

ABET Final Assessment Quiz

Note: According to new ABET requirements, you must achieve a score of 100% on this quiz in order to receive a passing grade in the class.

Course Outcomes Based on ABET Course Description:

According to the university's ABET description for this course, all students who pass the class will:

1. Understand how to make embedded software structured and maintainable
2. Be able to describe I/O interface techniques including:
 - Important issues in connecting sensors, switches, actuators, motors, etc. to a microcontroller
 - At least three methods of synchronizing software to I/O events
 - Sequence of operations in the interrupt handling of I/O events
3. Be able to describe aspects of a real-time operating system including:
 - Why multi-tasking is important for real time applications
 - At least two methods of task scheduling
 - At least three examples of operating system services
 - At least two different ways of communicating between multiple threads of execution
4. Be able to explain the effects of C versus assembly programming on code execution time and size.
5. Have hands-on experience with at least one high-level C compiler for a microcontroller
6. Have experience working on a team.
7. Have experience in documenting and explaining their work.

The assessment quiz will cover outcomes 1 – 3.

Sample Questions for the Quiz:

1. Give two principles (there are many) to follow in making embedded software structured and maintainable.
2. Show a basic circuit (don't need specific part numbers or component values) used to drive a relay with an 'HC11.
3. List three methods of synchronizing software to I/O events.
4. Give the sequence of operations required to handle an interrupt.
5. Explain what is meant by the term "context switch".
6. Why is multi-tasking (multiple threads of execution, including ISRs) important for real-time systems?
7. Give two methods of task scheduling in a multi-tasking system.
8. Give three examples of system services performed by the operating system.
9. Give two different ways of communicating between multiple threads of execution (e.g. between a main loop and an ISR).