Sample Final Exam
Adv Digital System Design

Dr Shoab A Khan

Marks: 100 Time: 3 hrs

(This exam is designed for OPEN BOOK & OPEN NOTES)

Q1. **Marks 25 = 15 + 10 (Take home)**

Design a distributed arithmetic based IIR filter architecture that implements following difference equation

$$y[n] = 4'b0100y[n-1] + 4'b0010y[n-2] + 4'b1101x[n]$$

Where all the 4-bit coefficients are in Q1.3 bit signed numbers. Design one lookup table, assume $x[n], y[n-1], y[n-2]$ are 4-bit signed numbers and $y[n]$ is an 8-bit signed number, truncate $y[n]$ to 4-bits while feeding it back to the design. (Copy your code on a separate sheet for take home exam. Take your time to code and simulate YOUR CODED design using any Verilog simulator and demonstrate your implementation with the Semester project.)

Q2. **Marks 30=(3,3,7,3,3,3,8)**

For the difference equation

$y[n] = 0.921y[n-3] + 0.432y[n-7] + 0.634x[n-3] + 0.321$x[n]

Assume multiply takes 2 and adder 1 time unit, do the following:

(a) Draw the DFG that implements the difference equation
(b) Identify all the loops in the DFG and compute IPB
(c) Unroll the DFG using unfolding transformation by a factor of 3.
(d) Compute the IPB of the unfolded DFG
(e) Retime the design by placing registers to paths that helps in achieving IPB (don't not pipeline the computational units),
(f) For the retimed design give non zero values of *r(i)s* and $w_r(e_{ij})s$
(g) Fold the original DFG to two multipliers and two adders, assume both multiplier and adder takes one time unit and are without any pipelining, consider any folding schedule that yields legal solution to the folding problem

Q3. **Marks 25=(8,10,7)**

Design a micro coded state machine that optimally implements the following micro codes

if($cond_m$) (R_k = R_i OP_n R_j) else (R_k = R_i OP_p R_j)
where $i, j, k \epsilon \{0,1,2,3, \ldots, 7\}$, $m = 0,1$ and $OP_n, OP_p \epsilon \{+, -, \&, |, \hat{}\}$
Jump LABEL,
If ($cond_m$) Jump $LABEL_0$ else Jump $LABEL_1$
If ($cond_m$) Call Subroutine at $LABEL_0$ else Call Subroutine $LABEL_1$
return

Where R is a register file with eight registers R_0, R_1, R_2, R_3,…,R_7, and all the registers are 16-bit wide. If the result of an arithmetic and logic micro code is zero or negative it sets the zero-flag ($cond_0$) or negative-flag ($cond_1$) respectively. Assume Program Memory (PM) is 512 deep. The LABEL, $LABEL_0$ and $LABEL_1$ are any legal addresses in the PM. Assume all operands are signed numbers.

(a) Draw RTL diagram for the design of the datapath and the controller. Clearly identify all control signals and their widths.
(b) Write RTL Verilog code of the controller ONLY (don't code datapath)
(c) Give micro code format for your design by giving width of different fields in the micro-code.

Q4. **Marks 20=(3,3,4,5,5)**

(a) Eliminate Horizontal sub-expression while multiplying $x[n]$ with the following 18-bit CSD number

$$0100\bar{1}0\bar{1}00100010\bar{1}0$$

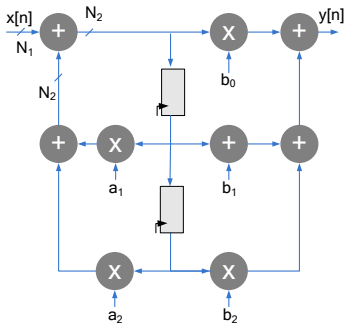
(b) Eliminate Horizontal and Vertical common sub-expression while implementing FIR filter with the following set of coefficients in CSD representation,

$$h[0] = 1000\bar{1}000000$$
$$h[1] = 100000\bar{1}0001$$
$$h[2] = 010000\bar{1}0000$$

Give the final difference equation of the filter.

(c) Rewrite the recursion equations for DDFS if $\Delta\theta_i = 3^{-i}$ for $i = 0,1,2,3, \ldots, 15$

(d) Using look-ahead transformation, add extra register in the feedback loop of the following DFG



(e) Design a **micro-program memory** based controller on a 2-bit interface that detects a sequence of following bits

01
10
11
00

Draw bubble diagram and fill the PM that implements the bubble diagram.