

Programación de Sistemas Embebidos (Electiva)

La materia tiene como finalidad la programación de computadoras embebidas con el fin de desarrollar sistemas para robótica, automatización industrial, o para lograr una solución creativa (ej: IOT, cafetera wireless, videoconsola portable, etc).

FUNDAMENTACIÓN DE LA MATERIA

Un sistema embebido es una combinación de hardware y software fabricado para realizar una única función. Diariamente interactuamos con miles de ellos, y muchos otros sistemas embebidos realizan funciones de control industriales o cotidianas sin intervención humana. Por ejemplo, la industria automotriz incorpora decenas de sistemas embebidos: el sistema de frenos antibloqueantes, el monitoreo y control de emisión del vehículo, el sistema de inyección electrónica, el sistema de airbag, o la pantalla que resume la información para presentar un estado general al conductor. Ejemplos similares se encuentran en todos los ámbitos, incluso en aplicaciones críticas, como las aeroespaciales, industriales y médicas. Usualmente, el software embebido se desarrolla para ser ejecutado en hardware con recursos limitados.

Una subclase de estos sistemas son los sistemas de tiempo real (utilizados mayormente en aplicaciones críticas.) Estos sistemas no sólo deben computar resultados correctamente, sino que deben hacerlo en un lapso de tiempo menor a tiempos de respuesta límites definidos.

Por lo tanto, las reglas que rigen el desarrollo de estos sistemas son diferentes a las reglas para desarrollar software de aplicación general. La enseñanza y el desarrollo de competencias sobre la programación de sistemas embebidos son importantes debido a que existe una demanda creciente de profesionales con estas aptitudes. Esta demanda proviene de la revolución industrial 4.0, las ciudades y hogares inteligentes, y los ámbitos de investigación y desarrollo tecnológico vinculados a ellos.

OBJETIVOS DE LA MATERIA

Lograr que el alumno adquiera los conocimientos básicos del funcionamiento de los sistemas embebidos; y las habilidades necesarias para la programación inicial de hardware sin sistema operativo hasta sistemas embebidos con gestión de tareas.

Entender el Ciclo de Compilación de software, y el uso individual de los componentes de la Cadena de Herramientas para la traducción del código fuente para su ejecución en hardware específico, y la optimización del código para mejorar su rendimiento o tamaño. Adquirir los conocimientos de cómo automatizar el proceso de compilación con el objetivo de incrementar el control del proceso, y reducir la posibilidad de errores humanos en proyectos complejos. Adquirir las habilidades necesarias para trabajar con un sistema de control de versiones en equipo, sobre un repositorio de manera descentralizada.

Comprender los conceptos y metodologías que permiten diseñar y desarrollar drivers de E/S de calidad industrial. Esto incluye conocer cómo establecer interfaces eficientes y seguras, entre las distintas capas de hardware y software, y las estructuras de datos adecuadas para controlar el hardware de bajo nivel. Conocer los estándares de programación en C para sistemas críticos.

Adquirir los conceptos que rigen la programación de sistemas concurrentes y de tiempo real, y el uso de Núcleos de Tiempo Real para desarrollar tales sistemas, incluyendo la solución a problemas de sincronización, inversión de prioridades e interbloqueos.

Interesados contactar al docente: Rafael Ignacio Zurita <rafa@fi.uncoma.edu.ar>
<https://se.fi.uncoma.edu.ar/pse/>