George Mason University **Final Exam**
CS 365 – Computer Architecture (Section 2) **May 7,1996**

Important Notes:

1. You have 150 minutes for answering the following questions. There are two parts to the exam. Part A has 20 short questions each worth 2 points. Part B has 3 questions each worth 20 points. There is an extra credit question in part B worth 10 points. The total number of points is thus 110. The points for a question are shown in bold at the end of each question.

2. Be concise and accurate in your answers. If you don't know the answer to a question, the best strategy is to move on, and return to the question later. The estimated time to answer Part A is one hour. The estimated time for Part B is also one hour. You have an extra half hour, for good measure.

3. Please answer the questions in the space provided below each question. If you need additional space, use the blank pages at the end.

PART A

1. What is the relationship between a silicon ingot, a wafer, a die, and a chip? **(2)**

2. In 2's complement notation, what are the largest and smallest integers that can be stored in 32 bits? (Your answers can be in the form of an arithmetic expression, i.e., you don't need to actually calculate the number). **(2)**

3. Suppose that the clock rate of your favorite microprocessor was *doubled* by the manufacturer. Does this imply that the *CPU time* of your favorite program will be reduced by 50% on the new microprocessor? If so, under what conditions? If not, why not? **(2)**

4. If the *CPU time* of your favorite program was reduced by 50% (in the previous question) does that imply that the *execution time* of the program was reduced by 50%? Explain briefly. **(2)**

5. What is the difference between double precision and single precision floating point numbers? **(2)**

6. What is the advantage of having a standard (such as the IEEE 754 standard) for representing floating

point numbers? **(2)**

7. What is the advantage of interrupt-driven data transfer between the CPU and a I/O device (as opposed to polling)? **(2)**

8. Are CPU caches constructed from static RAM (SRAM) or dynamic RAM (DRAMs)? What is the typical time for accessing SRAM? for accessing DRAM? (10 nanoseconds, 100 nanoseconds, 1 microsecond, 10 microseconds, 100 microseconds, 1 millisecond, 10 milliseconds, 100 milliseconds, 1 second?) **(2)**

9. Consider a microprocessor with a 32-bit address. What is the largest possible size of the address space of a program executing on this processor? What are the *logical segments* into which the address space

of any executing program is divided? **(2)**

10. What is the difference between asynchronous and synchronous buses? **(2)**

11. What is the advantage of creating standards for buses, e.g., SCSI, NuBus, etc.? **(2)**

12. What is a backplane bus? **(2)**

13. What is the average disk access time of a typical hard disk? (10 nanoseconds, 100 nanoseconds, 1

microsecond, 10 microseconds, 100 microseconds, 1 millisecond, 10 milliseconds, 100 milliseconds, 1 second?) **(2)**

14. Suppose you went out and bought a PC with the latest and greatest Pentium chip, a nice 17-inch color monitor, 32 MB of RAM, a 1 GB hard disk, a floppy drive, and a CDROM drive, and a 28.8 Kbps modem. Out of all these components which do you think would be the three most expensive ones? **(2)**

15. What is the function of a translation lookaside buffer? **(2)**

16. Why is the function of the following registers in the MIPS architecture: **(2)**

- Register 31.
- Register 0.
- the Hi and Lo registers.
- the EPC register.

- the Cause register.

17. What are the 5 steps into which the execution of an instruction (in the MIPS implementation of Ch 5) can be broken up? **(2)**

18. What is the basic idea behind pipelining? (No essays please!) **(2)**

19. What are the advantages and disadvantages of increasing the associativity of a CPU cache? **(2)**

20. Why do programs written in languages such as C, Pascal, and C++, need a stack in memory in order to execute? **(2)**

EXTRA SPACE FOR ANSWERS TO QUESTIONS IN PART A

PART B

I. { **60 minutes**} Page 9 reproduces the data-path and control-unit for the multiple cycle implementation of a processor that supports a subset of the MIPS instruction set, specifically the arithmetic-logical instructions, **lw**, **sw**, **beq**, and **j**. Page 10 contains a table that explains the actions caused by the setting of each control signal. Page 11 contains the finite state machine control for the implementation. Finally, a figure with the 3 instruction formats found in the MIPS architecture is reproduced below.

Based on this information, answer the following questions:

A. We wish to add the instruction **jal** (jump and link) to our instruction set. The format and meaning of this instruction is shown below.

Instruction	Format	Example	Meaning
jal	J	jal 10000	\$31 = PC+4; goto 10000

Add any necessary data-paths and control signals to the Figure on Page 9 for these instructions. Add any new control signals to the table on Page 10 (explain what happens when the signal is asserted or deasserted). Describe the additions you made in the space below. **(10)**

B. Show below the additions to the finite state machine of page 11 needed to implement the **jal** instruction. **(10)**

II. { **30 minutes** } Shown below is the cache for the DEC 3100. This is a direct mapped cache (set associativity = 1) of size 64 KB with a block size of 1 word.

Suppose you increase the size of the cache to 128 KB and reorganize it so that the block size becomes 4 words. Show the organization of the redesigned cache in the space below. **(20)**

III. { **30 minutes**} Consider a microprocessor with a 32 bit logical and physical addresses, and a paged virtual memory with a page size of 4 Kbytes. Suppose that pages that are not in physical memory are stored on a hard disk that can transfer data at the rate of 4 MBytes/second. Assume that the average seek time for a disk access is 10 ms and the average rotational latency is also 10 ms.

Suppose you were designing hardware and software for interfacing the disk with a 50 MHz processor. You are also given the following information:

1. The overhead for handling an interrupt to transfer 1 byte between the disk and the processor is 100 clock cycles.
2. The initial overhead for setting up the DMA transfer is 1000 clock cycles and the interrupt handling on DMA completion takes 500 clock cycles.

Based on the information above, find the average CPU overhead for loading a logical page from disk into memory using (a) interrupt-driven data transfer, and (b) DMA. Take the seek time and rotational latency into account in your calculations, but ignore any impact from factors not mentioned in this question, e.g., bus contention.) **(20)**

BONUS question: What are the advantages and disadvantages of increasing the page size to 8 Kbytes? **(10)**