

PROGRAMA DE ESTUDIOS: SISTEMAS EN TIEMPO REAL

PROTOCOLO

Fechas	Mes/año
Elaboración	05-2010
Aprobación	
Aplicación	09-2010

Clave			Semestre	8° a 10°		
Nivel	Licenciatura	X	Maestría		Doctorado	
Ciclo	Integración		Básico		Superior	X
Colegio	H. y C.S.		C. y T.	X	C. y H.	

Plan de estudios del que forma parte: Ingeniería de Software

Propósito(s) general(es):
 Que el estudiante desarrolle aplicaciones de sistemas en tiempo real con el uso de herramientas de desarrollo para sistemas de tiempo real, empleando estrategias y metodologías de la Ingeniería de Software.

Carácter		Modalidad				Horas de estudio semestral (16 semanas)					
Indispensable		Seminario		Taller		Con Docente	Teóricas	36	Autónomas	Teóricas	20
		Curso	X	Curso-taller			Prácticas	36		Prácticas	36
Optativa *	X	Laboratorio	X	Clínica		Carga horaria semanal: 4.5 +3.5= 8.0			Carga horaria semestral: 128		

Asignaturas Previas:	Asignaturas Posteriores:
Sistemas embebidos	Ninguna

Requerimientos para cursar la asignatura	Conocimientos y habilidades: Conocimientos de programación de sistemas, Arquitectura de computadoras y sistemas operativos.
---	--

Perfil deseable del profesor:	Ingeniero de Software, Licenciado en Ciencias Computacionales, Ingeniero en Computación o afín, preferentemente con postgrado y experiencia docente, de investigación o trabajo en el área.
--------------------------------------	---

Academia responsable del programa:	Diseñador (es):
Informática	Omar Nieto Crisóstomo

* Aquellas en las que se ofrece la posibilidad de cursar una de las asignaturas, para cubrir un requisito INDISPENSABLE será considerada INDISPENSABLE.

INTRODUCCIÓN

En las últimas dos décadas ha habido un gran crecimiento en las aplicaciones de la vida cotidiana que emplean sistemas embebidos y de tiempo real. Un sistema en tiempo real debe ser capaz de procesar tareas (muchas de las veces en paralelo) en un tiempo preciso, para ello necesita de hardware y software especializado para ejecutar en tiempo su actividad, así como, herramientas de desarrollo especializado para la programación, depuración, análisis, optimización y pruebas,

La materia de Sistemas en Tiempo Real, muestra a los estudiantes el proceso de desarrollos de aplicaciones de sistemas en tiempo real con el uso de herramientas de desarrollo específicas, empleando estrategias y metodologías de la Ingeniería de Software.

PROPÓSITOS GENERALES

Que el estudiante desarrolle aplicaciones de sistemas en tiempo real con el uso de herramientas de desarrollo para sistemas de tiempo real, empleando estrategias y metodologías de la Ingeniería de Software.

CONTENIDOS

TEMAS Y SUBTEMAS	PROPÓSITOS ESPECIFICOS
1. Introducción a los sistemas embebidos 1.1 Descripción general de un sistema en tiempo real. 1.2 Problemas en el diseño de los sistemas en tiempo real 1.3 Ejemplos de aplicaciones de sistemas en tiempo real	Que el estudiante identifique los sistemas de tiempo real aplicador en la vida cotidiana, y los problemas de diseño que se deben contemplar en los sistemas de tiempo-real.
2. Consideraciones de hardware 2.1 Arquitectura básica 2.2 Interfaces de hardware 2.3 Unidad de Procesamiento Central 2.4 Memoria 2.5 Entrada/Salida (puertos, DMA, Interrupciones) 2.6 Elementos para mejorar el desempeño (Caché, Pipeline, Coprocesadores) 2.7 Consideraciones de los ciclos de ejecución del CPU	Que el estudiante identifique y describa los elementos de hardware que conforman a un sistema de tiempo real y los periféricos con el que interactúa, así como, los elementos hardware para mejorar el desempeño.

<p>3. Sistema Operativo de Tiempo Real</p>	<p>Que el estudiante identifique y describa los elementos que proporciona un sistema operativo de tiempo real, para la programación de tareas, interrupciones de software e interrupciones de hardware, sincronizadas y comunicadas uso de memoria, con objetos del sistema operativo.</p>
<p>3.1 Kernel de tiempo real 3.2 Fundamentos teóricos de Sistemas operativos de tiempo real 3.3 Multitareas 3.4 Comunicación y sincronización entre tareas. 3.5 Gestión de memoria</p>	
<p>4. Requerimientos de Ingeniería de Software</p>	<p>Que el estudiante identifique y describa los requerimientos teóricos y metódicos de ingeniería de software aplicado a los sistemas en tiempo real.</p>
<p>4.1 Métodos formales en la especificación de software. 4.2 Máquinas de estado finito 4.3 Diseño de máquinas de estado finito 4.4 Opciones de implantación de máquinas de estado finito 4.5 UML para sistemas de tiempo-real</p>	
<p>5. Patrones de diseño para sistemas en tiempo real</p>	<p>Que el estudiante describa y aplique patrones de diseño en el diseño de sistemas de tiempo real.</p>
<p>5.1 Diseño de patrones en tiempo real 5.2 Patrones de concurrencia 5.3 Patrones de memoria 5.4 Patrones de recursos 5.5 Patrones de distribución</p>	
<p>6. Desarrollo de aplicaciones en tiempo real</p>	<p>Que el estudiante implante aplicaciones en tiempo real mediante herramientas de desarrollo para edición, compilación, depuración y prueba.</p>
<p>6.1 Herramientas de desarrollo para aplicaciones en tiempo real 6.2 Implantación de sistemas en tiempo real 6.3 Depuración y pruebas de sistema en tiempo real</p>	
<p>7. Análisis del desempeño y optimización</p>	<p>Que el estudiante identifique, describa y aplique mecanismos de optimización en sistemas de tiempo real.</p>
<p>7.1 Antecedentes teóricos 7.2 Análisis del desempeño 7.3 Aplicación de la teoría de colas 7.4 Desempeño de entrada/salida 7.5 Optimización del rendimiento 7.6 Optimización de los resultados del compilador 7.7 Análisis de requerimientos de memoria 7.8 Reducción de la memoria utilizada 7.9 Consideraciones de fallas, bugs y tolerancia a fallos</p>	

METODOLOGÍA PARA EL CURSO

La materia se impartirá con clases teóricas y prácticas de laboratorio. El curso se llevará a cabo mediante exposiciones orales, resolución de ejercicios, trabajos en equipo, trabajos de investigación, asesorías, prácticas de laboratorio en donde se programe la arquitectura de un sistema en tiempo real. Es preferente que el estudiante realice un proyecto final en donde desarrolle una aplicación de un sistema en tiempo real.

EVALUACIÓN DIAGNÓSTICA

Se aplicará un examen escrito donde se indague los conocimientos de programación, sistemas operativos y arquitecturas de computadoras. Esta evaluación le permite al docente, identificar los conocimientos, habilidades y deficiencias que poseen los estudiantes, de tal manera que, pueda realizar acciones que sean orientadas al mejor aprovechamiento del estudiante en la materia, en base a los resultados de la evaluación, el profesor podrá realizar repases, programar asesorías, proponer investigaciones en bibliografía, así como, identificar metodología convenientes en la enseñanza y el aprendizaje.

EVALUACIÓN FORMATIVA

Se recomienda al docente que aplique mínimamente tres evaluaciones formativas a lo largo del semestre, en donde el estudiante demuestre los conocimientos y habilidades adquiridas durante dicho periodo de evaluación. En la parte práctica, se recomienda que el profesor deje un trabajo de investigación previo a cada práctica, referente al tema a tratar, y que el estudiante entregue el reporte de la práctica. Además, al final del semestre, se recomienda que el estudiante entregue un trabajo escrito con el reporte técnico de la elaboración de un proyecto final.

EVALUACIÓN DE CERTIFICACIÓN

El examen de certificación evaluará los conocimientos y habilidades del estudiante adquiridos durante el curso. Este instrumento consta de dos partes: una evaluación teórica, y una evaluación práctica. La evaluación teórica consiste en la elaboración de un examen escrito en donde se evaluarán los conceptos, definiciones, las técnicas en el desarrollo de sistemas en tiempo real. La segunda parte consta de un proyecto, donde el estudiante desarrolle una aplicación que resuelva un problema basados en sistemas en tiempo real, el estudiante entregará un reporte técnico que describa los detalles en todo del proceso de desarrollo de software de su proyecto, también entregará los archivos fuente y ejecutable en un dispositivo de almacenamiento.

BIBLIOGRAFÍA

La bibliografía que se menciona a continuación es la básica para el estudiante y el profesor:

Bibliografía	Temas para los que se recomienda
1. Laplante, Phillip A, Real-Time Systems Design and Analysis . 3er Edition. IEEE Press, John Wiley & Sons, 2004	1, 2, 3, 4, 6 y 7
2. Williams Rob. Real-Time Systems Development . Elsevier. 2006.	1, 2, 3, 4, 6 y 7
3. Cheng, Albert M. K. Real-Time Systems, Scheduling, Analysis and Verification . , John Wiley & Sons, 2002	1, 2, 3, 4, 6 y 7
4. Powel Douglas, Bruce. Real-Time Design Patterns: Robust Scalable Architecture for Real-Time Systems , Addison Wesley, 2002	5
5. Lee, Insup. Leung, Joseph Y-T. Son, Sang H. Handbook of Real-Time and Embedded Systems , Chapman % Hall/CRC, 2008	4 y 5
6. Wellings Andy. Concurrent and Real-Time Programming in Java . John Wile & Sons, 2004	1, 2, 3, 4 y 6
7. Li Qing. Yao Carolyn. Real-Time Concepts for Embedded Systems . CMP Books, 2003	1, 2, 3, 4 y 6

OTROS RECURSOS

Proyector de video (Cañón), Computadora que tenga instalado un entorno de desarrollo integrado para sistemas de tiempo real. Se recomienda que se cuente con una arquitectura para la aplicación de un sistema en tiempo real, donde se practique lo visto en clase.