King Fahd University of Petroleum and Minerals

COLLEGE OF COMPUTER SCIENCES AND ENGINEERING

Computer Engineering Department

Sample PhD Comprehensive Exam

Notes:

1. The examination covers 3 areas;
    i. Computer Architecture and High Performance Computing,
    ii. Computer Networks and Security, and
    iii. Digital System Design & Automation.
2. With an average of 1:30 hours per area, the total exam time is 4:30 hour

Sample Exam Questions for Area 1: Computer Architecture and High Performance Computing

Question 1: Computer Performance (25 Minutes)

A compiler designer is trying to decide between three code sequences for a particular processor. The hardware designers have provided the following data below about the CPI for each class of instructions. Also, the instruction counts for each sequence have been provided.

Class	CPI for this instruction class
A	3
B	5
C	2
D	7
E	4

Code Sequence	Instruction Counts for Instruction Classes				
	A	B	C	D	E
1	200	125	35	80	42
2	155	127	55	29	96
3	188	144	87	23	77

Answer each of following questions:

1. How many cycles are required for each code sequence?
2. Which is faster and by how much compared to the slowest?
3. What is the average CPI for each code sequence?

Question 2: Dynamic Branch Prediction (25 Minutes)

A core program P consists of a sequence of three Conditional Branches denoted as CB-1, CB-2, and CB-3. The outcomes of these branches depend upon the value of a variable A at P entry. The following Table gives the CB outcomes when P is executed starting with specific values of A.

Value of A	CB-1	CB-1	CB-1
1	T	T	T
2	NT	T	NT
3	NT	T	T

Answer the following questions:

a) The following table lists the CB prediction (CB-P), the CB action (CB-A), and CB Update (CB-U) for all the three CBs. Each row corresponds to one run of P starting with a value of A. Fill in the Table below by setting the prediction, action, and update for each CB using 1-bit predictor which is initialized to predict “NT”. Evaluate the prediction accuracy (Pa).

Value of A	CB-1: P	CB-1:A	CB-1:U	CB-2: P	CB-2:A	CB-2:U	CB-3: P	CB-3:A	CB-3:U
2									
1									
3									

b) Repeat the above question using a 2-bit predictor, initialized to predict "NT" and evaluate the prediction accuracy (Pa).

Value of A	CB-1: P	CB-1:A	CB-1:U	CB-2: P	CB-2:A	CB-2:U	CB-3: P	CB-3:A	CB-3:U
2									
1									
3									

c) Repeat the above question using a correlating two-level GAp (1,1) branch predictor, initialized to predict "NT" and evaluate the prediction accuracy (Pa). Assumes initially all history is set to "NT".

Value of A	CB-1: P	CB-1:A	CB-1:U	CB-2: P	CB-2:A	CB-2:U	CB-3: P	CB-3:A	CB-3:U
2									
1									
3									

QUESTION 3: INSTRUCTION LEVEL PARALLELISM (20 Minutes)

Consider the following floating-point loop for computing the sum of arrays X[] and Y[] (X[i] = X[i] + Y[i]) using a micro-architecture with a fully pipelined FP Unit and internal forwarding:

```
Loop:  L.D     F1, 0(R1)
       L.D     F2, 0(R2)
       ADD.D   F2, F2, F1
       SD      F2, 0(R1)
       SUBI    R1, R1, # -8
       SUBI    R2, R2, # -8
       BNE     R1, R3, Loop
```

The latency of data dependent instructions is as follows: (1) 2 clocks for Load-FP, (2) 3 clocks for FP–Store, and (3) 1 clock following each Condition Branching. Answer each of the following questions:

1. Rewrite the loop after inserting the stalls due to data hazards. Evaluate the CPI.

2. Unroll the above loop for the least number of times so that to eliminate all the stalls. Evaluate the CPI of the unrolled loop and its speedup S over the original loop.

3. Apply the software pipelining technique on the original loop. Evaluate the CPI of the software pipelined loop and its speedup S over the original loop.

Question 4: Distribyted-Memory Coherence Protocol (20 Minutes)

A Distributed-Memory Multiprocessor (DMM) uses a Flat Directory Protocols based on the Write-Back Invalidate MSI Protocol. The DMM has three nodes N1, N2, and N3. In the reference Scenario shown below, N1 writes 5 into variable A which causes the presence bits in home directory N3 to be updated. Each address must have a status and a value if it is cached.

BLOCK ACCESS SCENARIO
Initially, block A is Modified in N1, value=5, and HD(A) is N3
N1: Write 5 to A
N1: Read A
N2: Read A
N2: Write 10 to A
N3: Write 20 to A
N1: Read 3 to A

Upon a page fault, the protocol uses the following messages: Read-miss (Rm), Write-miss (Wm), Invalidate (Inv), Fetch (F), Fetch/Invalidate (FI), Data value reply (DVR), and Data write-back (DWB) to communicate among a requesting node, home directory, and owner node.

Apply the MSI protocol in filling in the Transaction Table for the block access scenario shown below. Fill in all columns whenever applicable.

Transaction Table: Flat Directory Protocols based on the Write-Back Invalidate.

TRANSACTONS	*N1*			*N2*			*N3*			*Message*	
STEPS	*State*	*Addr*	*Value*	*State*	*Addr*	*Value*	*Presence*	*Owner*	*Addr*	*Type*	*Value*
N1: Write 5 to A	*M*	*A*	*5*				*1-0-0*	1	A		
N1: Read A											
N2: Read A											
N2: Write 10 to A											
N3: Write 20 to A											
N1: Read 3 to A											

Sample Exam Questions for Area 2: Computer Networks and Security

Problem 1 (15 points):

a) ARP and DNS both depend on caches; ARP cache entry lifetimes are typically 10 minutes, while DNS cache is on the order of days. Justify this difference. What undesirable consequences might there be in having too long a DNS cache entry lifetime?

b) There are two approaches for name resolution in DNS: iterative and recursive. Explain the differences between them and give an example for each to illustrate these differences.

Problem 2 (15 points):

Assume that TCP implements an extension that allows window sizes much larger than 64 KB. Suppose that you are using this extended TCP over a 1-Gbps link with a latency of 100ms to transfer a 10-MB file, and the TCP receive window is 1 MB. If TCP sends 1-KB packets (assuming no congestion and no lost packets):

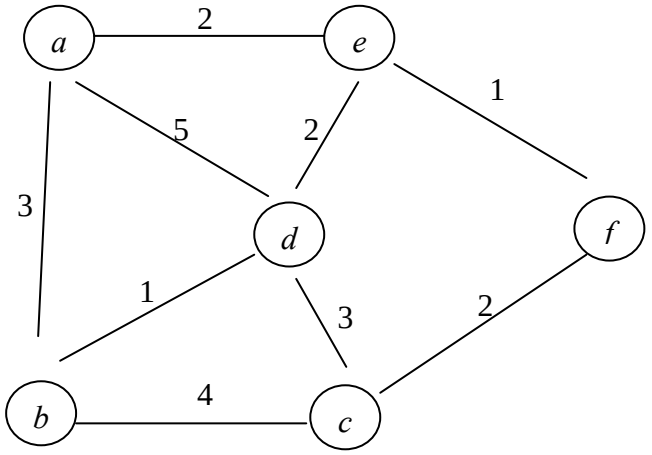
(a) How many RTTs does it take until slow start opens the send window to 1 MB?

(b) How many RTTs does it take to send the file?

(c) If the time to send the file is given by the number of required RTTs multiplied by the link latency, what is the effective throughput for the transfer? What percentage of the link bandwidth is utilized?

Problem 3 (25 points):

a) Consider the network shown below, using Kruskal algorithm; build the minimum weight spanning tree. Show all your steps



b) Now, using the link state protocol (Dijkstra's algorithm),

1. Fill in the below table showing each step of the algorithm considering Router **d** as the source.
2. Show the shortest path tree generated in (a).

Step	N	a	b	c	d	e	f
0							
1							
2							
3							
4							
5							
6							
7							

Problem 4 (15 points):

a) Consider a communication session between two computer nodes through a network of interconnected switching/routing nodes. Typically, the data link provides reliability be performing automatic repeat request (i.e. ARQ or retransmission) for erroneous frames on a hop-by-hop basis. If is desired to design a network that utilizes reliable physical links (such as fiber optics) where bit error rates are infinitesimal, would it be wise to maintain the retransmissions function on a hop-by-hop basis? Explain and justify your answer pointing out example technologies and implemented standards.

b) Go-back-N (GBN) and Selective-Reject Request (SRQ) are two popular alternatives for sliding window flow/error control protocols on data links. Assuming k bits for packet sequence numbers, what is the maximum window size, W, for GBN and SRQ. Comment on the relative efficiency (or throughput) for these protocols.

Problem 5 (15 points):

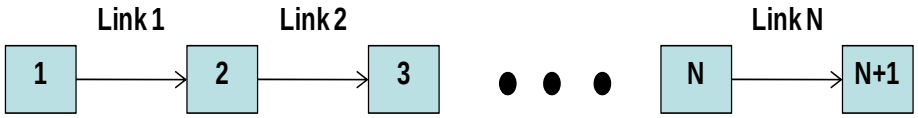
A file of size K bits is to be sent from Node 1 to Node $N+1$ traversing the intermediate N point-to-point identical links as shown in Figure below. The data link protocol operating on the links requires that V overhead bits be added to any frame transmitted on the link. Assume the link transmission rate is R bits/sec and ignore the link propagation time and node processing delay.

a. If the file is not segmented but rather sent as a whole on each of the links, write an expression for the total transmission time, T_{total1}.

b. If the file is segmented into packets of size P bits where an overhead of V bits is added to each packet, write an expression for the total transmission time, T_{total2}.

c. Find the optimal packet size, P, for minimum overall transmission time for the file in terms of N, R, K, and V. Assume file size, K, is very large compared to packet size P.

Hint: derivative rules: $\frac{d}{dx}(uv) = u'v + uv'$ and $\frac{d}{dx}\left(\frac{u}{v}\right) = \frac{u'v - uv'}{v^2}$.



Problem 6 (15 points):

For the Kerberos v4 protocol, a centralized key distribution center is crucial for facilitating mutual authentication and secure communication between any two entities wishing to communicate securely in the network.

a. What is the benefit of having a centralized KDC?

b. What type of vulnerability is a centralized authentication service prone to?

c. What is the role of timestamping of tickets issued by the KDC?

Provide explanations of the above answers

Sample Exam Questions for Area 3: Digital System Design & Automation

Question 1 (30 Points).

It is required to design a circuit that reads 4 8-bit numbers serially (i.e. starting with first number, it is read 1-bit at a time, then the 2nd number and so on) and produces their average as integer by simply dividing their sum by 4.

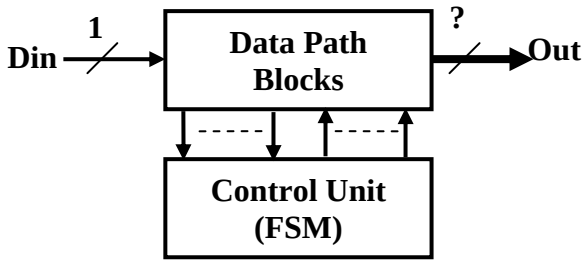
(i) Determine the number of bits required for the output

(ii) Determine the required data path blocks to design this circuit (show the block diagram of these circuits only)

(iii) Obtain a Moore-model state diagram of the control unit that will control the operation of the circuit

(iv) Design the control unit using D-FF and any other components of your choice

(v) Show the complete circuit implementation (block diagram).



Question 2 (35 Points).

(i) Give a partial VHDL or VeriLog code describing the operation of the shown data path given that:
R1: is an *n*-bit binary up-counter with parallel load and count-enable control inputs.
R2: is an *n*-bit register

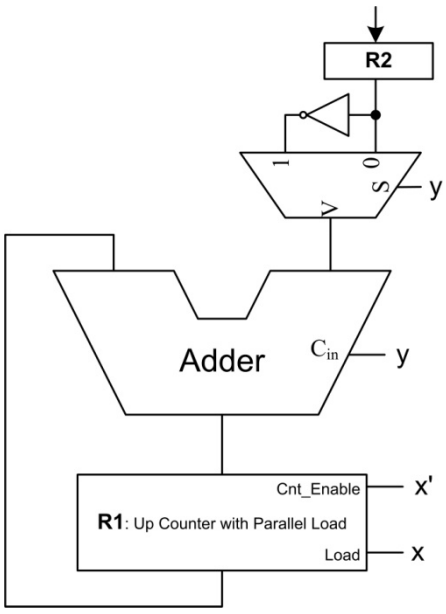
Clearly state any reasonable assumption you may make.
(you may define and use your own pseudo HDL code if you are not familiar with either VHDL or VeriLog)

(ii) Describe the operation of the data path

(iii) Testing the HDL model of this data path for $n = 4$, you obtained the following results in successive clocks

Clk		1	2	3	4
R1	1011	1110	1111	0101	0110
R2	0011	1101	1010	1001	0011
X	0	1	0	1	
Y	0	0	1	1	

Are these results correct? or are they erroneous? If they are correct justify but if they are erroneous what is the most likely source of error? Explain.



Question 3 (35 Points).

Consider the network given below with inputs *{a, b, c, d, e, f, g, h}* and outputs *{Y, Z}*. Assume that the delay of both the Adder and the Multiplier fit within one clock cycle and that the input values will be available to the circuit for only one clock cycle. Also assume that both addition and subtraction operations will be performed by the Adder.

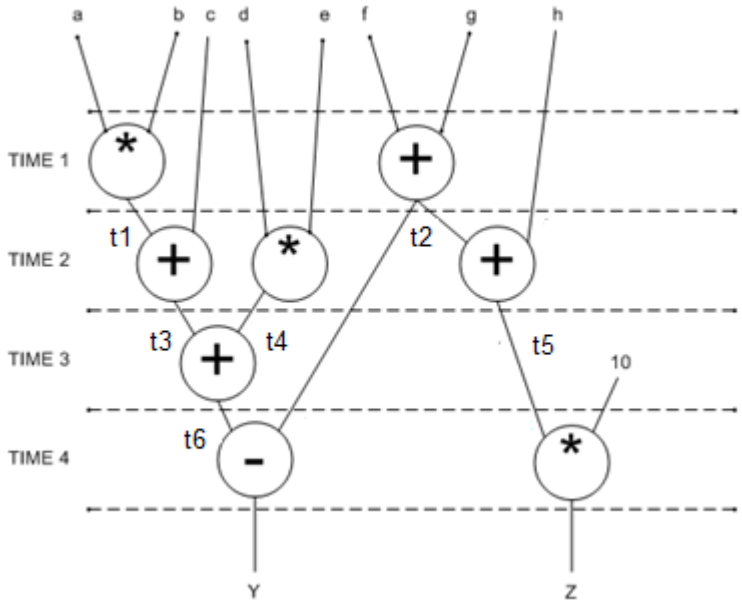
[1] = a * b; [2] = d * e; [3] = f + g; [4] = [1] + c; [5] = [3] + h;

[6] = [4] + [2]; Y = [6] – [3]; Z = [5] * 10;

(i) Schedule the sequencing graph into the **minimum number of cycles** under the resource constraints of one Adder and one Multiplier.

(ii) Assume that the delay of the Adder takes one clock cycle while the delay of the Multiplier takes two clock cycles. Schedule the sequencing graph under the latency constraint of **5 clock cycles** minimizing the number of resources required.

(iii) Consider the scheduled sequencing graph below:



a. Show the life-time of all variables.
b. Determine the minimum number of registers that are required to store all the variables. Show the mapping of variables to registers. Select a mapping that **minimizes the number of multiplexers and interconnect area** as much as possible.
c. Draw the **data-path** implementing the scheduled sequencing graph based on the variable-register mapping that you obtained in (b).