

Programa Regular

Programación en Tiempo Real

Modalidad de la Asignatura: Teórico-práctica.

Carga horaria: 5 hs.

Objetivos:

el estudiante tendrá los conocimientos para desarrollar y caracterizar los sistemas de tiempo real y los sistemas distribuidos de tiempo real, en particular a través de técnicas y herramientas avanzadas de programación para sistemas de tiempo real.

Contenidos:

Sistemas de Tiempo Real (STRs). Arquitectura de un STR. Herramientas de Análisis, Verificación y Diseño. Ingeniería de Software para STRs. Concurrencia. Sincronización. Aplicaciones. Control de Hardware. Drivers. Tratamiento de Señales. Sistemas Operativos orientados a aplicaciones en tiempo real.

Unidades temáticas:

Unidad I

Conceptos Básicos de Programación en C. Tipos de Datos. Estructuras de Datos. Estructuras de Control. Administración de Memoria. Manejadores de interrupciones. Metodología de diseño. Hardware first-Software first.

Unidad II

Definición de Sistemas de Tiempo Real. Arquitectura de un Sistema de Tiempo Real. Herramientas de Análisis, Verificación y Diseño. UML & STRs. HRT-HOOD (Hard Real-Time Hierarchical Object-Oriented Design). Taxonomía de las arquitecturas multicomputador, multiprocesador y multi-microprocesador.

Unidad III

Programación Concurrente. Procesos. Sincronización. Exclusion Mutua. Semaforos. Memoria Compartida. Comunicación entre procesadores y procesos. Modelos de sincronización por mensajes asincrónicos. RPC y Rendezvous.

Unidad IV

Aplicaciones. Arbitraje, detección de fallas, acceso al medio de comunicación. Modelos de buses. Tolerancia a fallas en arquitecturas distribuidas. Control de Hardware. Drivers. Tratamiento de Señales.

Bibliografía:

- Arzen, K. Real-Time Control Systems. Department of Automatic Control, Lund Institute of Technology, Lund. Año 2001.
- Burns, A.; Welling, A. Real-Time Systems and Programming Languages, Addison Wesley. Año 2009.
- Bianchi, E.; Dozio, L.; Mantegazza, P. RTAI, A Hard Real Time support for LINUX, Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano. Año 2001.
- Buttazzo, G. Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications. Springer; 3° ed. Año 2011.
- Gomaa, H. Designing concurrent, distributed and real-time applications with UML. Addison Wesley, ISBN 0-201-65793-7. Año 2000.

Propuesta didáctica: Las clases se desarrollarán en el Laboratorio de Informática. Se organizarán en modalidades teórico-prácticas con soporte de presentaciones digitales y prácticas en función de cada clase.

En las clases se presentan los contenidos teóricos y se van resolviendo en forma conjunta ejemplos que ayuden a comprender los nuevos conceptos introducidos.

La formación práctica está basada en la resolución de problemas tipo, de actividades experimentales y de problemas abiertos de ingeniería, cuyas resoluciones se realizan principalmente en las computadoras, utilizando aplicaciones de uso en la industria que permitan un contacto directo con las tecnologías actuales.

En cuanto a la formación experimental se realizarán las siguientes actividades:

- Introducir al estudiante al análisis de la problemática de entrada/salida: adquisición de datos, entradas y salidas digitales y analógicas por medio de simulación de arquitecturas.
- Introducir al estudiante al uso de sistemas operativos de tiempo real por medio de Live cds de linux-rt, con las herramientas instaladas necesarias (compiladores, herramientas de test, etc.).
- Introducir al estudiante al diseño de sistemas de tiempo real por medio de herramientas varias de software libre para desarrollo y simulación de microcontroladores.

En cuanto a los problemas abiertos de ingeniería, se realizarán trabajos relacionados a la implementación de una aplicación determinada, que haga uso de las herramientas de diseño de algoritmos y sistemas de programación de tiempo

real, integrando los conceptos de las unidades temáticas desarrolladas durante la cursada y en la asignatura Complejidad Temporal, Estructuras de Datos y Algoritmos. La realización de los trabajos implica la identificación de un problema dado y la solución del mismo, a partir de la aplicación de los conocimientos adquiridos hasta entonces, lo cual constituye la base formativa para que el estudiante adquiera las habilidades que le permitan encarar proyectos y diseños de ingeniería. Cada trabajo debe incluir un detalle de los problemas encontrados, las formas de solucionarlos, las condiciones de ejecución, formato de los datos de entrada e ideas o sugerencias para realizar una versión mejorada del mismo.

Los estudiantes deben realizar entregas de al menos un trabajo integrador durante el desarrollo de la cursada el cual será revisado por los docentes y devuelto a los estudiantes para su corrección.

El material correspondiente a las clases teóricas, así como los documentos de la práctica se encontrarán disponibles a través de un grupo Web al cual los estudiantes tendrán acceso. Este mecanismo también será utilizado para realizar consultas simples.

Actividades extra-áulicas: Se establecerán guías de actividades prácticas para que el estudiante pueda ejercitar, a fin de consolidar los conceptos aprendidos en clase.

Evaluación: La evaluación integradora de las instancias teórico-prácticas se realiza a través de un parcial teórico-práctico en máquina. Además, los estudiantes deberán desarrollar un trabajo final donde se integren los temas vistos en la materia. Las clases son obligatorias ya que implican participación y debate que forman parte de la evaluación.