Neuromorphic Engineering EEBM E6091, Spring 2012

Instructor
Dr. Bipin Rajendran
Research Staff Member, IBM T. J. Watson Research Center, Yorktown Heights, NY.

Course Information

The goal of Neuromorphic Engineering is to develop power efficient hardware technologies to process real-world noisy data, using techniques employed by the brain. Specifically, such systems are inspired by the functioning of neurons and synpases, their interconnectivity, and algorithmic principles of plasticity and learning.

The aim of this course is to provide an overview of the current trends and methods in Neuromorphic Engineering. This course will begin with a concise description of biological neural circuits including mechanisms of signal encoding, plasticity, and network architecture. Building on this, we will discuss the fundamental concepts and current trends in designing neuromorphic devices, circuits, and systems.

Course prerequisites include basic knowledge/familiarity with semiconductor devices and circuits, and probability and statistics. This course is intended for intermediate/advanced graduate students.

Course Outline

Jan 18 (week 1): Introduction to Neuromorphic Engineering
Jan 25 (week 2): Biological neurons and neuron models
Feb 1 (week 3): Synapses and plasticity rules, biological neural circuits
Feb 8 (weeks 4): Information processing in biology
Feb 15 (weeks 5): Neuromorphic design principles
Feb 22 (week 6): FETs - device physics and sub-threshold circuits
Feb 29 (week 7): Electronic neurons
Mar 7 (week 8): Semiconductor memory devices
Mar 14 (week 9): Spring Break
Mar 21 (week 10): Electronic synapses
Mar 28 (week 11): Project proposals - presentation and discussions
Apr 4 (week 12): Literature review: Analog neuromorphic VLSI
Apr 11 (week 13): Literature review: Digital neuromorphic VLSI
Apr 18 (week 14): Literature review: Novel technologies for Neuromorphic Engineering
Apr 25 (week 15): Literature review: Neuromorphic chips
May 2, 9 (weeks 16, 17): Project presentations

Grading

Class evaluation will be based on 3 programming assignments, participation in literature review and a final project.

The relative weightage will be :
Programming assignments (45%)
Participation in literature review (20%)
Final project (35%).

Students may choose a topic for the final project after discussing with the instructor. Team collaboration is encouraged. Project evaluation will be based on a short report and oral presentation in the class.