

Electrónica Digital II

Santiago Rúa Pérez, PhD.

25 de julio de 2022

Identificación

Datos del curso

- Asignatura: Electrónica Digital II
- Dias: Martes-Jueves 12-2 pm. Salon 3-218
- Créditos: 3
- Docente: Santiago Rúa Pérez
- Correo: srua@udemedellin.edu.co

Competencias del curso

- Adquirir habilidades en el uso del lenguaje de programación C para sistemas embebidos.
- Manipular, clasificar y desarrollar aplicaciones con los periféricos básicos de un microcontrolador de mediana complejidad.
- Distinguir y aplicar diferentes técnicas de comunicación serial usando un microcontrolador.
- Desarrollar aplicaciones microcontroladas teniendo en cuenta ciertas restricciones de diseño.
- Adoptar buenas prácticas de desarrollo, análisis y depuración de aplicaciones embebidas.

Compromiso académico

Evaluación del curso - teórico

Trabajo I: 25 %, 23 de Agosto

Lab 1: 10 %, 30 de Agosto

Lab 2: 10 %, 8 de Septiembre

Lab 3: 15 %, 27 de Septiembre

Lab 4: 15 %, 20 de Octubre

Trabajo Final: 25 %, 17 de Noviembre

Con 6 clases o 12 horas directas se cancela el curso.

Registro de notas

Registro 75 %: 1 de noviembre.

Registro 100 %: 18 de noviembre.

Compromiso académico

Cronograma

UNIVERSIDAD DE MEDELLÍN FACULTAD DE INGENIERÍA ELECTRONICA DIGITAL II 2022-20

Tuesday 26-Jul	Thursday 28-Jul	Tuesday 2-Aug	Thursday 4-Aug	Tuesday 9-Aug	Thursday 11-Aug	Tuesday 16-Aug	Thursday 18-Aug
Introducción al curso	Logica combinacional y secuencial	Lenguaje de programación en C	Lenguaje de programación en C - Ciclos y operadores logicos	Lenguaje de programación en C - Programación modular y librerías	Lenguaje de programación en C - Punteros	Lenguaje de programación en C - Estructuras y Uniones	Asesoría - Programación en C
	Semana 1						Trabajo 1 (25%)
23-Aug	25-Aug	30-Aug	1-Sep	6-Sep	8-Sep	13-Sep	15-Sep
Introducción Arquitectura Microcontroladores (ARM)	GPIO	Laboratorio 1 (GPIO)	Concurrencia de software	Excepciones e interrupciones	Laboratorio 2 (Interrupciones)	Interfaces analógicas	Temporizadores
	Semana 5						
		Lab 1 (10%)			Lab 2 (10%)		
20-Sep	22-Sep	27-Sep	29-Sep	4-Oct	6-Oct	11-Oct	13-Oct
Proyecto Aplicado	Proyecto Aplicado	Proyecto Aplicado	UART	I2C	SPI	Laboratorio 3 (Comunicaciones)	Laboratorio 3 (Comunicaciones)
	Semana 9						Lab 4 (15%)
		Lab 3 (15%)					
18-Oct	20-Oct	25-Oct	27-Oct	1-Nov	3-Nov	8-Nov	10-Nov
Estandar de codificación	Introducción a placa reducida - Raspberry Pi	Introducción a placa reducida - Raspberry Pi	Características de hardware	Características de hardware	Asesoría Trabajo Final	Asesoría Trabajo Final	Entrega Trabajo Final - Individual
	Semana 13						Trabajo Final (25%)
Evaluación							
Nota	%	Contenido	Fecha	Nota	%	Contenido	Fecha
Trabajo 1	25	Programación en lenguaje C	18/08/2022	Proyecto 1	15	GPIO, Interrupciones, Interfaces Analógicas, Temporizadores	27/09/2022
Lab 1	10	GPIO	30/08/2022	Lab 3	15	Comunicaciones	13/10/2022
Lab 2	10	Concurrencia de Software	08/09/2022	Trabajo Final	25	Trabajo final	10/11/2022

Software y Hardware

Software

- MCUExpresso
- Compilador de C (VSCode)
- Arduino

Hardware

- Kit de Laboratorio por estudiante

Kit de Laboratorio

Item	Referencia	CANTIDAD POR ESTUDIANTE
Freescape Freedom Development Platform K64F	FRDM-K64F	1
Cable microUSB de 50cm	CA-MICROUSB-50CM	1
Buzzer-Voltaje variable	BZ-002	1
Módulo detector de sonido	SEN-SON-2	1
Kit cables conexión fácil Dupont M/M 20cm x10	EASY-CAB-KIT-MM-20	1
Kit cables conexión fácil Dupont H/H 20cm x10	EASY-CAB-KIT-HH-20	1
Módulo SH-HC-08 Bluetooth 4.0 BLE, UART	SH-HC-08	1
Tarjeta de desarrollo para A9, GSM, GPRS, GPS	A9-DEV1-ANT	1
Módulo XBee 2.4Ghz Serie1	XB24-API-001	2
Display LCD 16x2 3.3V	LCD-1602-3V3	1
Shield LCD 3.5" Touch para Arduino	LCD-3.5-R3A	1
Sensor de luz y proximidad (I2C)	CJMCU-3216	1
Termómetro digital de resolución programable TO-92 (1-Wire)	DS18B20+	1
Sensor de temperatura y humedad SHT30 (I2C)	SHT30-D	1
Fotorresistencia LDR 9mm	LDR10K9MM	1
Teclado matricial 4x4 de membrana	TEC-MEM-16	1
Reloj de tiempo real y memoria (I2C)	RTC-I2C-TINY	1
Convertidor de nivel lógico bidireccional 8 canales	TXS0108E	1
sensor ultrasónico HCSR004	HC-SR04	1
Tarjeta Raspberry Pi 4B 2GB	RPI4-2GB	1
Adaptador Raspberry Pi 4 - 5V 3A	ADP-RASP4-3A	1
Kit disipadores para Raspberry Pi 4 con ventilador	BOXRASP4-VENT-CL	1
Memoria micro SD 16GB Clase 10 SanDisk	SD-16GB-C10	1
Arduino Uno	Arduino Uno	1

Introducción a los sistemas embebidos

- Qué es un sistema embebido?
 - Aplicaciones específicas de los sistemas de cómputo
 - Se utilizan en sistema mas complejos.
- Porque usar un computador en un sistema complejo?
 - Mejor desempeño.
 - Mas funciones y características.
 - Bajo costo.
 - Mayor dependencia.
- Economía
 - Microcontroladores se produce en alto volúmenes.
 - Costo no recurrente dominado por el desarrollo de software
- Red
 - A menudo, el sistema integrado utilizará varios procesadores que se comunican a través de una red para reducir los costos de piezas y ensamblaje y mejorar la confiabilidad.

Opciones para construir Sistemas Embebidos

Dedicated Hardware	Implementation	Design Cost	Unit Cost	Upgrades & Bug Fixes	Size	Weight	Power	System Speed
	Discrete Logic	low	mid	hard	large	high	?	very fast
	ASIC	high (\$500K / mask set)	very low	hard	tiny - 1 die	very low	low	extremely fast
	Programmable logic – FPGA, PLD	low	mid	easy	small	low	medium to high	very fast
	Microprocessor + memory + peripherals	low to mid	mid	easy	small to med.	low to moderate	medium	moderate
	Microcontroller (int. memory & peripherals)	low	mid to low	easy	small	low	medium	slow to moderate
Software Running on Generic Hardware	Embedded PC	low	high	easy	medium	moderate to high	medium to high	fast

Ejemplo sistema embebido: PC de bicicleta

■ Funciones

- Medir velocidad y distancia.
- Calcular quema de calorías.

■ Restricciones

- Tamaño.
- Costo.
- Potencia y consumo de energía.
- Peso.

■ Entradas

- Sensor de rotación de la rueda.
- Teclado de entrada para modos.

■ Salida

- Pantalla de cristal líquido.

■ MCU de bajo rendimiento

- 8-bits, 10 MIPS.



Ejemplo: Unidad de control motor de gasolina

■ Funciones

- Inyección de combustible, toma de aire.
- Tiempo de chispa.
- Circulación de gases de escape.
- Control electrónico acelerador.

■ Restricciones

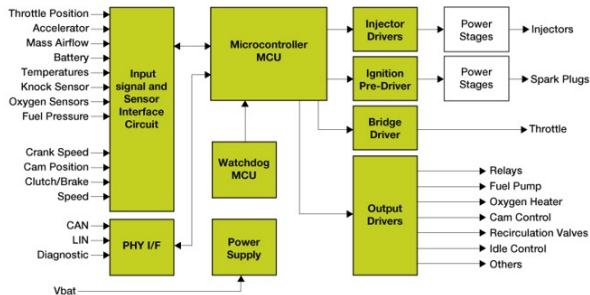
- Fiabilidad en entornos hostiles
- Costo y peso.

■ Muchas entradas y salidas

- Sensores discretos y actuadores
- Interfaz de red en el carro.

■ MCU de alto desempeño

- 32-bits, 3 MB flash, 150-300 MHz.



Beneficios de los sistemas embebidos

- Gran desempeño y eficiencia
 - El software permite proporcionar un control sofisticado.
- Bajo costo
 - Se pueden utilizar componentes menos costosos.
 - Reducción de los costes de fabricación.
 - Costos operativos reducidos.
 - Reducción de los costes de mantenimiento.
- Mas características
 - Muchos no son posibles o prácticos con otros enfoques.
- Mejor confiabilidad
 - Sistema adaptativo que puede compensar fallas.
 - Mejores diagnósticos para mejorar el tiempo de reparación.

Funciones de los sistemas embebidos

- **Sistemas de control en lazo cerrado**
 - Monitoree un proceso, ajuste una salida para mantener el punto de ajuste deseado (temperatura, velocidad, dirección, etc.)
- **Secuenciación**
 - Pasar por diferentes etapas según el entorno y el sistema.
- **Procesamiento de señales**
 - Elimina el ruido, selecciona las funciones de señal deseadas
- **Comunicaciones y redes**
 - Intercambie información de manera confiable y rápida

Bibliografía

- Deitel, P and Deitel, H (2016). *C how to program with an introduction to C++*. Pearson, Ed. 8, pp 1006
- Dean, A (2017). *Embedded systems fundamentals with ARM Cortex-M based Microcontrollers*. ARM Education Media, pp 292.
- Ali, M; Chen, S; Naimi, S; and Naimi, S. (2014) *Freescale ARM Cortex'M Embedded Programming using C Language*. Microdigital, pp 336.