Example Exam (Ex-tenta) DARK2 991206

(Answers to appear on the course home page on 991213)
This example Exam has a bit too many sub-questions. I'll make a better attempt to specify when a short answer is sufficient at the real exam and remove some of the sub-questions.

Problem 1, random questions (8p)

a) Show how the expression C := A + B is computed on a stack machine (2p)
b) What is key advantage of a stack machine over a load/store architecture?
c) What is an anti dependence and an output dependence?
d) What is a TLB?

Problem 2, pipelining (8p)

a) What is a branch target buffer? (2p)
b) What improvement are introduced in a folding branch-target-buffer? (2p)
c) What purpose does the scoreboard logic in CDC6600 have? (2p)
d) What is the name of the improved algorithm used instead of scoreboarding in IBM360/91 (2p)

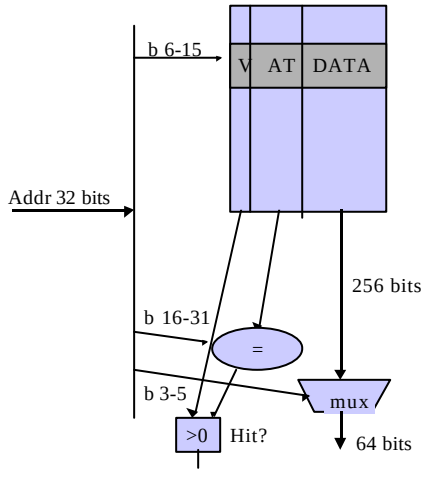
Problem 3, virtual caches (8p)

Most caches are so-called physical caches, rather than virtual caches, even though virtual caches could potentially have several advantages over physical caches

a) What is a virtual cache? (2p)
b) What is its potential advantage over a physical caches? (2p)
c) What is the inherent problem with a virtual cache? (2p)
d) Describe one technique used to overcome that problem (2p)

Problem 4, cache organization (8p)

The following picture shows a direct-mapped cache for 64byte large cachelines.
a) What is the cache size, i.e., what is its data capacity (2p)



b) Make your own drawing of one cache of twice that capacity, with 32 byte cache lines, and with a 4-way associative organization (4p)
c) What are conflict, compulsory, and capacity misses, and describe cache properties used to remove each category. (2p)

Problem 5, loop scheduling (8p)

Consider the loop and the corresponding compiler-generated code

```
for (i=1; i<=1000; i = i+1)
    x[i] = x[i] + 10;
```

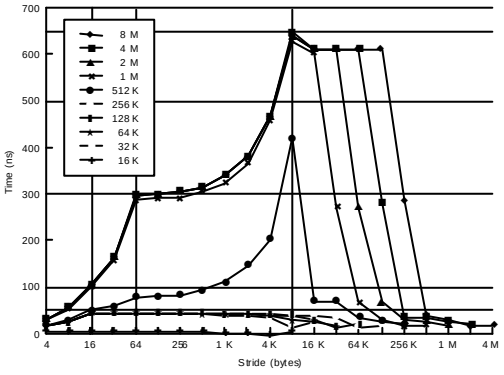
```
loop:   LD F0, 0(R1)        ;line 1
        ADDD F4, F0, F2     ;line 2
        SD 0(R1), F4        ;line 3
        SUBI R1, R1, #8     ;line 4
        BNEZ R1, loop       ;line 5
```

a) Write one-sentence comments for each line? (2p)
b) Show where the bubbles are in each iteration take, assuming an architecture with one cycle branch-delay slot and a load dealy of two cycles? (2p)
c) Show how the loop can be staticaly scheduled to improve the performance and calculate the number of cycles required for each iteration. (2p)
d) Show how loop unrolling could help to avoid all the stalls in this loop (2p)

Problem 5, microbenchmarks (8p)

A computer architecture has been characterized with using a simple microbenchmark according to the graph below.

```
for (times = 0; times < Max; time++) /* many times*/
  for (i=0; i < ArraySize; i = i + Stride)
    dummy = A[i]; /* touch an item in the array */
```



a) What is the cache line sizes for the L1 and L2 caches? (2p)
b) What is the access time to the memory? (2p)
c) The curve corresponding to the array size of 512k has one point much higher than its other points, still it is much lower than the corresponding points for Array sizes > 512k. Describe what is happening at that point. (2p)
d) Estimate the upper and lower bound for the number of TLB entries in this architecture (2p)

Problem 6, Storage systems (8p)

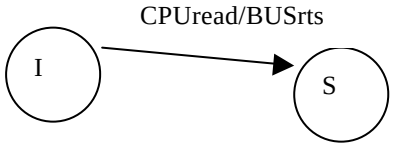
a) A disk is marked with an average seek time of 8ms, it rotates at 7200RPM and has a data transfer rate of 5Mbyte/s. What is the average time for accessing a file of 8kbytes? (2p)
b) Describe how RAID3 techniques can be used to improve the availability of data stored in a disk array (2p)
c) Describe how RAID5 techniques can be used to improve the "small writes" to a disk array, compared with a RAID 3 system,
d) Describe the the disk reads, disk writes and bit-logical operations for small writes in a RAID 5 system (2p)

Problem 7, Cache Coherence (12p)

a) Draw a state transition diagram of a write invalidate MOSI protocol showing state transitions forced by the CPU operations: CPUread, CPUwrite and CPUrepl (replacement). If there is no state change, show that as a loop-back to the same state. Each state transition should be clearly marked with "CPU input signel/BUS output signal" (if any). There are four different BUS transactions in this system

BUSrts: ReadtoShare (reading the data with the intention to read it)
BUSrtw, ReadToWrite (reading the data with the intention to modify it)
BUSwb: Writing data back to memory
BUSinv: Invalidating other caches copies

(The partial example below shows that a CPUread transaction forces a BUSrts transaction BUSrts to be sent, and the state to change from I to S.) (4p)



b) Show in a new drawing how external the BUS transactions

BUSrts: ReadtoShare (reading the data with the intention to read it)
BUSrtw, ReadToWrite (reading the data with the intention to modify it)
BUSwb: Writing data back to memory
BUSinv: Invalidating other caches copies

forces state changes in the MOSI protocol (4p)

C) Compulsory, Capacity and Conflict misses have got a new companion miss category (starting on the same letter "C") in a shared memory system. What is its name and how does it occur? (2p)
D) What is false sharing? (2p)

Problem 8, Synchronization (8p)

The CPUs of a shared-memory multiprocessor system can perform an atomic test&set operation TAS R, M, that atomically switches the contents of memory location A and register R.

a) Show how pseudo C-code for the **lock** and **unlock** primitives with a single input variable ADDR. They should produce the smallest possible amount of BUS traffic, assuming a heavily contended lock variable. (4p)
b) What is a barrier synchronization? (2p)
c) Write the pseudo C-code for a barrier using the lock/unlock primitives you (hopefully) completed under item a. The algorithm should minimize BUS traffic (2p)

Also: Exercises similar to the assignments selected from the book are good candidates for exam questions. At least two such questions will be included in the exam. You will automatically get maximum score for one, or both of them, if you have complete the home assignment (including selected exercises) before noon Dec 14th.