PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING

Midterm Examination: Semester 2 **Academic Year**: 2014

Date: 16 March 2015 **Time**: 9.00 – 12.00 (3 hours)

Subject Number: 242-441 **Room**: R200

Subject Title: Advanced Computer Architecture and Organization

---

Exam Duration: 3 hours

This paper has 14 pages, 8 questions and 165 marks (25%).

Authorised Materials:

- Writing instruments (e.g. pens, pencils).
- Textbooks, a notebook, handouts, and dictionaries are permitted.

Instructions to Students:

- Scan all the questions before answering so that you can manage your time better.
- Answers **must** be written in **Thai**.
- Write your name and ID on every page.
- Any unreadable parts will be considered wrong.

When drawing diagrams or coding, use good layout, and short comments; marks will not be deducted for minor syntax errors.

Cheating in this examination

Lowest punishment: Failed in this subject and courses dropped for next semester.

Highest punishment: Expelled.

NO	Time (Min)	Marks	Collected	NO	Time (Min)	Marks	Collected
1	30	30		5	10	10	
2	33	33		6	10	10	
3	12	12		7	30	30	
4	20	20		8	20	20	
Total	165	165			25%		

Question 1 General Knowledge (30 marks; 30 minutes)

Tell whether the following statements are true (T) or false (F).

a) _____ In *Bit-Level Parallelism*, speed-up was driven by doubling computer word size which is the amount of information the processor can manipulate (จัดการ) per cycle.

b) _____ *Data parallelism* usually scales with the size of a problem.

c) _____ Memory hierarchies take advantage of *memory locality*.

d) _____ Usually, *Miss Penalty* is much smaller than *Hit Time*.

e) _____ Normally, *Hit Rate* is much smaller than *Miss Rate*.

f) _____ *Miss Penalty* is time to replace a block from lower level plus time to replace in CPU.

g) _____ The Least Recently Used (LRU) *block replacement* in cache is easy for *high associativity*.

h) _____ *Average Memory Access Time* can be calculated from Hit Time, Miss Rate and Miss Penalty.

i) _____ *CPU Time* can be calculated from *Instruction Count*, *Clock Cycles per Instruction* and *Clock Cycle Time* and does **not** involve *Miss Rate* and *Miss Penalty*.

j) _____ The higher number of cache levels, the shorter overall memory access time.

k) _____ The higher associativity, the smaller Conflict Misses, the longer Hit Time and more power consumption.

l) _____ The larger total cache capacity, the smaller miss rate, the longer hit time and more power consumption.

m) _____ The larger the block size, the smaller Compulsory Misses, the bigger capacity and Conflict Misses, and Miss Penalty.

n) _____ *Giving priority to read misses over writes* helps reduce *Miss Penalty*.

o) _____ Increasing associativity increases *Conflict Misses*.

p) _____ *Prefetching* relies on having *extra memory bandwidth* that can be used without penalty.

q) _____ *Prefetching* may cause exceptions (e.g. page faults).

r) _____ *Virtual Memory Architecture* keeps processes in their own memory space, provide mechanisms for switching between user mode and supervisor mode, and to limit memory accesses.

s) _____ *Virtual Machine Monitor (VMM)* determines how to map virtual resources to physical resources.

t) _____ In *Virtualization*, physical resources may be time-shared, partitioned, or emulated in software.

u) _____ An architecture with simple instructions reduces the complexity of the control unit and the data path so that the processor can work at a high clock

frequency and pipelines can be used more efficiently.

v) _____ *Loop Unrolling* increases impact of branches on pipeline.

w) _____ Branch target calculation is costly and stalls (เหนี่ยวรั้ง/ทำให้ชะงักหรือหยุด) the instruction fetch.

x) _____ SIMD is more energy efficient than MIMD because it can fetch more than one instruction per data operation.

y) _____ It is possible to spread the work of one program across p processors and achieve a speedup greater than p.

z) _____ Making a one-processor system into a p-processor system automatically improves single-threaded performance by a factor of p.

aa) _____ A shared memory style of parallel programming does not allow processes to have local variables.

bb) _____ *Vector Processors* are deeply pipelined, while *GPUs* hide latency with multithreading.

cc) _____ The principles of computer design concerns Parallelism and Locality.

dd) _____ In *Cache Block Replacement*, when we want to increase the associativity, we have to shrink the tag and expands the index in the block address.

Question 2 Comparisons (34 marks; 34 minutes)

Compare the following items.

a) SIMD and MIMD (2 marks)

b) MTBF and MTTR (2 marks)

c) *Temporal and Spatial Locality* in Memory Hierarchy Management (2 marks)

d) *Hit* and *Miss* in Memory Hierarchy Management (2 marks)

e) Direct Mapped, Fully Associative and Set-Associative Block Placements in cache (3 marks)

f) *Write-Through* and *Write-Back* strategies in cache (4 marks)

g) *Giving Reads Priority over Writes on Misses* means serving reads before writes have been completed. Compare **how to implement** it when using Write-Through and Write-Back strategies in cache. (2 marks)

h) What are three advantages of *Virtual Memory*? (3 marks)

i) What are the differences between *Hypervisor Type I and II*? (2 marks)

j) Explain two different approaches of *Complex Instruction Set Computer (CISC)* and *Reduced Instruction Set Computer (RISC)* that try to enable new High Level Languages programs to be compiled and executed efficiently. (2 marks)

k) What are differences among *Structural, Data and Control Hazards*? (3 marks)

l) What are three *loop unrolling limits*? (3 marks)

m) Compare Branch History Table (BHT), 2-bit Scheme Dynamic Branch Prediction, and Correlated Predictor. (3 marks)

Question 3 Calculations (12 marks; 12 minutes)

Calculate the following answers.

a) A clock's battery needs to be changed after 5000 hours of use and it needs half an hour to remove and replace the battery, find *MTTF* and *MTBF*. (2 marks)

b) Find the *speedup* when there are 2 processors and 60% sequential code. (2 marks)

c) Find the *speedup* when there are 16 processors and 40% parallelizable code. (2 marks)

d) Find the *speedup* when there are 16 processors and 20% sequential code. (2 marks)

e) What is *the upper bound of the overall speed up* when the code is 80% parallelizable (2 marks)

f) According to *2:1 Cache Rule*, the miss rate of a 2-way set associative cache of size 32 Kbyte equals to the miss rate of a direct mapped cache of what size? (2 marks)

Question 4 Explanation (20 marks; 20 minutes)

Answer the following questions.

a) Explain the **causes** of Cache Misses. (3 marks)

b) How to improve Cache Performance? (3 marks)

c) How can larger block size help reduce cache misses? (3 marks)

d) Why increasing block size does not help when the cache size is larger? (2 marks)

e) What are 4 disadvantages of higher associativity? (4 marks)

f) Give at least 5 benefits of Virtualization. (5 marks)

Question 5 Memory Management (10 marks; 10 minutes)

Explain the problems in the following programs and show how we can improve the performance. Also, demonstrate the improved codes.

a) (4 marks)

```
for (a=0; a<10; a++)
  for (b=0; b<100000; b++)
    x[a][b] = f(x[a][b]);
```

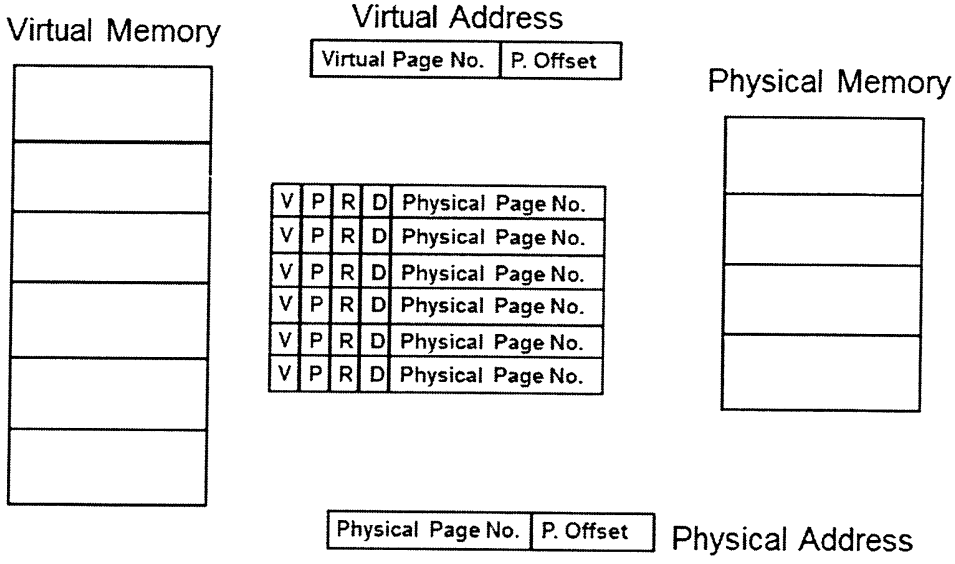
b) (6 marks)

```
for (a=0; a< 100000; a++)
  for (b =0; b < 100000; b++) {
        sum = 0;
        for (c=0; c<100000; c++)
          sum+= H[c][b]*G[a][c];
        F[a][b] = sum;
}
```

Question 6 Domonstration (10 marks; 10 minutes)

Answer the following questions.

a) From the following picture, explain the page table entry and demonstrate the relationship between the virtual memory and physical memory. (5 marks)



Name_____ ID_____

b) How do Table Look-Aside Buffer work with Page Table Entry? (5 marks)

Question 7 Instruction-Level Parallelism (30 marks; 30 minutes)

Answer the following questions.

a) Explain how we can improve the performance of the following program and also demonstrate the improve code. (4 marks)

```
for (i=0; i<100000; i=i+2)
    a[i] = b[i] * c[i];
```

b) Give two examples each from the following code of true data dependencies, anti-dependencies, and output dependencies. (6 marks)

S1: LD F0, 0(R1)

S2: ADDD F4, F0, F2

S3: ADDD F6, F0, F4

S4: LD F2, 0(R3)

S5: SUBI R3, R3, #4

S6: MULT F6, F12, F2

S7: LD F2, 0(R3)

c) Explain how we can improve the performance of the following program and also demonstrate the improve code. (6 marks)

I: add r1, r2, r3

J: sub r4, r1, r3

K: mul r1, r2, r3

L: div r1, r2, r3

d) Which kind of branch prediction works best for the below code? (2 marks)

```
for (i=100000; i>0; i=i–1)
  x[i] = x[i] + s;
```

---

e) Which kind of branch prediction is suitable for the below code? (2 marks)

```
for (i=0; i<100; i++)
  for (j=0; j<3; j++)
    x[i] = y[j] + s;
```

---

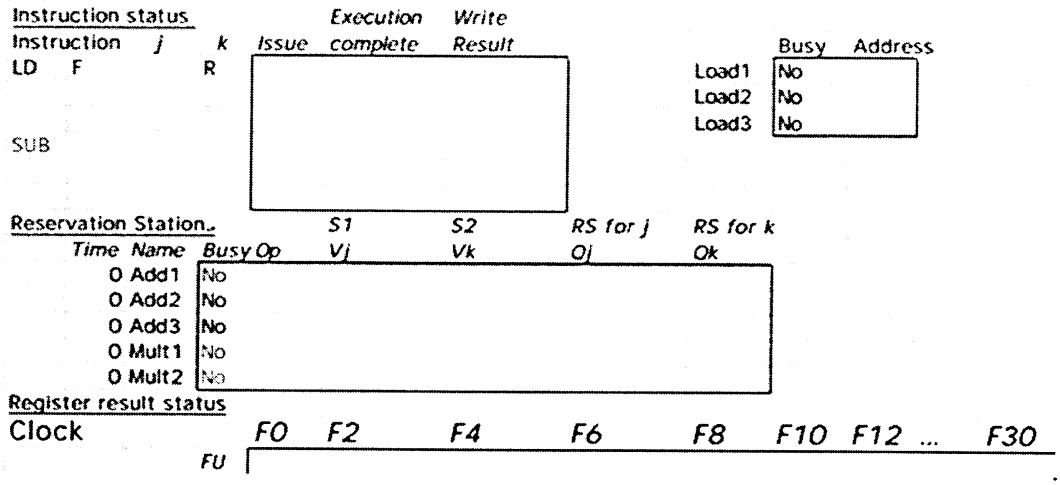
f) Consider the following instructions, fill in the first three cycles applying the Tomasulo algorithm. (10 marks)

LD F0, (0)R1

MULTD F4, F0, F2

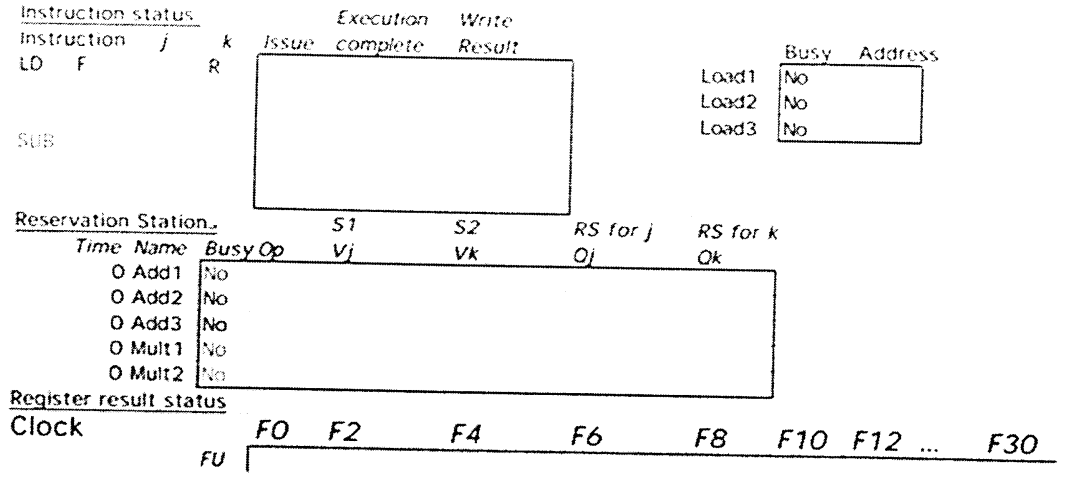
LD F8, (0)R1

SUBI R1, R1 #8

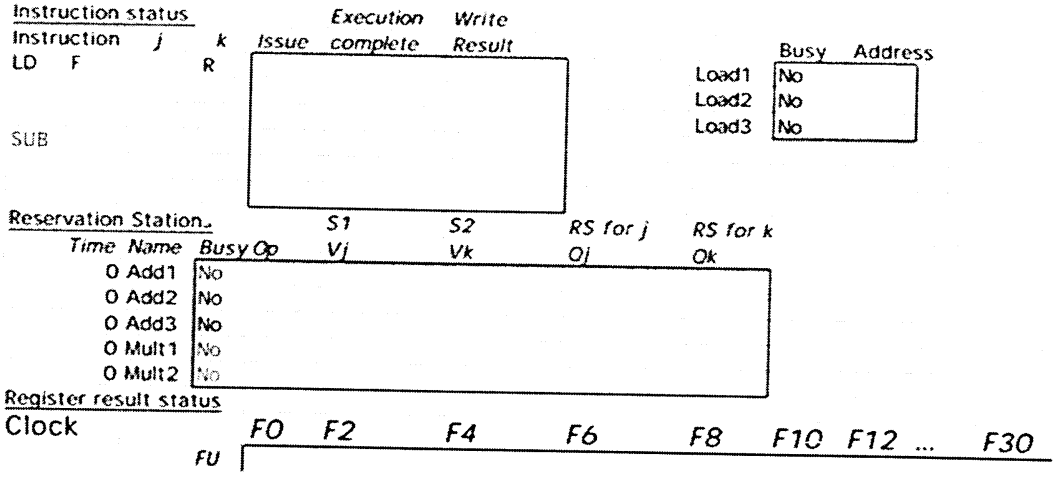


Clock Cycle 1

Name_____ ID_____



Clock Cycle 2



Clock Cycle 3

Name_____ID_____

Question 8 Data-Level Parallelism (20 marks; 20 minutes)

a) Explain the concepts of *Convoy* and *Vector Chaining* and give examples. (2 marks)

b) Suppose that the total number of data is 1024 and the size of *Vector Length* is 32, derive the following code using *Vector Length Registers*. (5 marks)

```
for (i=0; i< 1024; i++)
  C[i] = A[i] + B[i];
```

c) From the following code, explain how to apply *Vector Mask Register*. (3 marks)

```
for (i = 128; i <=0 ; i=i-1)
  if (X[i] != 0)
    X[i] = 5*X[i];
```

d) Give an example of code that has more than 1 *stride*. (3 marks)

e) From the following code, explain how to apply *scatter and gather operation.* (3 marks)

```
for (i = 0; i < n; i=i+1)
  A[K[i]] = A[K[i]] + C[M[i]];
```

f) Compare Vector Processor, SIMD Multimedia Extension and GPU according to the following list (4 marks)

List	Vector Processor	SIMD Multimedia Extension	GPU
CPU			
Vector Mask Register			
Scalar Register			
Vector Register			

----End of Examination----

Pichaya Tandayya Lecturer

Name_____ ID_____