RIDAA
Repositorio Institucional
Digital de Acceso Abierto de la
Universidad Nacional de Quilmes

Universidad
Nacional
de Quilmes

Juárez, José

Diseño basado en microcontroladores

Esta obra está bajo una Licencia Creative Commons Argentina.
Atribución - No Comercial - Compartir Igual 2.5
https://creativecommons.org/licenses/by-nc-sa/2.5/ar/

Documento descargado de RIDAA-UNQ Repositorio Institucional Digital de Acceso Abierto de la Universidad Nacional de Quilmes de la Universidad Nacional de Quilmes

Cita recomendada:
Juárez, J. y Perri, V. (2024). Diseño basado en microcontroladores. (Programa). Bernal, Argentina: Universidad Nacional de Quilmes. Disponible en RIDAA-UNQ Repositorio Institucional Digital de Acceso Abierto de la Universidad Nacional de Quilmes http://ridaa.unq.edu.ar/handle/20.500.11807/6739

Puede encontrar éste y otros documentos en: https://ridaa.unq.edu.ar

Universidad
Nacional
de Quilmes
Roque Sáenz Peña 352 // Bernal
Buenos Aires // Argentina
t.: (+41 11) 4365 7100
f.: (+54 11) 4365 7101
info@unq.edu.ar

PROGRAMA de DISEÑO BASADO EN MICROCONTROLADORES

Carrera: Ingeniería en Automatización y Control Industrial.

Asignatura: Diseño Basado en Microcontroladores [1]

Núcleo al que pertenece: Núcleo Superior Básico [2]

Profesoras/es: Ing. José Juárez – Ing. Víctor Perri

Asignaturas previas necesarias para favorecer el aprendizaje: Organización y Arquitectura de Computadores

Objetivos:

Comprender los fundamentos básicos de los microcontroladores y su aplicación en problemas de Automatización y Control.

Conocer la Arquitectura interna y el funcionamiento de los microcontroladores y sus circuitos periféricos.

Aprender técnicas de programación, depuración y simulación para resolver problemas utilizando microcontroladores.

Resolver, utilizando herramientas de desarrollo o simulación, un problema completo de automatización y control, interfaces de comunicación, periféricos analógicos y digitales.

Contenidos mínimos:

Programación del μC y sus sistemas periféricos con lenguajes de alto nivel. Sistemas periféricos y sus aplicaciones en automatización y control: Periféricos de Temporización, Generación y Medición de señales digitales y Periféricos de comunicación. Interfaces de I/O, Periféricos de Conversión A/D y D/A. Análisis y diseño de sistemas controlados por tiempo o por eventos. Introducción a los Sistemas Operativos de Tiempo Real (RTOS) y su aplicación en microcontroladores.

1 En el plan vigente Plan vigente, RCS Nº455-15. Para el Plan RCS Nº 183-03 es equivalente a *Diseño de Controladores Digitales*

2 En el plan vigente *Plan vigente, RCS Nº455-15. Para el Plan RCS Nº 183-03 pertenece al Núcleo Básico del Ciclo Superior*

Carga horaria semanal: *6 horas.*

Programa analítico:

1. Arquitectura de microcontroladores. Conceptos Básicos. Diagrama en bloques de un microcontrolador. Memorias Volátiles y no volátiles. Bus de direcciones, Bus de datos y Control. Procesadores CISC y RISC. Arquitecturas Von Newman y Harvard. Mapas de memoria. Subsistema E/S. DMA.
2. Diagrama en bloques de la CPU. Registros de CPU. Unidad aritmética lógica ALU, Códigos de condición. Procesamiento de Instrucciones en etapas Pipeline. Ciclo de máquina y ciclo de instrucción, ciclo de fetch y ejecución, contador de programa, decodificador de Instrucción. Memoria interna, Periféricos. Estructuras CMOS de entrada-salida. Niveles Lógicos.
3. Interrupciones. Internas, externas, por software, enmascarables y no enmascarables. Reset y Watchdog. Priorización de interrupciones. Procesamiento de interrupciones, cambio de contexto. Tabla de vectores.
4. Protocolo de comunicación serie sincrónico (SPI e I2C) y asincrónico (RS-232). Programación por interrupciones y poolling periódico.
5. Temporizador, contador free running, entradas de captura, salidas de comparación, interrupción de tiempo real, acumulador de pulsos. Generación de señales con Temporizador y PWM, modos de funcionamiento, aplicación sobre relés, solenoides, encoders, motores DC y paso a paso.
6. Conversor AD. Conversor de aproximaciones sucesivas. Acondicionamiento de señal de sensores, Error de cuantización. Conversor DA, Conversor R-2R. Implementación de Controlador PID discreto.
7. Introducción a los Sistemas Operativos de Tiempo Real (RTOS) y su aplicación en microcontroladores.

Bibliografía Obligatoria:

- The AVR Microcontroller and Embedded Systems Using Assembly and C:

Using Arduino Uno and Atmel Studio. 2nd Edición. Mazidi-Naimi, Prentice Hall, 2017.

- Microcontroller Based Applied Digital Control, Dogan Ibrahim, John Wiley, 2006.
- Michael Pont, Patterns for Time Triggered Embeded Systems. Addison Wesley, 2014.

Bibliografía de consulta:

- Galeano, Programación de Sistemas Embebidos en C. Alfa Omega, 2009.
- Embedded Microcomputer Systems Real Time Interfacing, Jonathan Valvano, Thomson, 2007.
- Kernighan Ritchie, El Lenguaje de Programación C, 1999.
- Hojas de datos del microcontrolador, de los circuitos utilizados y notas de aplicación.
- Manuales de las herramientas de desarrollo de software utilizadas

Organización de las clases:

El curso es dictado en clases teórico prácticas y clases de laboratorio a desarrollar en dos clases semanales de tres horas cada una.

Las clases teóricas, a desarrollarse en la primera hora y media, consisten en la presentación, a través de medios audiovisuales y uso del pizarrón, de los distintos temas que componen el programa de la asignatura.

Las clases prácticas consisten en la resolución de problemas presentados en las guías de trabajos prácticos, que refuerzan los conceptos desarrollados en la teoría, fomentando la participación de los alumnos y la evacuación de consultas de los trabajos prácticos correspondientes. Se dictan al finalizar la teoría y duran hasta completar las tres horas de clase. Se utiliza principalmente el pizarrón y el uso del proyector cuando es necesario recurrir algún recurso bibliográfico.

Las clases de laboratorio están enfocadas a resolver problemas específicos utilizando las herramientas de desarrollo de software (editor-compilador-simulador-depurador) y los kits de desarrollo (o simulador) con un conjunto de circuitos auxiliares, por ejemplo leds, switchs, display LCD, sensores, etc. y además utilizando instrumentos de medición como multímetros, osciloscopios o

generadores de señal cuando corresponda.

Se utiliza además el proyector para presentar el problema, las herramientas de desarrollo, ejemplos de programas, manuales, simulaciones, etc. de manera de orientar a los alumnos en la resolución del mismo, aplicando los conceptos desarrollados y sobre todo fomentar la participación de los mismos en clase y el trabajo en equipo. Por esta razón los laboratorios son grupales.

La materia cuenta con 5 laboratorios realizados cada uno en dos clases completas de seis horas. Los laboratorios se listan a continuación:

1) Introducción al manejo del MCU y las herramientas de desarrollo. Operaciones entrada/salida. Control de Leds y pulsadores.
2) Control de display LCD y teclado matricial.
3) Utilización de Timer. Generación de señales de frecuencia variable y PWM. Medición de período con entradas de capturas.
4) Utilización de interrupciones. Utilización de interfaces de comunicación UART/USB con PC u otros dispositivos.
5) Utilización del conversor A/D y D/A e implementación del controlador PID.

Modalidad de evaluación:

La modalidad de evaluación y aprobación será según el Régimen de estudios vigente (Res. CS 201/18): está compuesta de dos exámenes teórico-prácticos con sus correspondientes recuperatorios. Se dispone además de la instancia de evaluación integradora al final el cuatrimestre.

Modalidad de evaluación de exámenes libres:

En la modalidad libre, se evaluarán los contenidos de la asignatura con un examen práctico y un examen escrito. La aprobación de la evaluación práctica será requisito para poder rendir el examen escrito. Se evaluarán todos los contenidos especificados en el programa analítico, incluyendo demostraciones teóricas y problemas de aplicación, en forma escrita. En el examen práctico se evaluarán los contenidos de los laboratorios mediante la resolución de problemas utilizando el software y las herramientas de desarrollo.

ANEXO II

CRONOGRAMA TENTATIVO

Semana	Tema/unidad	Actividad*				Evaluación
		Teórico	Práctico			
			Res Prob	Lab	Proyecto	
1	U1: Arquitectura de computadoras. Memorias Volátiles y no volátiles. Buses. Procesadores CISC y RISC. Arquitecturas Von Newman y Harvard. Mapas de memoria. Subsistema E/S. U2: Registros de CPU. Unidad aritmética lógica ALU, Procesamiento de Instrucciones en etapas Pipeline. Mapa de Memoria interna.	X				
2	U2: Estructuras CMOS E/S. Niveles Lógicos.	X	X			
3	LAB1		X	X		
4	LAB2		X	X		
5	U3: Interrupciones. Priorización de interrupciones. Tabla de vectores.U3: Procesamiento de interrupciones, cambio de contexto. Reset y Watchdog.	X	X			
6	Evaluación Teórico-práctico 1					X
7	U4: Protocolo de comunicación serie sincrónico (SPI, I2C) y asincrónico (RS-232). Programación por interrupciones y pooling periódico.	X	X			
8	LAB3		X	X		
9	U5: Temporizador, interrupción de tiempo real, acumulador de pulsos.	X	X			
10	U5: Generación de señales con Temporizador y PWM.	X	X			
11	LAB4		X	X		
12	Recuperación Evaluación Teórico-práctico 1					X
13	U6: Conversor AD. Conversor de aproximaciones sucesivas. Acondicionamiento de señal de sensores, Error de cuantización.	X	X			
14	U6: Conversor DA, Conversor R-2R. Implementación de Controlador PID discreto. U7: Introducción a los RTOS para Microcontroladores.	X	X			
15	LAB5		X	X		
16	Evaluación Teórico-práctico 2					X
17	Recuperación Evaluación Teórico-práctico 2					X
18	Evaluación Integradora					X

Firmado digitalmente por: SUAREZ Mariana Alejandra
Directora del Departamento de Ciencia y Tecnología
UNIVERSIDAD NACIONAL DE QUILMES