Course: (EL – 426) Digital System Architecture (Processor Architecture)

In this course both Electronics and CS/IT background students are welcome. This course is designed for MTech and senior BTech students.

Textbook:

1. Computer Organization and Design: The hardware/software Interface (ARM edition) by John L Hennessy & David A Patterson (This book will be referred to as COD)
2. Computer Architecture: A Quantitative Approach, 5th Edition By John L Hennessy & David A Patterson (you may use earlier editions. It may have different Chapter numbers) (CAQA)
3. ARM Assembly Language: Fundamentals and Techniques by William Hohl, (AAL)

IMPORTANT NOTE: Lecture notes are for my convenience. It may not contain all the topics which I discuss in the class. You are supposed to attend the classes regularly and not depend upon the lecture notes

Course Philosophy: This course is meant to be a "hands on" course. You may have seen some of the material in IT 209. This course will extend the understanding and will introduce a design component it to it. At the end of the course, student will acquire knowledge about Processor design through verilog and C. For students interested in electronics, Cadence tool flow for both front end /back end design will be taught in the lab. For CS/IT students, MPI, OpenMP, MARE based energy efficient programming will be taught.

ARM based board level programming, Parallel programming using MPI, OpenMP, MARE and openCL (?) and Cadence design flow.

Course Outline:

ARM Architecture (AAL): Chapters 2,3,4,5,6,7,8,10,11,14

Chapter 4 (COD): The Processor

4.1 Introduction

4.2 Logic Design Convention

4.3 Building a Datapath
4.4 , 4.5 An overview of Pipelining
4.6 Pipelined Datapath and Control
4.7 Data dependency and hazard
4.8 Control hazard and Structural Hazard
4.9 Exceptions

Chapter 1 (CAQA)
1.1/.2/.3 Defining Computer Architecture
1.4 Trends in technology
1.8 Measuring, reporting and summarizing performance

Chapter 3 (CAQA)

3.1 Instruction Level Parallelism (ILP) Concepts and Challenges
3.2 Basic compiler techniques for exposing ILP
3.3 Reducing Branch costs with advanced branch prediction techniques
3.4 Overcoming Data hazards with dynamic scheduling
3.5 Dynamic scheduling: examples
3.6 Hardware based speculation
3.7 Exploiting ILP using multiple Issue and static scheduling (Super-scalar)
3.8 ILP using dynamic scheduling, multiple issue and speculation
3.9 Limitations of ILP

Chapter 4 (CAQA) Data-level parallelism in Vector, SIMD and GPU architecture
4.1, 4.2, 4.3 Introduction to Vector, SIMD and GPU architecture

Appendix B (CAQA) Review of Memory Hierarchy

B.1 Introduction
B.2 Cache performance
B.3 Six basic cache optimization
B.4 Virtual Memory

Chapter 5 (CAQA) (Thread-Level Parallelism TLP) / Issues in Multicore processors

5.1 Introduction
5.2 Centralized shared-memory architectures
5.3 Performance of Symmetric shared memory multiprocessors
5.6 Models of memory consistency
5.4 Distributed Shared memory and directory based coherence
5.5 Synchronization: basics

Grading Policy

Midterm:	30%-40%
Project:	30%
HW/Quiz (TBD):	0- 10%
Final exam:	30%

Plagiarism Policy: Students are not allowed to copy HW, Labwork, Quiz or represent someone's work as their own. Same is true for copying in the exams. Anyone caught cheating (and a person or persons helping him/her to cheat) will get penalty of a full grade (A A to BB or AB to BC etc). This rule will be strictly enforced.

Attendance Policy: 80% attendance is compulsory. Attendance may be taken at irregular intervals. You are supposed to be present in minimum 80% of those days. Anyone with less than 80% attendance will not be allowed to give exam.