

<p style="text-align: center;">UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p>  <p style="text-align: center;">UNIDAD ACADÉMICA:</p> <p style="text-align: center;">FACULTAD DE INGENIERÍA</p> <p style="text-align: center;">PROGRAMA ANALÍTICO DE LA UNIDAD DE APRENDIZAJE:</p> <p style="text-align: center;"><u>RTOS (SISTEMAS OPERATIVOS EN TIEMPO REAL)</u></p>	DES:	INGENIERÍA
	Programa académico	Ingeniería en Computación
	Tipo de materia (Obli/Opta):	Optativa
	Clave de la materia:	OPCO808
	Semestre:	Octavo
	Área en plan de estudios:	Específica
	Total de horas por semana:	6
	<i>Teoría: Presencial o Virtual</i>	0
	<i>Laboratorio o Taller:</i>	4
	<i>Prácticas:</i>	0
	<i>Trabajo extra-clase:</i>	2
	<i>Créditos Totales:</i>	6
	Total de horas semestre (x sem):	96
	Fecha de actualización:	Octubre 2024
<i>Prerrequisito (s):</i>	N/A	
DESCRIPCIÓN:		
<p>En este curso el alumno aprenderá a utilizar el kernel de un sistema operativo en tiempo real con la finalidad de desarrollar aplicaciones de tiempo real profundamente integradas (en sistemas embebidos) que se ejecutan en microcontroladores o microprocesadores pequeños.</p>		
COMPETENCIAS PARA DESARROLLAR:		
B4. Transformación Digital		
<p>Transforma la cultura digital en la sociedad, en las organizaciones e instituciones educativas para aprovechar al máximo el potencial de las tecnologías y herramientas digitales; propiciar su uso responsable y ético que estimule la creatividad, innovación, la comunicación efectiva y el trabajo colaborativo e interdisciplinar en la solución de problemas de la sociedad digital; promoviendo la privacidad y la seguridad, así como el respeto a los derechos de autor y la propiedad intelectual.</p>		
COMPETENCIAS ESPECÍFICAS:		
E5. ANÁLISIS, DISEÑO Y DESARROLLO DE SISTEMAS INFORMÁTICOS Y ELECTRÓNICOS.		
<p>Resuelve problemas complejos y diseña soluciones efectivas en el campo de los sistemas informáticos y electrónicos. Desde la evaluación de requisitos hasta la creación de arquitecturas robustas y la implementación eficiente. Fomenta la aplicación de conceptos de última generación, la resolución de problemas y el uso de tecnologías emergentes. Tiene la capacidad para trabajar en equipo, comunicar ideas de manera efectiva y contribuir al avance continuo de la informática y la electrónica.</p>		
E4. APLICACIONES EMBEBIDAS, AUTOMATIZACIÓN y CONTROL.		
<p>Usar de forma apropiada teorías, conceptos, procedimientos y herramientas de la ingeniería a través del diseño, desarrollo, implementación y aplicación de Tecnologías Emergentes (TE) tales como las utilizadas en Automatización y Control, Sistemas Embebidos e Internet de las Cosas.</p>		

DOMINIOS (Se toman de las competencias)	OBJETOS DE ESTUDIO (Contenidos necesarios para desarrollar cada uno de los dominios)	RESULTADOS DE APRENDIZAJE (Se plantean de los dominios y contenidos)	METODOLOGÍA (Estrategias, secuencias, recursos didácticos)	EVIDENCIAS (Productos tangibles que permiten valorar los resultados de aprendizaje)
<p>B4.9 Se mantiene actualizado en tendencias y herramientas digitales.</p> <p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>1. INTRODUCCIÓN A LOS SISTEMAS OPERATIVOS DE TIEMPO REAL (RTOS).</p> <p>1.1 Características de un kernel RTOS. 1.2 Distribución libre de un RTOS (FreeRTOS). 1.3 Archivos fuente y configuración de un proyecto RTOS. 1.4 Tipos de datos y estilos de codificación.</p>	<p>Describe a un nivel superior la estructura del directorio de un RTOS, los archivos fuente, la configuración de un proyecto para iniciar con RTOS y los tipos de datos.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>

<p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>2. GESTIÓN DE MEMORIA.</p> <p>2.1 Prerrequisitos y alcance. 2.2 Asignación de memoria dinámica. 2.3 Esquemas de asignación de memoria. 2.4 Funciones y macros. 2.5 Asignación de memoria estática.</p>	<p>Describe el procedimiento para la gestión y asignación de la memoria RAM, así como el esquema a utilizar dentro de un proyecto con RTOS.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>
<p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>3. GESTIÓN DE TAREAS.</p> <p>3.1 Funciones y creación de Tareas. 3.2 Estados y nivel superior de prioridades de tareas. 3.3 Medición del tiempo e interrupción por Tick. 3.4 Tarea inactiva y el enganche de una tarea inactiva. 3.5 Cambio de prioridades de tareas. 3.6 Eliminación de tareas. 3.7 Almacenamiento local y reentrada de hilos. 3.8 Algoritmos de planificación.</p>	<p>Describe cómo implementar en una aplicación RTOS tareas, crear una o más instancias de una tarea, como usar los parámetros de una tarea, el tiempo de procesamiento de cada tarea en una aplicación, elección de la tarea a ser ejecutada en un tiempo dado, prioridad entre tareas y eliminación de tareas.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>

<p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>4. GESTIÓN DE COLAS.</p> <p>4.1 Características y uso colas. 4.2 Recibiendo datos de múltiples fuentes. 4.3 Trabajando con una gran cantidad de datos o de tamaño variable. 4.4 Recibiendo de múltiples colas.</p>	<p>Describe los pasos para crear y gestionar una cola, así como los datos que contiene.</p> <p>Describe el procedimiento para enviar y recibir datos a una cola.</p> <p>Describe el concepto de bloqueo de colas.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>
<p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>5. GESTIÓN DEL TEMPORIZADOR POR SOFTWARE.</p> <p>5.1 Funciones de devolución de llamada del temporizador por software. 5.2 Atributos y estados de un temporizador por software. 5.3 Creando e inicializando un temporizador por software. 5.4 El identificador (ID) de un Temporizador. 5.5 Cambio del periodo de un temporizador. 5.6 Reseteando un temporizador por software.</p>	<p>Describe el propósito de los temporizadores de software para programar la ejecución de una función en un momento determinado en el futuro, o periódicamente con una frecuencia fija.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>

<p>plataformas apropiadas en cada proyecto.</p>		<p>rutinas de servicio de interrupción.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>
<p>Diseñar y desarrollar hardware y/o software embebido, dependiendo del campo de aplicación de la TE.</p> <p>Diseño de arquitectura de software y hardware, conocimientos en patrones de diseño, y la capacidad de evaluar opciones tecnológicas para seleccionar plataformas apropiadas en cada proyecto.</p>	<p>7. MANEJO DE RECURSOS Y GRUPOS DE EVENTOS.</p> <p>7.1 Secciones críticas y suspensión del planificador.</p> <p>7.2 Mutexes y semáforos binarios.</p> <p>7.3 Tareas del guardián (gatekeeper).</p> <p>7.4 Características de un grupo de eventos.</p> <p>7.5 Gestión de eventos mediante grupos de eventos.</p> <p>7.6 Sincronización de tareas mediante un grupo de eventos.</p>	<p>Describe por qué es necesaria la gestión y el control de recursos y la suspensión del planificador.</p> <p>Describe qué es y cómo utilizar un mutex.</p> <p>Describe cómo crear y utilizar una tarea de guardián.</p> <p>Describe las aplicaciones prácticas para los grupos de eventos, así como las ventajas y desventajas de éstos en relación con otras funciones de RTOS.</p> <p>Describe cómo configurar un grupo de eventos y como utilizar un grupo de eventos para sincronizar un conjunto de tareas.</p>	<p>Lectura Crítica.</p> <p>Búsqueda de Información.</p> <p>Resolución de Problemas Analíticos.</p> <p>Entorno de desarrollo Integrado para programación de microcontroladores</p>	<p>Tareas y ejercicios extra clase.</p> <p>Programas/scripts de computadora y/o reporte de práctica.</p>

FUENTES DE INFORMACIÓN (Bibliografía, direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES (Criterios, ponderación e instrumentos)
<ol style="list-style-type: none"> Barry, R. (2023). Mastering the FreeRTOS™ Real Time Kernel. AMAZON. Walls, C. (2021). Embedded RTOS Design: Insights and Implementation. Newnes. Kopetz, H., & Steiner, W. (2022). <i>Real-time systems: Design principles for distributed embedded applications</i> (3rd ed.). Springer Nature. ISBN 9783031119910. 	<p>Estrategias de evaluación:</p> <ul style="list-style-type: none"> Repositorio con tareas de clase, reportes de prácticas de laboratorio y/o programas de computadora. <p>Instrumentos:</p> <ul style="list-style-type: none"> Listas de cotejo. Rúbricas de evaluación para los reportes de las prácticas de laboratorio y/o programas de computadora. <p>Ponderación:</p> <ul style="list-style-type: none"> Primer parcial Tareas 40% Práctica de laboratorio/Desarrollo de programas 60% Segundo parcial Tareas 40% Práctica de laboratorio/Desarrollo de programas 60% Tercer parcial Tareas 40% Práctica de laboratorio/Desarrollo de programas 60% <p>La calificación mínima aprobatoria será de 7.0 Las actividades asignadas, así como tareas deben presentar rubricas y/o listas de cotejo.</p>

CRONOGRAMA

Objetos de estudio	Semanas																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
1. INTRODUCCIÓN A LOS SISTEMAS OPERATIVOS DE TIEMPO REAL (RTOS).																	
2. GESTIÓN DE MEMORIA.																	
3. GESTIÓN DE TAREAS.																	
4. GESTIÓN DE COLAS.																	
5. GESTIÓN DEL TEMPORIZADOR POR SOFTWARE.																	
6. MANEJO DE INTERRUPCIONES.																	
7. MANEJO DE RECURSOS Y GRUPOS DE EVENTOS.																	