

# ESTRUCTURA DE COMPUTADORES

## GUÍA DOCENTE

### OBJETIVOS Y COMPETENCIAS

Con la teoría y los ejercicios del documento relativo a esta asignatura se pretenden adquirir las competencias básicas relacionadas con la informática, para ello se comienza desde el nivel más bajo tanto en el ámbito del hardware como del software, con el diseño circuital de una unidad aritmético-lógica y la realización de diagramas de flujo y programas en ensamblador.

Las competencias específicas de la asignatura son las siguientes:

- 1- Comprender el lenguaje interno de la máquina y programas en lenguaje ensamblador, incluyendo los mecanismos internos para la gestión de subrutinas
- 2- Analizar el rendimiento del sistema de memoria para evaluar su influencia en las prestaciones de un computador
- 3- Programar aplicaciones de entrada/salida para el control del entorno (programación dirigida por eventos hardware)
- 4- Comprender el funcionamiento de periféricos estándar de un computador
- 5- Analizar diferentes protocolos de buses para determinar su influencia en el rendimiento de las transferencias de datos en el computador
- 6- Comprender la arquitectura interna del ordenador y las funciones de cada unidad en la ejecución de instrucciones, con las consecuencias que ello conlleva en los tiempos de ejecución y ocupación de memoria de los programas

Para poder seguir la asignatura es necesario tener claros algunos conceptos básicos relacionados con los fundamentos físicos y tecnológicos de la informática (polarización, magnetización,...) y conocimientos de electrónica digital (álgebra de Boole, funcionamiento de las puertas lógicas, FFs y circuitos tales como multiplexores, sumadores etc.). Es además interesante tener unos conocimientos básicos de programación: qué es un puntero, qué es una estructura if, for... así como algoritmos básicos de programación. Para el desarrollo de los ejercicios con ordenador en los que se trabaja en ensamblador con las

instrucciones del microprocesador 8085 es suficiente con unos conocimientos muy básicos; para los ejercicios de programación dirigida por eventos (PDE) es necesario haber tenido contacto con algún lenguaje de programación como C o java que facilite la comprensión de los ejemplos que se utilizarán para programar un Arduino. *Hay que tener en cuenta que para estos últimos ejercicios se necesita un hardware que obviamente no es gratuito.*

DESCRIPCIÓN Y TEMARIO
-----------------------

La asignatura está compuesta por seis temas (más un tema 0 en el que se repasan algunas nociones básicas). La teoría se complementa con cuestiones y ejercicios, tanto de “lápiz y papel” como de ordenador.

El temario de la asignatura está compuesto por:

### **T0- Nociones básicas**

*Se repasan conceptos básicos y se complementan con códigos detectores/correctores de error y algunos circuitos digitales integrados MSI comerciales.*

### **T1 Arquitectura von Neumann**

*Se analizan las características de la arquitectura von Neumann en comparación con la Harvard, pasando después a un estudio exhaustivo de los subsistemas que la componen.*

#### 1.1 von Neumann

#### 1.2 Ruta de datos

Unidad aritmética

Unidad lógica

#### 1.3 Unidad de control

Unidad de control cableada de ciclo único

Unidad de control microprogramada de ciclo múltiple

Computadora en canalización

### **T2- Instrucciones de un computador**

*Se definen conceptos sobre la arquitectura del juego de instrucciones y el formato de instrucción. Se ven diferentes ejemplos que permiten interiorizar los conceptos y relacionar el tema actual con el anterior.*

#### 1.1 Formatos de instrucción

1.2 Modos de direccionamiento

1.3 Tipos de instrucciones

### **T3- Subrutinas**

Se presenta el concepto de subrutina y se estudia su gestión. Se pone de manifiesto la importancia de la pila y se estudia la gestión del bloque de activación.

1.1 Definición e instrucciones básicas

1.2 Soporte para el tratamiento de subrutinas (ejecución de la subrutina y gestión del bloque de activación)

1.3 Interrupciones vs llamadas a procedimiento

### **T4- Subsistema de memoria**

*Se clasifican los diferentes tipos de memorias en función de sus principales características. Se analizan los requerimientos de las memorias del computador concluyendo el concepto de jerarquía de memorias: se introducen los conceptos relacionados con la memoria caché y la memoria virtual*

1.1 Características y clasificación de memorias

1.2 Jerarquía de memorias

Memoria caché

Memoria virtual

1.3 Expansión de memorias

### **T5- Subsistema de entrada/salida**

*Se analiza el funcionamiento de varios periféricos y la necesidad del controlador. Se comparan mecanismos de sincronización (encuesta vs interrupción) y de E/S (mapeada vs independiente). Finalmente se analiza el acceso directo a memoria (DMA) y las ventajas que ello supone.*

1.1 Entrada/salida

1.2 Controlador de E/S

1.3 E/S mapeada en memoria vs independiente

1.4 Sincronización

E/S programada

E/S mediante interrupciones

DMA

1.5 Periféricos

## T6- Conexión entre subsistemas: buses

Se define el concepto de bus y la jerarquía de buses analizando las características de diseño. Finalmente se ven ejemplos de buses reales.

1.1 Definiciones, estructura y jerarquía de buses

1.2 Características de diseño

1.3 Ejemplos de buses comerciales

### METODOLOGÍA

COMPETENCIA	RESULTADOS DE APRENDIZAJE	ACTIVIDAD
1, 3	Diagrama del diseño, desarrollo y prueba funcional de un programa en ensamblador. Programación dirigida por eventos hardware	Ejercicios de ordenador
1, 6	Reconocimiento de las características de la arquitectura von Neumann	Ejercicios y teoría
1, 6	Reconocimiento del tipo de unidad de control e identificación de sus partes en base a un esquema circuital	Ejercicios y teoría
1, 6	Desarrollo e implementación de una UAL en base a las especificaciones funcionales dadas	Ejercicios y teoría
5	Interpretación de un cronograma de una operación de bus y extracción de las características del bus a partir del mismo	Ejercicios y teoría
2, 4, 5	Conocer las características de los diferentes tipos de buses, memorias y periféricos, que permitan identificar y seleccionar según necesidad	Ejercicios y teoría
2	Cálculo y diseño para una expansión de memorias usando los circuitos de memoria disponibles. Identificación de la localización física de un dato a partir de la dirección dada	Ejercicios y teoría

CRONOGRAMA DE ACTIVIDADES

CRONOGRAMA DE ACTIVIDADES. ORDEN Y HORAS ESTIMADAS																														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
<b>TEORÍA</b>	T0					T1.1	T1.2						T1.3						T2		8085									
<b>EJERCICIOS</b>			T0							T1.2						T1.3						T2								
<b>ORDENADOR</b>																														Ensamblador 8085
CRONOGRAMA DE ACTIVIDADES. ORDEN Y HORAS ESTIMADAS																														
	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53							
<b>TEORÍA</b>	T3		T4							T5		T6					P.D.E.													
<b>EJERCICIOS</b>							T4									T6														
<b>ORDENADOR</b>																			<i>Arduino (ejemplos)</i>		<i>Propuesta proyecto Arduino</i>									