ENEE 446: Digital Computer Design

Fall 2018 **Handout #28**

Final Exam Review

1 Time and Location

The final exam will be given from 1:30p.m.-3:30p.m., on Saturday December 15th, in the normal meeting place, EGR 0108.

2 Format

The final exam will have an identical format to the midterm. See Handout #14 for information about exam format.

3 Scope

The final exam is comprehensive, and covers all material from the first day of class up and including the last lecture, given on December 10th. Essentially, this includes the introductory material in Chapter 1 of H&P, basic pipelining, instruction-level parallelism, caches, multicore processors, power, virtual and main memory, and data-level parallelism. However, an emphasis will be placed on caches, multicore processors, power, and virtual and main memory (the material covered in the second half of the semester, but not emphasizing data-level parallelism).

4 Course Content

Since the final exam is comprehensive, you should refer to Handout #14 for an outline of the course content relevant to the first half of the semester. Below, you will find a fairly comprehensive outline of the topics covered since the midterm. Disclaimer: this is not meant to be an absolutely water-tight complete list of topics. In other words, if there is a topic *not* present in this list, it may still show up on the final exam. However, it is a pretty good first-cut at what we have covered.

```
IV.  Memory Systems
   A.  Caches
      1.  Principle of locality
      2.  Cache organization
         a.  Direct-mapped
         b.  Set associative
         c.  Fully associative
```

d. Addressing the cache
3. Cache Management
a. Replacement policies
b. Write-hit policies
c. Write-miss policies
d. Instructions vs. data
4. Cache performance
a. Average memory access time
b. Impact on CPU time
5. 3 C's
a. Compulsory
b. Capacity
c. Conflict
6. Design tradeoffs / performance optimizations
a. Increase block size
b. Increase associativity
c. Victim cache
d. Prefetching
e. Critical-word first / early restart
f. Lock-up free caches
g. Multi-level caches
1). Property of inclusion

V. Multicore Processors
A. Parallel programming models
1. Model examples
a. Shared memory
b. Message passing
c. Others
2. Explicit parallelism
a. Partitioning
b. Communication
c. Synchronization
3. Example: Jacobi relaxation
B. Machine organization
1. Symmetric multiprocessor (SMP)
2. Cache-coherence problem
3. Memory consistency problem
4. Bus-based cache-coherence protocols
a. 3-state protocol
b. MESI protocol

VI. Power
A. The problem
1. Why we care: battery life, thermal issues, energy cost.
2. Components
a. Dynamic power due to switching
b. Static power due to leakage
3. Technology scaling impact
a. Device scaling

b. Frequency scaling
c. Voltage scaling

4. Inability to continue scaling voltage due to leakage; halting of frequency scaling

B. Assessing power

1. Metrics
   a. Individual metrics (Joules, Watts)
   b. Efficiency metrics
      1). MIPS/W
      2). MIPS^2/W
      3). MIPS^3/W
      4). PDP
      5). EDP
      6). ED^2P

C. Mitigation techniques

1. Improving both power and performance
   a. Algorithmic improvement
   b. Compiler optimizations
2. Reducing waste
   a. Clock gating
   b. Power gating
3. Dynamic voltage and frequency scaling
   a. Discrete voltage-frequency levels
   b. Cubic reduction in power, linear degradation in performance
   c. Management policies
      1). Temperature-based
      2). Performance-based
4. Exploiting thread-level parallelism
   a. Impact on power consumption
   b. Core type
5. Cache hierarchy techniques
   a. Filter cache
   b. Access tag arrays first, than data arrays
   c. Way prediction

VII. Virtual and Main Memory

A. Naming and Protection

1. How programs use names
2. Relocation: base-register addressing
3. Adding protection: base-length register addressing
4. Adding sharing: segmented addressing
   a. Segment tables
   b. Translation lookaside buffers
   c. Privilege bits

B. Resource management

1. Virtual memory
2. Fragmentation problem
3. Paged addressing
   a. Page tables
   b. Translation lookaside buffers

4. Virtual memory and caching
   a. Synonym problem
   b. Homonym problem
   c. Solutions to these problems

C. Main memory
1. Memory sub-system organization
2. Increasing memory system width
3. Increasing number of channels
4. Interleaved memory
   a. Low-order interleaving
   b. Increasing number of banks
5. DRAM architecture
   a. Row and column access
   b. Cache-line interleaving
6. Dual data rate
7. Clock rate scaling (DDR, DDR2, DDR3, DDR4)

VIII. Data-Level Parallelism

A. Vector architectures
1. Vector register files
2. Chaining
3. Streaming memory systems
   a. Sequential access
   b. Strided access
   c. Scatter-gather
4. Vector-length registers
5. Vector-mask registers
6. Multiple lanes

B. SIMD instruction extensions
1. Narrow-width data
2. x86 ISA support: MMX, SSE, AVX
3. Exposing SIMD parallelism via loop unrolling

C. GPUs