Department of Electrical and Computer Engineering

The University of Texas at Austin

Yale N. Patt, Instructor

TAs: Siavash Zangeneh, Ben Lin, Juan Paez

Course Outline

January 20, 2021

January 20: Lecture 1: Intro to the course. Levels of Transformations. Basic architectural choices. Science of Tradeoffs.

January 21,22: First Discussion Session. Bookkeeping, overview, expectations, PL0, Intro/Focus

> Programming Lab 0 is due, Sunday night, January 24, 11:59pm. (The program itself will be empty. The purpose of the assignment is to make sure we are on the same page re: using the system.)

January 25: Lecture 2: Intro/focus, continued. Tradeoffs, Latency and Bandwidth, Role of Parallelism, Role of Speculation, Overview of Quantitative Insights, Role of the Architect, Alternative Design Points.

January 27: Lecture 3: Intro to Instruction Set Architecture, with examples taken from many diverse ISAs. Difference between ISA and Microarchitecture, ISA tradeoffs. Detailed discussion of LC-3b, with Assembly language constructs. The Assembler, how it works.

January 28,29: Discussion Session. The ISA, Assembly Process, PL1.

> Problem set 1 due before class, February 1. (Emphasis: ISA, the Assembly Process.)

February 1: Lecture 4: Microarchitecture, LC-3b data path, state machine, microsequencer, two-level microprogramming, Wilkes' Diode Matrix, Choice of ASICs, FPGAS, EMT instruction for enhanced performance

February 3: Lecture 5: Microarchitecture, continued. Microarchitecture Tradeoffs, Enhancing Performance. Pipelining, and its implications. Scoreboarding and its limitations.

February 4,5: Discussion Session. Microarchitecture of the LC-3b, PL1.

> Programming Lab 1 is due, Sunday night, February 7, 11:59pm. Write a program (Program A) in LC-3b assembly language to solve a problem. Write an assembler to translate LC-3b assembly language code into the instructions in the LC-3b ISA. Assemble Program A with the Assembler you wrote.

February 8: Lecture 6: Microarchitecture, continued. (Out of order Execution, The Tomasulo Algorithm why it did not work, and how we fixed it).

February 10, Lecture 7: Branch Prediction and other performance enhancements.

February 11,12: Discussion session: Pipelining, Scoreboard, Tomasulo, Branch Prediction, PL2.

> Programming Lab 2 is due, Sunday night, February 14, 11:59pm. (Write a program in C that simulates at the instruction cycle level the baseline LC-3b ISA. Test your simulator with the output of the assembler for the application program written in Programming Lab 1.)

February 15: Lecture 8: Physical Memory. SRAM, DRAM, NVM. Interleaving, Unaligned accesses.

> Problem set 2 due before class, February 17. (Emphasis: uarch of the LC-3b, the Assembly Process, Pipelining Branch Prediction, Tomasulo)

February 17, Lecture 9. Physical Memory, continued.

February 18,19: Discussion Session. Physical Memory, Prepare for first midterm.

> Problem set 3 due before class on February 22. (Emphasis on Physical Memory)

February 22, Lecture 10: Review for Exam 1.

February 24: Lecture 11: Exam 1

February 25,26: Discussion Session. Go over the exam, PL3

March 1, Lecture 12: Virtual memory, page tables, TLB, VAX model, IA32 model, Translation and Protection, contrast with segmentation.

March 3, Lecture 13: Virtual Memory, continued.

March 4,5: Discussion Session. Virtual memory, PL3.

> Programming Lab 3 is due, Sunday night March 7, 11:59pm. (Finish the clock-cycle level Simulator for the LC-3b. Test it on the application program written in Programming Lab 1.)

March 8, Lecture 14: Virtual Memory, continued.

March 10, Lecture 15: The notion of Process, interaction with the O/S, Exceptions and Interrupts.

March 11,12: Discussion session: Virtual Memory, process, interrupts and exceptions.

March 15,19: Spring break.

March 22: Lecture 16: Cache memory.

March 24, Lecture 17: Cache memory, continued.

March 25,26: Discussion session: Cache Memory, PL4.

Problem set 4 due before class, March 29. (Virtual Memory, Cache memory, Process, interrupts/exceptions)

March 29, Lecture 18: Input/Output. Asynchronous/Synchronous, Arbitration, Transaction.

March 31, Lecture 19: Input/Output, continued.

April 1,2: Discussion session: I/O, PL4

Programming Lab 4 is due, Sunday night April 4, 11:59pm. (Interrupts/Exceptions)

NOTE: April 5 is the last day to drop a class, change to CR/NC for academic reasons.

April 5: Lecture 20: Integer Arithmetic (Long integers, BCD, Kogge-Stone adders, Boothe's Algorithm, Residue Arithmetic)

April 7: Lecture 21: Floating point Arithmetic (Instruction formats, Gradual underflow, Rounding modes, NaNs, Floating Point Exceptions)

April 8,9: Discussion Session: Integer Arithmetic, Floating point arithmetic, Prepare for Exam 2.

Problem set 5 due before class, April 12. (Emphasis on I/O, Arithmetic)

April 12: Lecture 22: Review.

April 14: Lecture 23: Exam 2.

April 15,16: Discussion Session: Go over Exam 2. PL5.

April 19: Lecture 24: Single-thread parallelism. SIMD, Vectors, VLIW vs Wide Issue, DAE.

April 21: Lecture 25: Single-thread parallelism, continued. (HPS, Data Flow).

April 22,23: Discussion Session. Single Thread Parallelism, PL5

Programming Lab 5 is due, Sunday night April 25, 11:59pm. (Virtual memory)

April 26: Lecture 26: Intro to Multiprocessing, Amdahl's Law, Speed-up, efficiency, Interconnection networks, Cache Coherency, Memory Consistency.

April 28, Lecture 27: Multiprocessing, continued.

April 29,30. Discussion session: Multiprocessing, PL6

May 3: Lecture 28: Pot Pourri -- Selected topics from Measurement methodology, GPUs, Spatial computing, RISC, Intro Intellectual Property.

May 5: Lecture 29: Last class, free for all!

May 6,7: Discussion session. Review of the course, Prepare for Final exam.

Programming Lab 6 is due, Friday afternoon, May 7, 5pm. (Pipelining)

Problem set 6. To be used as part of your study guide for the final exam, not to be turned in.

Final exam: Probably Friday, May 14, 7-10pm.

Please note: The Registrar has the right to change the dates of the final exams. Please keep checking the Registrar's web site and our announcements to be sure when the final exam will be given.