Final Program Exam  
Master of Science in Computer Engineering  
(Effective Spring 2016)  

General Information:  
If you are enrolled in the non-thesis Master of Science in Computer Engineering (MSCpE) degree program, you must pass the “Final Program Examination for the Master’s Degree.” To pass the exam and receive a degree, all regular, proxy, and field-appointed members of the examining committee must assent that the exam was passed.

A non-thesis candidate must successfully pass a final program examination that covers breadth within computer engineering degree program. This examination includes topics from the foundations of computer engineering, computer systems, networks, hardware, and software. The final program exam for MSCpE will have emphasis on the material related to the following graduate courses:

- ECE 5534 – Computer Networks 1
- ECE 5520 – Computer Architecture
- ECE 5565 – Embedded and Real-Time Systems

Exam Policies and Procedures:
- The exam is taken in a written form and has duration of 3 hours.
- The exam is administered two times per academic year (in fall and spring semesters).
- The exams are in a closed-book, closed-notes form with calculator allowed.
- For successful completion of MSCpE requirements, students must pass the exam which includes the sections listed below.
- There is a maximum of three attempts. If the student does not pass the exam in three attempts, he/she will not be able to graduate with MSCpE degree. (see Graduate Policy 1.6.5)
- The exam is developed with questions that will require a variety of approaches and methodologies, including design, analysis, and application.
- A student must be enrolled during the term the exam is taken or retaken in the case of a failed exam, else there is an examination fee. (see Graduate Policy 1.6.4)
- No notes or study materials are allowed.
- No cell phones allowed.
- No other unspecified electronic devices.

Exam Knowledge Area(s):
I. Computer Systems
   A. Numeric and Non-numeric Formats
      1. Number representation
      2. Character representation
      3. Encoding schemes
4. Error detection and correction

B. Computer Architecture
   1. Computer organization
   2. Processor structures
   3. Memory systems
   4. Hardware fault tolerance
   5. System performance metrics
   6. Instruction-set architecture
   7. Processor pipeline and advanced micro-architecture issues

II. Networks
   A. Computer Networks
      1. Protocols
      2. OSI (layer) model
      3. Topology
      4. Hardware
      5. Security
      6. Switching: Circuit/Packet
   B. Physical Layer Implementation
      1. Synchronization techniques
      2. Transmission media
      3. Asynchronous
      4. Wireless
      5. Optical
   C. Information Theory
      1. Data compression
      2. Channel capacity
      3. Sampling theory

III. Hardware/Software
   A. Hardware Description Languages
      1. Analysis and design
      2. Synthesis issues
      3. Assertions and verification
   B. System Software
      1. Operating systems
      2. Real-time operating systems
      3. Computer security
      4. Device drivers
      5. Interrupts
   C. Development / Applications
      1. Programming language characteristics
      2. Software design and documentation
      3. Algorithm design and data structures
      4. Asymptotic algorithm analysis

Suggested Reference Material:
- Computer Networks: A Systems Approach, by Peterson and Davie
- Computer Networking: A Top-Down Approach by Kurose and Ross
- Computer Networks by Tanebaum and Wetherall
- Computer Architecture: A Quantitative Approach by Hennessy and Patterson
- Computer Organization and Design “The Hardware/Software Interface” by Patterson and Hennessy
- Introduction to Algorithms by Cormen, Leiserson, Rivest, Stein
- Data Structures and Algorithms
- Topics in Hardware/Software Design
- Topic in Hardware/Software Integration