Ministerio de Cultura y Educación
Universidad Nacional de San Luis
Facultad de Ciencias Físico Matemáticas y Naturales
Departamento: Informatica
Area: Area II: Sistemas de Computacion

(Programa del año 2017)
(Programa en trámite de aprobación)
(Presentado el 27/06/2017 10:44:48)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
SISTEMAS EMBEBIDOS	ING. EN COMPUT.	28/12	2017	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
GROSSO, ALEJANDRO LEONARDO	Prof. Responsable	P.Asoc Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	**Teóricas**	**Prácticas de Aula**	**Práct. de lab/ camp/ Resid/ PIP, etc.**	**Total**
Hs	3 Hs	1 Hs	4 Hs	8 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1º Cuatrimestre

Duración			
Desde	**Hasta**	**Cantidad de Semanas**	**Cantidad de Horas**
13/03/2017	23/06/2017	15	120

IV - Fundamentación

Los sistemas embebidos están presentes en la mayoría de los dispositivos con los que interactuamos: electrodomésticos, automóviles, aviones, ruteadores, teléfonos móviles, etc.. En la mayoría de estos sistemas la ausencia de fallas y un tiempo de respuesta preciso son requerimientos indispensables para su funcionamiento. Por lo tanto es necesario utilizar, y en algunos casos desarrollar, herramientas adecuadas para la especifición, diseño y verificación del sistema embebido. En este curso introduciremos algunas herramientas que ayudan al desarrollo de los sistemas embebidos.

V - Objetivos / Resultados de Aprendizaje

Introducir métodos para el diseño de sistemas embebidos (Modelos discretos y analógicos).
Introducir métodos formales para la verificación de sistemas embebidos.
Utilizar los métodos anteriores para desarrollar sistemas concretos sobre plataformas concretas y comprobar su funcionamiento.
Conocer distintas plataformas utilizadas en el desarrollo de sistemas embebidos.

VI - Contenidos

Programa Analitico

Unidad 1: Sensores y Actuadores

Sensores: Rango,rango dinámico, quantización, ruido, muestreo, acondicionamiento de señal, sensores comunes:

medición de inclinación y aceleración, medición de posición y velocidad, medición de rotación, otros sensores. Actuadores: diodos emisores de luz, control de servos, parlantes, switches (llaves).

Unidad 2: Procesadores.

Introduccion Cortex-M3 de ARM: elementos básicos, registros, mapa de memoria, instrucciones, interrupciones. Introducción a Cortex-M4. Procesadores multicores simétricos y asimétricos.
Análisis de documentación de fabricantes. Plataforma de desarrollo. Introducción a un entorno de desarrollo para microcontroladores de 32 bits. Análisis del hardware empleado en el curso.

Unidad 3

Interfaces del microcontrolador utilizadas para la adquisición y almacenamiento de datos y para control y comunicación con otros microcontroladores y periféricos. Diseño de PCB(placas de circuitos impresos) utilizando herramientas CAD de licencia libre.

Unidad 4: Modelos reactivos y transformacionales.

Introduccion a los StateCharts (Diagramas de estados): estados, eventos, jerarquía de máquinas, superestados, subestados, estados OR, estados AND. Modelado utilizando StateCharts. Introducción de herramientas que generan código C a partir de una especificación que utiliza StateCharts.

Unidad 5: Sistemas operativos para sistemas embebidos

Predicción de los tiempos de ejecución. Planificación en sistemas de tiempo real. Clasificación de los algoritmos de planificación. Sistemas Operativos embebidos.

Unidad 6: Programación de microcontroladores de 32 bits en C.

Análisis en bajo nivel de la asignación de memoria. Estructura de programas reactivos sin sistemas operativos de tiempo real. Especificación de drivers. Generación de eventos. Estructura de sistemas embebidos que usan Sistemas Operativos de Tiempo Real.

Unidad 7; Introduccion a Codiseño de hardware/software y Chequeo de modelos (Model Checking).

Metodologías que no presuponen que partes de un sistema se implementan en hardware o software. Introducci'on al codiseño hardware software.
La necesidad de metodos formales. Verificación de Hardware y Sofware. El proceso de Model Checking. Logicas temporales. Expresando propiedades con formulas LTL (Linear Temporal Logic).

VII - Plan de Trabajos Prácticos

Todos los prácticos utilizan las placas disponibles en el laborario de sistemas embebidos.

Práctico N°1: Titilar un led. Utilizar el ambiente de desarrollo eclipse y la herramienta de diseño VisualSTATE.

Práctico N°2: Titilar un led definiendo el periodo. Utilizar el ambiente de desarrollo eclipse y la herramienta de diseño VisualSTATE.

Práctico N°3: Interactuar con los periféricos a través de las distintas interfaces de la placa para el LPC 1769.

Práctico N°3: Led con atenuación utilizando PWM (pulse-width modulation) a partir de la intensidad definida por un potenciómetro. Programación de Timers. Utilizar el ambiente de desarrollo eclipse y la herramienta de diseño VisualSTATE.

Práctico N°4: Debouncing de llaves (switches). Utilizar el ambiente de desarrollo eclipse y la herramienta de diseño VisualSTATE.

Práctico N°5: Implementar un proceso de pesado de precisión.

Práctico N°6: implementar un sistema de control de velocidad vehicular utilizando sensores magnéticos.

Práctico N°7: implementar un sistema de control de ascensor.

Práctico N°8: Comunicaciones en bus serie: SPI e I2C. Utilización de un magnetrómetro y un sevo motor angular para implementar una brújula. Desarrollar un programa que almacene los datos recibidos de un magnetómetro en una tarjeta SD

Práctica N°9: Dispositivo USB. Utilizar un teclado matricial para convertirlo en un keypad (bloque numérico).

VIII - Regimen de Aprobación

La asignatura es promocional. El alumno debe asistir al 80% de las clases teóricas y prácticas, aprobar con al menos el 70% todos los laboratorios y el parcial, aprobar con al menos el 70% un proyecto de mediana complejidad, aprobar un examen integrador con al menos el 70%. En caso de aprobar todas la evaluaciones el alumno promociona la materia. En caso de aprobar todas la evaluaciones con excepción del examen integrador el alumno obtiene la regularidad de la materia.

IX - Bibliografía Básica

[1] P. Marwedel; Embedded System Design; Springer; 2006
[2] D.E. Simon; An Embedded Software Primer; Addison-Wesley; 1999
[3] Q. Li, C. Yao; Real-Time Concepts for Embedded Systems; CMP Books; 2003
[4] J. Labrosse et. al.; Embedded Software: Know It All; Newnes; 2008
[5] T. Noergaard; Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers; Newnes; 2005
[6] J. Yiu. The Definitive Guide to the ARM® Cortex-M3 (Second Edition). Newnes. 2010.
[7] E. Lee, S. Seshia. Introduction to Embedded Systems A Cyber-Physical Systems Approach Second Edition. LeeSeshia.org, 2015.
[8] M. Samek. Practical UML Statecharts in C/C++ Second Edition. Newnes. 2009.
[9] D. HAREL. STATECHARTS: A VISUAL FORMALISM FOR COMPLEX SYSTEMS. Science of Computer Programming 8 (1987) 231-274. North-Holland

X - Bibliografia Complementaria

[1] S.C. McConnell; Code Complete: A Practical Handbook of Software Construction; Microsoft Press, 2nd ed., 2004

XI - Resumen de Objetivos

Introducir métodos para el diseño de sistemas embebidos.

Conocer distintas plataformas utilizadas en el desarrollo de sistemas embebidos.

XII - Resumen del Programa

Sensores y actuadores. Procesadores. Modelos reactivos. Sistemas operativos para sistemas embebidos. Programación de microcontroladores de 32 bits. Model checking.

XIII - Imprevistos

Comunicarse con la cátedra.
Sistemas Embebidos.
Departamento de Informática.
Of. 25. Bloque II. 1er. Piso.
Facultad de Cs. Fisico, Matemáticas y Naturales.
Universidad Nacional de San Luis.
Ejercito de los Andes 950. CP 5700.

XIV - Otros

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	