the data-path and RTL to minimize the number of cycles per instruction at the expense of increased cycle time. Redraw the data-path and rewrite the RTL to reflect your new design. Assume the memory elements (register file, and memory) have asynchronous read and synchronous write.

b) Making reasonable assumptions about delays through the data-path elements, compare the total time per instruction for your solution versus the one presented in class.

5. A simple processor takes input from one memory and writes its result to another, as shown below:

MEM1 supplies a 64-bit data item every 10ns, and MEM2 can accept a new 16-bit item every 10ns. Data in MEM1 is organized in groups of four 16-bit words, \{a,b,c,d\}. Your processor must form \( y = a + b + c + d \). Assume that you have 16-bit adders with a delay of 10ns and registers with \( T_{\text{setup}}=0.5\text{ns} \) and \( T_{\text{CLK} \rightarrow Q}=0.5 \). Your job is to design the processor data-path that can supply a new \( y \) value to MEM2 every 12ns, while minimizing the cost. Draw your data-path below. You do not need to show the controller.
6. A simple processor receives three n-bit wide inputs, W, X, and Y each cycle and is required to produce an n-bit output Z per cycle, where:

\[ Z_i = Z_{i-1} + W + X + Y. \]

Draw the abstract computation-graph for the processor. Draw a block diagram (registers and adders) for a circuit design that maximizes the throughput of this calculation. Draw the resource utilization chart showing at least four iterations of the computation.