1. **Course number and name**
   EE 234: Microprocessor Systems

2. **Credits and contact hours**
   4 (3 lecture hours and 3 lab hours per week)

3. **Instructor's or course coordinator's name**
   Clint Cole

4. **Text book, title, author, and year**
   ARM Assembly Language Programming and Architecture (2nd ed.). Muhammad Ali Mazidi. 2016.
   Project Materials: (design projects, exercises, reading): <www.realdigital.org>.
   Other supplemental materials
   Blackboard circuit board (*Real Digital*); Vivado Webpack (*Xilinx*); Instructor notes and slides for some topics.

5. **Specific course information**
   a. *Catalog description:* Microprocessor system architecture, instruction sets, and interfacing; assembly language programming
   b. *Prerequisites or co-requisites:* CptS 122 with a C or better; EE 214 with a C or better.

1. **Specific goals for the course**
   At the end of this course, students must be able to:
   - Understand computer system architecture, including memory systems and peripheral systems (1,2)
   - Understand Microprocessor architecture, including instruction formatting and execution timing (1,2,6)
   - Understand interrupt systems, their use, and programming (1,2,6)
   - Know and use the ARM assembly instruction set (2)
   - Understand serial communications and protocols, and how and when to use UART, I2C, and SPI busses (2,6)
   - Know when to use interrupts and when to use polling (2)
   - Understand system timers and their use (2)
   - Be able to construct an interrupt-based software system to periodically acquire data, store it, and send/receive data over a serial port (1,2)
   - Be able to write efficient C and/or Assembly programs to access on-board peripherals and perform general computations (1,2,6)
   - Be able to present technical design details to peers and instructional personnel (3)

7. **Brief list of topics to be covered**
   - Review of digital circuits, with emphasis on arithmetic circuits and ALUs,
   - Overview of microprocessors and microcontrollers,
   - Microprocessor architectures and block diagrams,

- Microprocessor instructions, controller, and execution environment,
- Real-time environments and the need for interrupts,
- Interrupt handling,
- Timer/counter modules and their use,
- AXI bus overview, signaling protocols, and timing/control,
- Use of custom IP blocks,
- Serial busses: UART, I2C, SPI and USB,
- Memory interfaces,
- PWM and PDM signals and their use,
- Embedded software system design and partitioning.