

23-24

MÁSTER UNIVERSITARIO EN
INGENIERÍA INDUSTRIAL

GUÍA DE ESTUDIO PÚBLICA



MICROCONTROLADORES PIC

CÓDIGO 28806555

UNED

ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN	
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR ESTA ASIGNATURA	
EQUIPO DOCENTE	
HORARIO DE ATENCIÓN AL ESTUDIANTE	
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE	
RESULTADOS DE APRENDIZAJE	
CONTENIDOS	
METODOLOGÍA	
SISTEMA DE EVALUACIÓN	
BIBLIOGRAFÍA BÁSICA	
BIBLIOGRAFÍA COMPLEMENTARIA	
RECURSOS DE APOYO Y WEBGRAFÍA	
PRÁCTICAS DE LABORATORIO	

Nombre de la asignatura	MICROCONTROLADORES PIC
Código	28806555
Curso académico	2023/2024
Título en que se imparte	MÁSTER UNIVERSITARIO EN INGENIERÍA INDUSTRIAL
Tipo	CONTENIDOS
Nº ETCS	5
Horas	125.0
Periodo	SEMESTRE 1
Idiomas en que se imparte	CASTELLANO

PRESENTACIÓN Y CONTEXTUALIZACIÓN

IMPORTANTE: La asignatura se basará fundamentalmente en las arquitecturas ARM de 32 bits.

Las arquitecturas ARM (*Advanced RISC Machine* o *Acorn RISC Machines*) están entre las más versátiles, estudiadas y utilizadas ("*200 billion Arm-based chips shipped by our partners over the past three decades*", fuente ARM 18-10-2021) en la actualidad para sistemas embebidos (en comparación con la obsolescencia, y dificultad de adquisición, de muchos productos basados en las arquitecturas PIC). Así mismo, en la actualidad existen multitud de tarjetas de evaluación, y *hardware* asociado, a precios muy asequibles para arquitecturas ARM. Cabe destacar también el numeroso material y bibliografía disponibles sobre ARM (principalmente en inglés). Se recomienda leer la información sobre ARM dada, por ejemplo, en *wikipedia* para tener una idea más detallada (preferiblemente en inglés).

La asignatura pertenece a la materia "Sistemas electrónicos y automáticos" y tiene como objetivo dar una visión completa de las principales arquitecturas de la familia ARM Cortex de 32 bits, principalmente las familias M3 y M4, así como de su programación en lenguaje C y en ensamblador. Es importante diferenciar entre la arquitectura (bajo nivel) y el microcontrolador y el entorno de desarrollo disponible en el mercado. De hecho, mientras que los dispositivos Arduino más básicos (Uno R3 y Mega entre otros) emplean arquitecturas AVR de 8 bits (con reloj a 16 MHz), los dispositivos más potentes de Arduino emplean microcontroladores ARM de 32 bits (Arduino Due monta ARM M3 de 32 bits, coma fija y 84 MHz de reloj), mientras que algunos de los modelos de Raspberry montan ARM de 64 bits. Cabe comentar que en la última versión R4 de la placa Arduino Uno (junio 2023) se emplea un núcleo ARM - M4 con unidad de coma flotante FPU, rango de tensiones extendido (tecnología CMOS con tensiones extendidas al rango TTL, pero corriente reducida en los puertos I/O), reloj de 48 MHz y memoria ampliada; lo cual supone un salto considerable con respecto a la versión anterior R3 basada en AVR. Para sistemas de tiempo real, también hay que resaltar la notable diferencia, en resolución y rapidez, entre los ADC empleados en los microcontroladores basados en ARM de 32 bits y las familias Arduino basadas en AVR de 8 bits. En muchos de los microcontroladores ARM de 32 bits es posible incluso instalar un sistema operativo de tiempo real (RTOS). No obstante, en aplicaciones con constantes de tiempo muy pequeñas (incluso del orden de microsegundos) es necesario implementar la aplicación a bajo nivel sin sistema operativo, tal y como se plantea en esta asignatura.

La asignatura se basa en ARM, fundamentalmente la familia M3 (coma fija) y la familia M4

(con unidad de coma flotante FPU). Se centra principalmente la atención en los entornos de desarrollo avanzados para ARM, aunque también se realiza una breve introducción al entorno de desarrollo basado en Arduino, y otro tipo de plataformas, ya que es importante saber seleccionar la arquitectura más adecuada en función de la aplicación (simplicidad vs potencia) y combinar diferentes dispositivos y entornos. Además, muchos de los accesorios *hardware* para microcontroladores disponibles actualmente en el mercado provienen de las plataformas Arduino basadas en AVR. La familia Arduino-AVR presenta la ventaja de su facilidad de desarrollo ofreciendo unas prestaciones moderadas (válidas para sistemas con constantes de tiempo lentas). Por su parte, la arquitectura y entornos ARM presentan mayor complejidad de desarrollo y aprendizaje, pero una mucha mayor potencia, permitiendo su uso en aplicaciones con constantes de tiempo muy reducidas y sistemas de tiempo real estricto (*Hard Real Time*), como el control en tiempo real y el procesado digital de señales. Las prácticas de la asignatura son obligatorias (ver apartado de prácticas y evaluación). Para su realización se empleará un entorno de simulación software (QtARMSim) junto con una tarjeta de desarrollo STM32 de la familia M4 (con unidad de coma flotante FPU). Se introducen algunos de los periféricos más habituales empleados en sistemas embebidos y algunas cuestiones importantes sobre el diseño *hardware* (por ej. divisores de tensión, transistores, optoacopladores, etc). Se presentará también la posibilidad de combinar diversos tipos de microcontroladores y dispositivos para optimizar las prestaciones y minimizar el tiempo de desarrollo. Así mismo, se introducen conceptos y herramientas sobre conectividad de dispositivos (por ej. IoT).

El enfoque de la asignatura es eminentemente aplicado y se pretende que el alumno ponga en práctica los conocimientos teóricos adquiridos. Las familias de la arquitectura ARM (por ej. M3, M4, M7) emplean la filosofía RISC (*Reduced Instruction Set Computer*) y presentan gran compatibilidad entre ellas (juego de instrucciones). Se comienza con un repaso de la arquitectura ARM y su lenguaje ensamblador (familia M3 de coma fija). Se continua introduciendo la familia M4 (con unidad de coma flotante FPU) y su programación en lenguaje C. Por último, se realiza una actividad práctica de implementación para aplicar los conocimientos adquiridos a la programación de un microcontrolador ARM Cortex M4 en un sistema embebido y analizando los resultados con instrumental de laboratorio y *software* científico (por ej. Matlab o Python).

A nivel profesional, el aprendizaje de estos conocimientos resultará de gran utilidad a la hora de trabajar con cualquier sistema electrónico, ya que las habilidades adquiridas aquí servirán tanto para el prototipado como para el desarrollo de sistemas electrónicos que hagan uso de microcontroladores, que son numerosos. De especial interés son las habilidades obtenidas en la programación de bajo nivel de microcontroladores para sistemas de tiempo real estricto y sistemas embebidos, perfil muy demandado en la industria.

Esta guía proporciona las directrices básicas que el estudiante necesita para estudiar la asignatura. Por esta razón, es recomendable leer detenidamente la guía antes de comenzar el estudio, para adquirir una visión general de la asignatura, las actividades y las prácticas que se desarrollarán durante el curso. Esta asignatura es optativa del primer semestre del Máster Universitario en Ingeniería Industrial. Al tratarse de una asignatura del primer semestre del plan de estudios, no se basa en conocimientos previos de ninguna asignatura

del mismo. No es imprescindible, aunque sí recomendable, poseer conocimientos previos de programación y de arquitectura de computadores para cursar la asignatura. También es interesante que el alumno posea algunos conocimientos básicos de electrónica digital, electrónica analógica, métodos numéricos y procesado digital de señales. Es importante tener cierta soltura en la lectura de textos técnicos en inglés.

Los estudiantes conseguirán 5 ECTS con esta asignatura optativa después de superarla con éxito.

REQUISITOS Y/O RECOMENDACIONES PARA CURSAR ESTA ASIGNATURA

Se recomienda tener los siguientes conocimientos previos:

- Pensamiento analítico y de síntesis.
- Capacidad de aplicar los conocimientos a problemas reales.
- Capacidad para trabajar y aprender de manera autónoma.
- Habilidades para la investigación, la planificación y la organización.
- Capacidad de aprender nuevas metodologías y tecnologías.
- Habilidades en tecnologías de la información y programación.
- Habilidad en el uso de las TIC.
- Conocimientos de Electrónica Analógica y Digital.
- Conocimientos de Arquitectura de Computadores.
- Conocimientos de control, métodos numéricos y procesado de señal.
- Habilidades en comunicación oral y escrita en Inglés.

EQUIPO DOCENTE

Nombre y Apellidos
Correo Electrónico
Teléfono
Facultad
Departamento

SANTIAGO MONTESO FERNANDEZ (Coordinador de asignatura)
smonteso@ieec.uned.es

HORARIO DE ATENCIÓN AL ESTUDIANTE

La comunicación entre el equipo docente y los alumnos se hará a través de la plataforma virtual aLF o por e-mail con los profesores. El día de contacto por teléfono serán los martes por la mañana de 10:00 a 14:00 horas en el teléfono 913986481.

Santiago Monteso (coordinador de la asignatura): smonteso@ieec.uned.es

C/ Juan del Rosal, 12
28040 Madrid

Se recomienda al alumno la utilización del curso virtual creado al efecto como soporte de la asignatura (al que puede acceder desde las páginas Web de la UNED), así como la asistencia periódica a las tutorías en su Centro Asociado.

TUTORES

Se recomienda a los Tutores de la asignatura que se pongan en contacto con el Profesor a principio de curso para verificar si existe alguna anomalía respecto de las directrices dadas en esta guía de curso y, si ello fuera necesario, para pedir recomendaciones metodológicas en los aspectos didácticos de la misma.

COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Competencias Básicas:

CB6 - Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación

CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio

CB8 - Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios

CB9 - Que los estudiantes sepan comunicar sus conclusiones -y los conocimientos y razones últimas que las sustentan- a públicos especializados y no especializados de un modo claro y sin ambigüedades

CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

Competencias Generales:

CG1 - Iniciativa y motivación

CG2 - Planificación y organización

CG3 - Manejo adecuado del tiempo

CG4 - Análisis y síntesis

CG5 - Aplicación de los conocimientos a la práctica

CG6 - Resolución de problemas en entornos nuevos o poco conocidos

CG7 - Pensamiento creativo

CG8 - Razonamiento crítico

CG9 - Toma de decisiones

CG10 - Seguimiento, monitorización y evaluación del trabajo propio o de otros

CG11 - Aplicación de medidas de mejora

CG12 - Innovación

CG13 - Comunicación y expresión escrita

CG14 - Comunicación y expresión oral

CG15 - Comunicación y expresión en otras lenguas

CG16 - Comunicación y expresión matemática, científica y tecnológica

CG17 - Competencia en el uso de las TIC

CG18 - Competencia en la búsqueda de la información relevante

CG19 - Competencia en la gestión y organización de la información

CG20 - Competencia en la recolección de datos, el manejo de bases de datos y su presentación

CG21 - Habilidad para coordinarse con el trabajo de otros

CG22 - Habilidad para negociar de forma eficaz

CG23 - Habilidad para la mediación y resolución de conflictos

CG24 - Habilidad para coordinar grupos de trabajo

CG25 - Liderazgo

CG26 - Conocimiento y práctica de las reglas del trabajo académico

CG27 - Compromiso ético y ética profesional

CG28 - Conocimiento, respeto y fomento de los valores fundamentales de las sociedades democráticas

CG29 - Tener conocimientos adecuados de los aspectos científicos y tecnológicos de: métodos matemáticos, analíticos y numéricos en la ingeniería, ingeniería eléctrica, ingeniería energética, ingeniería química, ingeniería mecánica, mecánica de medios continuos, mecánica de fluidos, electrónica industrial, automática, fabricación, materiales, métodos cuantitativos de gestión, informática industrial, urbanismo, infraestructuras, etc.

CG36 - Conocimiento, comprensión y capacidad para aplicar la legislación necesaria en el ejercicio de la profesión de Ingeniero Industrial.

Competencias Específicas:

CE7 - Capacidad para diseñar sistemas electrónicos y de instrumentación industrial.

CE8 - Capacidad para diseñar y proyectar sistemas de producción automatizados y control avanzado de procesos.

CE16 - Capacidad para la gestión de la Investigación, Desarrollo e Innovación tecnológica.

CE20 - Conocimiento y capacidades para el proyectar y diseñar instalaciones eléctricas y de fluidos, iluminación, climatización y ventilación, ahorro y eficiencia energética, acústica, comunicaciones, domótica y edificios inteligentes e instalaciones de Seguridad.

CE22 - Conocimientos y capacidades para realizar verificación y control de instalaciones, procesos y productos.

CE23 - Conocimientos y capacidades para realizar certificaciones, auditorías, verificaciones, ensayos e informes.

RESULTADOS DE APRENDIZAJE

Con el estudio de esta asignatura el alumno adquirirá los siguientes resultados de aprendizaje:

- Aprendizaje de la arquitectura de los microcontrolados ARM de 32 bits: CPU, memoria y registros de control de periféricos, manejo de interrupciones y comunicaciones entre otros.
- Diseño de sistemas digitales a través de la programación en ensamblador y en C.
- Análisis y diseño de sistemas embebidos basados en microcontroladores.

- Fundamentos de las arquitecturas AVR y del entorno Arduino.

CONTENIDOS

Bloque 1. Introducción a la arquitectura de computadores basada en ARM

Se introducen los fundamentos de la arquitectura de computadores, desde un punto de vista conceptual y práctico, tomando como referencia la familia ARM Cortex M3 de 32 bits (Arduino Due) y coma fija. Se utilizan como complemento parte de las hojas de datos de los fabricantes de microcontroladores.

Bloque 2. Programación en ensamblador para ARM

Se presentan los fundamentos de la programación en ensamblador para la arquitectura ARM Cortex M3 (Arduino Due) de coma fija. Se usa el simulador QtARMSim para la realización de prácticas en lenguaje ensamblador.

Bloque 3. Introducción a la arquitectura ARM M4

Se profundiza en la arquitectura ARM Cortex M4 (STM32F4), con unidad *hardware* de coma flotante (FPU), y se ve en detalle su estructura interna, así como la configuración y circuitería de la tarjeta de desarrollo que se usará en las prácticas. Se utilizan como complemento parte de las hojas de datos de los fabricantes de microcontroladores.

Se estudian las características físicas y eléctricas de los elementos hardware del microcontrolador, prestando especial atención a su interfaz y entrada-salida (CMOS vs ECL vs TTL).

Bloque 4. Entornos de desarrollo y programación para ARM

Se presentan los entornos de desarrollo, profesionales y semi-profesionales, para ARM existentes en la actualidad y que se utilizarán para la realización de las prácticas.

Bloque 5. Lenguaje de programación C para microcontroladores

Se introduce el lenguaje de programación C para microcontroladores, el cual se usará a lo largo del texto base 1 y de los siguientes bloques (el texto base 4 trata de la programación en C para arquitecturas AVR de Arduino, no obstante, su adaptación a la familia STM32 es muy intuitiva, caso muy diferente a si se tratase de la portabilidad del lenguaje ensamblador, el cual es específico de cada arquitectura).

Los estudiantes que ya posean conocimientos del lenguaje de programación C pueden hacer una lectura más rápida del contenido de este bloque.

Bloque 6. Programación en C de la arquitectura ARM M4

Se estudia el manejo y la programación de las diferentes unidades *hardware* del microcontrolador ARM Cortex M4 tipo STM32 (entrada-salida, interrupciones, *timers*, ADC y DAC, comunicaciones y acceso a memoria entre otros) y la relación entre la arquitectura del microcontrolador y su programación en C y en lenguaje ensamblador. Se utilizan como complemento parte de las hojas de datos de los fabricantes de microcontroladores.

Bloque 7. Sistemas embebidos basados en ARM M4

Se estudian algunas de las aplicaciones más habituales, y de cierta dificultad, de la programación de microcontroladores en sistemas embebidos (manejo de periféricos, procesamiento digital de señal y control en tiempo real). Se introducen algunos de los periféricos y accesorios más habituales empleados en sistemas embebidos (por ej. sensores y actuadores) y algunas cuestiones importantes sobre el diseño *hardware* (por ej. divisores de tensión, transistores, optoacopladores, etc). Se realiza también un breve repaso de algunos métodos numéricos empleados habitualmente en la programación de sistemas embebidos.

Bloque 8. Otras configuraciones para sistemas embebidos

Este bloque tiene un enfoque conceptual y práctico. Su objetivo es introducir otras alternativas a la familia STM32 de ARM disponibles en el mercado (en particular, AVR y ARM de Arduino) para que el alumno tenga una visión más global del mundo de los microcontroladores y de los sistemas embebidos. Se presenta la opción de combinar en un mismo sistema embebido varios microcontroladores con funciones diferenciadas y jerarquizadas. De hecho, la tendencia actual en aplicaciones de altas prestaciones es incluir en un mismo PCB, o en un mismo dispositivo (*chip*), diferentes (o similares con *multicore*) tecnologías (ARM, DSP y FPGA entre otras) con objeto de optimizar el diseño, aprovechando las ventajas de cada una de ellas, así como de minimizar el tiempo de desarrollo.

Es importante conocer diferentes familias de microcontroladores para su comparativa. Se realiza una breve introducción a las principales arquitecturas de Arduino (AVR vs. ARM) con objeto de introducir los fundamentos para seleccionar la arquitectura más adecuada en función de la aplicación (simplicidad vs potencia) cuando se dispone de varias alternativas. Así mismo, muchos de los accesorios *hardware* para microcontroladores disponibles en el mercado provienen de las plataformas Arduino basadas en arquitecturas AVR. Se introducen conceptos y herramientas sobre conectividad de dispositivos (por ej. IoT). Se utilizan como complemento parte de las hojas de datos de los fabricantes.

METODOLOGÍA

La metodología de estudio utiliza la tecnología actual para la formación a distancia en aulas virtuales, con la participación del Equipo Docente, los Profesores Tutores y todos los alumnos matriculados. En este entorno se trabajarán los contenidos teórico-prácticos cuya herramienta fundamental de comunicación será el curso virtual, utilizando la bibliografía básica y el material complementario.

El trabajo autónomo con las actividades de ejercicios y pruebas de autoevaluación disponibles, bajo la supervisión del tutor, con las herramientas y directrices preparadas por el equipo docente completará el tiempo de estudio y preparación de la asignatura.

Por último esta asignatura tiene programadas unas prácticas cuya realización y superación son requisitos imprescindibles para aprobar la asignatura.

Cronológicamente el estudiante debe estudiar y preparar cada tema siguiendo el orden dado a los contenidos, ya que cada uno se apoya en los anteriores.

SISTEMA DE EVALUACIÓN

TIPO DE PRUEBA PRESENCIAL

Tipo de examen	Examen de desarrollo
Preguntas desarrollo	3
Duración del examen	120 (minutos)
Material permitido en el examen	
Calculadora no programable	
Criterios de evaluación	

Prueba Personal Presencial

Como orientación, la Prueba Personal estará estructurada como sigue:

Una primera parte (30% de la nota del examen) con cuestiones teórico-conceptuales y/o problemas breves.

Una segunda parte (30% de la nota del examen) consistente en la resolución de un problema programando en ensamblador.

Una tercera parte (40% de la nota del examen) consistente en la resolución de un problema programando en C.

Las PEC tendrán un planteamiento similar al del examen final y en ellas se detallarán, y concretarán, los puntos anteriores.

Para las preguntas que requieran código fuente en algún lenguaje de programación, será necesario adjuntar comentarios de cada línea de código para facilitar su comprensión. Se recomienda incluir un diagrama (flujograma) también para mostrar de manera visual el algoritmo desarrollado. Se valorará también una explicación textual inicial describiendo cómo abordará el problema.

Entre los criterios que se tendrán más en cuenta en la corrección de estas preguntas se encuentran los siguientes:

Estructura de programa correcta

Estructuras de control bien diseñadas.

Buen conocimiento de las principales instrucciones y funciones.

Utilización del algoritmo óptimo.

Salida del programa dejando registros de manera estable.

Es obligatorio contestar a todas las partes (es necesario obtener una nota mínima del 20 % en cada parte). Hay que obtener una nota mínima de 5 en el examen para poder hacer media con el resto de prácticas y pruebas de evaluación de la asignatura..

El espacio de respuesta de cada una de las preguntas del examen no está tasado, el estudiante puede utilizar tanto espacio como le sea necesario.

% del examen sobre la nota final 65

Nota del examen para aprobar sin PEC 5

Nota máxima que aporta el examen a la calificación final sin PEC

Nota mínima en el examen para sumar la PEC 5

Comentarios y observaciones

CARACTERÍSTICAS DE LA PRUEBA PRESENCIAL Y/O LOS TRABAJOS

Requiere Presencialidad Si

Descripción

Realización de un examen teórico/práctico en el que se evaluarán todos los contenidos de la asignatura.

Se realizarán más indicaciones en el curso virtual durante el desarrollo de la asignatura y en el enunciado del examen.

Criterios de evaluación

Entre los criterios que se tendrán más en cuenta en la corrección de estas preguntas se encuentran los siguientes:

Estructura de programa correcta

Estructuras de control bien diseñadas.

Buen conocimiento de las principales instrucciones y funciones.

Utilización del algoritmo óptimo.

Salida del programa dejando registros de manera estable.

Ponderación de la prueba presencial y/o los trabajos en la nota final 65%

Fecha aproximada de entrega

Comentarios y observaciones

PRUEBAS DE EVALUACIÓN CONTINUA (PEC)

¿Hay PEC? Si, PEC no presencial

Descripción

Son optativas. Consisten en la realización de problemas y cuestiones similares a los del examen. En el caso de los problemas, estos deben de ser desarrollados en detalle por el alumno para su evaluación y posibles comentarios.

La entrega de las PEC es única (convocatoria ordinaria) y la nota se guarda de febrero a septiembre del mismo curso, pero no para cursos posteriores.

Criterios de evaluación

Ponderación de la PEC en la nota final 10%

Fecha aproximada de entrega Mediados de enero

Comentarios y observaciones

OTRAS ACTIVIDADES EVALUABLES

¿Hay otra/s actividad/es evaluable/s? Si, presencial

Descripción

Prácticas obligatorias

Las prácticas obligatorias constan de las siguientes actividades:

- P1: Práctica no presencial basada en ensamblador y el simulador QtARMSim.
- P2: Práctica no presencial basada en C y el entorno de desarrollo para STM32.
- P3: Práctica presencial con tarjeta de desarrollo STM32F4 (usada en el texto base 1) en el laboratorio de electrónica de la E.T.S.I. Industriales de la UNED (sede central).

Proyecto de prácticas optativo

Es opcional y no presencial. Puede sumar hasta 1 punto en la nota final de la asignatura.

COMENTARIOS SOBRE LAS PRÁCTICAS OBLIGATORIAS

Las prácticas obligatorias persiguen los siguientes objetivos:

Adquisición de destreza y rapidez en la resolución de las prácticas de la asignatura.

Aclaración y consolidación de los conocimientos adquiridos en el estudio aplicados al desarrollo de las prácticas.

Comprobación del nivel de conocimientos.

Implementación de lo estudiado en un sistema embebido real.

Características:

Ejercicios obligatorios.

Para la realización de las prácticas, el alumno deberá descargar los enunciados, el simulador QtARMSim y el entorno de desarrollo para el microcontrolador STM32F4 usado en el texto base 1 (TB1).

La primera parte de las prácticas obligatorias (P1, no presencial) trata sobre la programación en lenguaje ensamblador y la arquitectura de los microcontroladores ARM de 32 bits (M3 y M4). Los ejercicios consistirán en la implementación de un programa de acuerdo a las indicaciones dadas en el enunciado, pudiéndose incluir algunas cuestiones teóricas y problemas. Se usará el simulador QtARMSim.

La segunda parte de las prácticas obligatorias (P2, no presencial) trata sobre la programación en C de la arquitectura STM32. Los ejercicios consistirán en la implementación de un programa de acuerdo a las indicaciones dadas en el enunciado, pudiéndose incluir algunas cuestiones teóricas y problemas. Se usará el entorno de desarrollo para STM32F4 y *software* de cálculo científico (por ej. Matlab o Python).

La tercera parte de las prácticas obligatorias (P3, presencial) consiste en la implementación física, en un sistema embebido real, de lo estudiado en las prácticas previas en una tarjeta de desarrollo STM32F4 (usada en el texto base 1) en el laboratorio de electrónica de la E.T.S.I. Industriales de la UNED (sede central). Así mismo, se empleará instrumental de laboratorio y se incluirán algunos montajes adicionales, junto con el análisis de los resultados y el empleo de *software* de cálculo científico (por ej. Matlab o Python). Se hará uso de hojas de datos de fabricantes.

La sesión de prácticas presencial (P3) es obligatoria y se debe aprobar (nota mínima de 5) para superar la asignatura. Para poder asistir a las prácticas presenciales es requisito imprescindible haber entregado y superado (nota mínima de 5) los trabajos previos de prácticas obligatorias (P1 y P2, no presenciales), así como haber aprobado el examen final (presencial) de la asignatura (nota mínima de 5).

El conjunto de prácticas obligatorias (P1 + P2 + P3) tiene un peso del 35 % en la nota final de la asignatura. Cada parte de las prácticas tiene el siguiente peso en la nota de prácticas:

P1 (ensamblador - no presencial): 25 %

P2 (lenguaje C - no presencial): 35 %

P3 (con tarjeta STM32F4 - presencial): 40 %

(es necesario sacar una nota mínima de 5 en cada práctica)

Es importante resaltar que para realización de la práctica P3 (presencial) se supone que ya se han adquirido los conocimientos previos (mediante el examen y la P1 y P2) y su objetivo es implementar, con posibles modificaciones y ampliaciones, los programas que ya se han hecho en las prácticas no presenciales P1 y P2. Es muy recomendable que el alumno adquiriera al principio del curso la tarjeta de desarrollo STM32F4 utilizada en el texto base 1 (TB1) y que se familiarice con ella, y el entorno de desarrollo, para sacar mucho mayor partido a las prácticas, especialmente a la práctica P3 (presencial).

COMENTARIOS SOBRE EL PROYECTO OPTATIVO

Hay una última parte opcional de prácticas (no presencial) que consiste en el estudio e implementación de conceptos avanzados (por ej. el diseño e implementación de un sistema con varios microcontroladores y/o periféricos avanzados y/o ampliar lo visto en las prácticas obligatorias y/o comunicaciones). El alumno puede, y es recomendable, escoger un tema para el proyecto optativo que le resulte de interés (se valorará la originalidad y complejidad del proyecto). Estos ejercicios tienen como objetivos:

Complementar los conocimientos adquiridos en la asignatura.

Aclaración y consolidación de los conocimientos adquiridos en el estudio aplicados al desarrollo de las prácticas.

Comprobación del nivel de conocimientos.

Profundizar en los contenidos de la asignatura.

Implementación de lo estudiado en un sistema embebido real.

Fomentar la iniciativa del alumno en la resolución de tareas complejas.

Características:

Proyecto optativo, de realización voluntaria.

Es evaluable y puede sumar hasta 1 punto extra para la nota final de la asignatura, siempre y cuando la nota en la prueba presencial, y en las prácticas obligatorias, sea igual o superior a 5 (en cualquier caso la nota máxima de la asignatura será un 10).

Criterios de evaluación

Criterios Evaluación Prácticas y Proyecto

El desarrollo de las prácticas y proyecto se deberá acompañar de una memoria descriptiva de las mismas, incluyendo un diagrama (flujograma) para mostrar de manera visual los algoritmos desarrollados. Así mismo, el código fuente desarrollado deberá incluir comentarios de cada línea de código para facilitar su comprensión. En caso de haber sido necesario incluir algún tipo de hardware, se debe justificar su uso y los criterios de diseño y cálculos (por ej. divisores de tensión, transistores, optoacopladores, etc).

Entre los criterios que se tendrán más en cuenta en la corrección se encuentran los siguientes:

El programa se ejecuta correctamente y sin errores.

El programa realiza la funcionalidad deseada.

Estructura de programa correcta

Librerías necesarias incluidas.

Pines configurados y usados correctamente.

Estructuras de control bien diseñadas.

Buen conocimiento de las principales instrucciones y funciones.

Utilización del algoritmo óptimo.

Salida del programa dejando memoria y registros de manera estable.

El *hardware* y circuitos utilizados funcionan correctamente.

Proyecto optativo

El alumno interesado deberá enviar al equipo docente, en las fechas indicadas en el foro, un breve informe de una hoja (2 páginas máximo) indicando título, descripción y un estudio preliminar.

Ponderación en la nota final

35 % de la nota final de la asignatura

Fecha aproximada de entrega

Las prácticas finales se entregarán una vez finalizados los exámenes de febrero (o septiembre).

Comentarios y observaciones

¿CÓMO SE OBTIENE LA NOTA FINAL?

La PEC cuenta un 10% sobre la nota de teoría y sólo se tiene en cuenta si sube nota, siempre y cuando la nota del examen de teoría sea mayor de 5. Las prácticas (P1 + P2 + P3) son un 35% de la nota final y la teoría un 65%. Es necesario obtener un mínimo de 5 en las prácticas obligatorias y un mínimo de 5 en el examen de teoría.

La nota final de la asignatura se obtiene con la siguiente fórmula:

$$\text{Nota final} = \min(10 ; 0,65 \cdot \text{NT} + 0,35 \cdot \text{NP} + \text{PR})$$

siendo:

NT la nota de la parte teórica:

$$\text{NT} = \max(\text{Nota_examen} ; 0,9 \cdot \text{Nota_examen} + 0,1 \cdot \text{PEC})$$

NP la nota de las prácticas (obligatorias)

PR la nota del proyecto optativo (entre 0 y 1)

Para aplicar la fórmula de cálculo de la nota final es necesario haber obtenido, por separado, una nota mínima de 5 en el examen y en las prácticas.

El proyecto de prácticas optativo puede sumar hasta 1 punto extra para la nota final, siempre y cuando la nota en la prueba presencial, y en las prácticas obligatorias, sea igual o superior a 5 (en cualquier caso la nota máxima de la asignatura será un 10).

Las partes que se aprueben en febrero se guardan para septiembre, pero no para cursos posteriores.

BIBLIOGRAFÍA BÁSICA

ISBN(13):9781796854879

Título:EMBEDDED CONTROLLERS USING C AND ARDUINO (2021)

Autor/es:James M. Fiore ;

Editorial:James M. Fiore (dissidents)

ISBN(13):9783030884383

Título:EMBEDDED SYSTEM DESIGN WITH ARM CORTEX-M MICROCONTROLLERS (2022)

Autor/es:Cem Ünsalan ; Mehmet Erkin Yücel ; Hüseyin Deniz Gürhan ;

Editorial:Springer

ISBN(13):9788416546763

Título:INTRODUCCIÓN A LA ARQUITECTURA DE COMPUTADORES CON QTARMSIM Y ARDUINO (2018)

Autor/es:Sergio Barrachina ; Maribel Castillo ; Germán Fabregat ; Raúl Montoliu ; Germán León ; José Vicente Martí ; Rafael Mayo ; Juan Carlos Fernández ;

Editorial:UNIVERSITAT JAUME I

ISBN(13):9788481588552

Título:ARQUITECTURA DE ORDENADORES. EJERCICIOS PRÁCTICOS DE ARM / THUMB (2020)

Autor/es:Manuel José Fernández ; Martín Llamas ; Fernando Ariel ; Juan Manuel Santos ; Luis Anido ;

Editorial:UNIVERSIDAD DE VIGO

TB1: EMBEDDED SYSTEM DESIGN WITH ARM CORTEX-M MICROCONTROLLERS

TB2: INTRODUCCIÓN A LA ARQUITECTURA DE COMPUTADORES CON QTARMSIM Y ARDUINO

TB3: ARQUITECTURA DE ORDENADORES. EJERCICIOS PRÁCTICOS DE ARM / THUMB

TB4: EMBEDDED CONTROLLERS USING C AND ARDUINO

El texto base 1 (TB1) es por el que se seguirá mayoritariamente la asignatura (arquitectura STM32 M4 y basado en lenguaje C), junto con el texto base 2 (TB2) (arquitectura STM32 M3 y basado en ensamblador) que se usará parcialmente (ambos textos incluyen numerosos ejercicios). El texto base 3 (TB3) es únicamente para complementar la colección de problemas de programación en ensamblador, principalmente para las prácticas. El texto base 4 (TB4) se usa principalmente como introducción al lenguaje C para microcontroladores y para hacer una introducción a la arquitectura AVR.

El texto base 1 incluye la programación de microcontroladores en lenguaje C, C++ y MicroPython. Para la asignatura sólo se usará el lenguaje de programación C que es el más extendido y usado en aplicaciones industriales de microcontroladores con requisitos estrictos de tiempo (*Hard Real Time*).

Los textos base 2, 3 y 4 están disponibles on-line bajo la licencia Creative Commons (CC). Se darán instrucciones en el foro de la asignatura.

Se recomienda también la consulta del libro "EXPLORING ARDUINO: TOOLS AND TECHNIQUES FOR ENGINEERING WIZARDRY (2ª ed.)" incluido en la bibliografía complementaria (basado en Arduino Uno - AVR de 8 bits - en la fecha de publicación de esta guía).

Para aquellos alumnos que necesiten repasar conceptos básicos de electrónica digital, y de fundamentos de arquitectura y estructura de computadores, es recomendable la lectura del libro "ESTRUCTURA Y TECNOLOGÍA DE COMPUTADORES I" incluido en la bibliografía complementaria.

Se usarán ampliamente a lo largo de la asignatura las hojas de datos de los principales fabricantes de microcontroladores y periféricos (mayoritariamente en inglés). Esta información está disponible on-line y se darán más indicaciones en el foro de la asignatura.

BIBLIOGRAFÍA COMPLEMENTARIA

ISBN(13):9780124080829

Título:THE DEFINITIVE GUIDE TO ARM® CORTEX®-M3 AND CORTEX®-M4 PROCESSORS
(2013)

Autor/es:Joseph Yiu ;

Editorial:ELSEVIER

ISBN(13):9780201795233

Título:EMBEDDED C (2002)

Autor/es:Michael J. Pont ;

Editorial:ADISON WESLEY

ISBN(13):9780982692677

Título:EMBEDDED SYSTEMS WITH ARM CORTEX-M MICROCONTROLLERS IN ASSEMBLY LANGUAGE AND C (2023)

Autor/es:Yifeng Zhu ;

Editorial:E-Man Press

ISBN(13):9780997925944

Título:STM32 ARM PROGRAMMING FOR EMBEDDED SYSTEMS (2018)

Autor/es:Muhhammad Ali Mazidi ; Eshragh Ghaemi ; Shujen Chen ;

Editorial:MicroDigitalEd

ISBN(13):9781482229851

Título:ARM ASSEMBLY LANGUAGE. FUNDAMENTALS AND TECHNIQUES. (2014)

Autor/es:Christopher Hinds ; William Hohl ;

Editorial:CRC Press

ISBN(13):9788426733399

Título:ARM CORTEX M4 Y ESP32. PROGRAMACIÓN Y EJEMPLOS. (2021)

Autor/es:Daniel Schmidt ;

Editorial:MARCOMBO

ISBN(13):9788441544987

Título:ARDUINO PRÁCTICO (2022)

Autor/es:Daniel Lozano Equisoain ;

Editorial:ANAYA MULTIMEDIA

ISBN(13):9788499646138

Título:ARDUINO. GUÍA PRÁCTICA DE FUNDAMENTOS Y SIMULACIÓN (1)

Autor/es:López Aldea, Eugenio ;

Editorial:RA-MA EDITORIAL

ISBN(13):9788499647555

Título:MICROCONTROLADOR STM32. PROGRAMACIÓN Y DESARROLLO. (2018)

Autor/es:Jesús María Pestano Herrera ;

Editorial:RA-MA

ISBN(13):9788429126204

Título:ESTRUCTURA Y DISEÑO DE COMPUTADORES: LA INTERFAZ HARDWARE/SOFTWARE (2011)

Autor/es:John L. Hennessy, David A. Patterson ;

Editorial:REVERTÉ

ISBN(13):9788483223475

Título:TRATAMIENTO DIGITAL DE SEÑALES. PRINCIPIOS, ALGORITMOS Y APLICACIONES (4ª ed.)

Autor/es:Proakis, John ; Manolakis, Dimitri G. ;

Editorial:PRENTICE-HALL

ISBN(13):9781119405375

Título:EXPLORING ARDUINO: TOOLS AND TECHNIQUES FOR ENGINEERING WIZARDRY (2020)

Autor/es:Jeremy Blum ;

Editorial:WILEY

ISBN(13):9788436246421

Título:ESTRUCTURA Y TECNOLOGÍA DE COMPUTADORES I (GESTIÓN Y SISTEMAS) (1ª)

Autor/es:Yeves Gutiérrez, Fernando ; Castro Gil, Manuel Alonso ; Sebastián Fernández, Rafael ;

Pérez Molina, Clara ; Peire Arroba, Juan ; Míguez Camiña, Juan Vicente ; Mora Buendía, Carlos De ;

Mur Pérez, Francisco ; López-Rey García-Rojas, África ;

Editorial:U.N.E.D.

Es importante consultar y manejar con soltura las hojas de datos de los fabricantes y los recursos disponibles.

RECURSOS DE APOYO Y WEBGRAFÍA

Como recursos adicionales para el estudio de la asignatura, se recomiendan los siguientes materiales:

- La guía didáctica de la asignatura.
- Pruebas de evaluación a distancia (PECs).
- Ejercicios teórico-prácticos que el alumno puede usar como ejercicios de autoevaluación (incluidos en los textos base).
- Software* para la simulación de microcontroladores (QtARMSim) y entornos de desarrollo.
- Tarjetas de desarrollo basadas en microcontroladores STM32 M4 (coma flotante).

Se recomienda al alumno con acceso a Internet que visite las páginas sugeridas en la bibliografía de cada capítulo de los textos base así como hojas de datos de los principales fabricantes.

PRÁCTICAS DE LABORATORIO

¿Hay prácticas en esta asignatura de cualquier tipo (en el Centro Asociado de la Uned, en la Sede Central, Remotas, Online,..)?

Sí

CARACTERÍSTICAS GENERALES

Presencial: Sí

Obligatoria: Sí

Es necesario aprobar el examen para realizarlas: Sí

Fechas aproximadas de realización: Al finalizar los exámenes de febrero y de septiembre (los trabajos previos de prácticas se realizan con anterioridad), según calendario de prácticas.

Se guarda la nota en cursos posteriores si no se aprueba el examen: No, pero sí de febrero a septiembre del mismo curso.

Cómo se determina la nota de las prácticas: ver apartado "Sistema de evaluación"

REALIZACIÓN

Lugar de realización (Centro Asociado/ Sede central/ Remotas/ Online):

Sede central, en el laboratorio de electrónica de la E.T.S.I. Industriales de la UNED

N.º de sesiones: 1 día (mañana + tarde)

Actividades a realizar: Ejercicios teórico-prácticos y de programación de microcontroladores. Montaje y análisis de sistemas embebidos. Análisis de resultados con instrumental de laboratorio y *software* científico (por ej. Matlab y/o Python).

OTRAS INDICACIONES:

La sesión de prácticas presencial es obligatoria y se debe aprobar (nota mínima de 5) para superar la asignatura. Para poder asistir a las prácticas presenciales es requisito imprescindible haber entregado y superado (nota mínima de 5) los trabajos previos de prácticas obligatorias (no presenciales), así como el examen final (presencial) de la asignatura (nota mínima de 5).

Sólo en casos muy justificados, cabe la posibilidad de que si el alumno, o el centro asociado correspondiente, dispone del material necesario para realizar las prácticas (tarjetas de desarrollo con microcontrolador, osciloscopio, generador de señales, fuente de alimentación, periféricos, etc), y éstas son debidamente tutorizadas, las prácticas se puedan realizar fuera de la E.T.S.I. Industriales de la UNED, previa autorización del coordinador de la asignatura. Preferiblemente debe usarse la tarjeta de desarrollo propuesta en el texto base TB1. El alumno que se encuentre en esta situación debe ponerse en contacto (email) con el coordinador de la asignatura a la mayor brevedad (antes de finales de Noviembre).

IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la

comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.