

# **MODULO I: INTRODUCCION**

## **Tema 1: Introducción a la estructura de computadores**

- 1. Objetivos de la asignatura**
- 2. Niveles de descripción de un computador**
  - 2.1. Nivel Físico
  - 2.2. Nivel Electrónico
  - 2.3. Nivel Lógico
  - 2.4. Transferencia de Registros (RT)
  - 2.5. Arquitectura (lenguaje máquina)
  - 2.6. Software básico (sistema operativo)
  - 2.7. Lenguajes de alto nivel
  - 2.8. Algoritmos
  - 2.9. Aplicaciones
- 3. Estructura básica de un computador convencional**
  - 3.1. Arquitectura von Neumann
  - 3.2. Sistema de interrupciones
  - 3.3. Sistema de memoria caché
  - 3.4. Sistema de memoria virtual
- 4. Evolución histórica: tecnología, estructura y arquitectura**
  - 4.1. Aumento de rendimiento a través de la organización y arquitectura
    - 4.1.1. Paralelismo y Segmentación
    - 4.1.2. Procesadores Segmentados
    - 4.1.3. Procesadores Superescalares
    - 4.1.4. Procesadores VLIW
    - 4.1.5. Procesadores Vectoriales
    - 4.1.6. Multiprocesadores de Memoria Compartida
    - 4.1.7. Multicomputadores
- 5. Lenguajes de descripción hardware.**
  - 5.1. Lenguajes precursores
    - 5.1.1. CDL (Computer Design Language)
    - 5.1.2. DDL (Digital systems Design Language)
    - 5.1.3. AHPL (A Hardware Programming Language)
    - 5.1.4. ISPS (Instruction Set Processor Specifications)
    - 5.1.5. TI-HDL (Texas Instruments-HDL)
  - 5.2. Lenguajes actuales
    - 5.2.1. Verilog
    - 5.2.2. UDL/I
    - 5.2.3. VHDL (VHSIC Hardware Description Language)

## MODULO II: ARQUITECTURA DEL PROCESADOR

### Tema 2: Formato de instrucciones y modos de direccionamiento

1. **Introducción al repertorio de instrucciones**
2. **Formato de las instrucciones: diseño del repertorio**
  - 2.1. **Número de operandos explícitos por instrucción**
  - 2.2. **Forma de almacenar operandos en la CPU**
    - 2.2.1. **Arquitectura de pila**
    - 2.2.2. **Arquitectura de acumulador**
    - 2.2.3. **Arquitectura de registros de propósito general**
      - 2.2.3.1. **Arquitectura registro-registro (carga-almacenamiento).**
      - 2.2.3.2. **Arquitectura registro-memoria**
      - 2.2.3.3. **Arquitectura memoria-memoria**
  - 2.3. **Códigos de operación de longitud fija y variable**
  - 2.4. **Propiedades generales del direccionamiento.**
    - 2.4.1. **Resolución**
    - 2.4.2. **Orden de los bytes en memoria**
      - 2.4.2.1. **Modo big-endian**
      - 2.4.2.2. **Modo little-endian**
    - 2.4.3. **Alineación**
    - 2.4.4. **Espacios de direcciones**
3. **Modos de direccionamiento.**
  - 3.1. **Inmediato.**
  - 3.2. **Directo (memoria o registros)**
  - 3.3. **Indirecto (memoria)**
  - 3.4. **Indirecto (registro)**
  - 3.5. **Modos con desplazamiento.**
    - 3.5.1. **Direccionamiento base más desplazamiento**
    - 3.5.2. **Direccionamiento relativo**
    - 3.5.3. **Direccionamiento indexado**
    - 3.5.4. **Direccionamiento indexado con factor de escala**
  - 3.6. **Modos compuestos(MC 68.X)**
  - 3.7. **Modos de direccionamiento del MIPS R-2000**
  - 3.8. **Modos de direccionamiento del Pentium II**
  - 3.9. **Modos de direccionamiento del ARM**
4. **Soporte de los modos de direccionamiento a los lenguajes de alto nivel**
  - 4.1. **Visibilidad en C y Pascal**
  - 4.2. **Acceso a variables escalares locales o globales (contenido)**
  - 4.3. **Acceso a variables escalares no locales (contenido)**
  - 4.4. **Acceso a variables escalares locales (dirección)**
  - 4.5. **Acceso a variables escalares no locales (dirección)**
  - 4.6. **Acceso a variables de tipo *array* locales (contenido)**
  - 4.7. **Acceso a variables de tipo *array* no locales (contenido)**
  - 4.8. **Acceso a variables de tipo *array* locales (dirección)**
  - 4.9. **Acceso a variables de tipo *record* locales (contenido)**

## **Tema 3: Formato de los datos y clases de instrucciones**

- 1. Tipos de datos y formatos de representación.**
  - 1.1. Números naturales**
    - 1.1.1. Representación binaria
    - 1.1.2. Representación BCD (Decimal Codificado en Binario)
  - 1.2. Números enteros**
    - 1.2.1. Signo y magnitud
    - 1.2.2. Complemento a dos
    - 1.2.3. Complemento a uno
  - 1.3. Números reales**
    - 1.3.1. Coma flotante: Estándar IEEE 754
  - 1.4. Caracteres**
  - 1.5. Compresión de datos Lempel-Ziv (LZ),**
  - 1.6. Tipos de datos en el 68000 y el Pentium II**
- 2. Instrucciones que operan sobre datos.**
  - 2.1. Instrucciones de movimiento o transferencia de datos**
  - 2.2. Instrucciones de desplazamiento y rotación**
  - 2.3. Instrucciones lógicas y de manipulación de bits**
  - 2.4. Instrucciones aritméticas**
  - 2.5. Instrucciones de transformación de datos**
  - 2.6. Instrucciones de entrada/salida**
  - 2.7. Instrucciones de manipulación de direcciones**
  - 2.8. Instrucciones que operan sobre datos en el Pentium II, 68000 y ARM**
- 3. Instrucciones paralelas SIMD (Single Instruction Multiple Data)**
  - 3.1. Tipos de datos: registros MMX**
  - 3.2. Repertorio de instrucciones MMX**
  - 3.3. Ejemplos de codificación MMX**
- 4. Instrucciones de control del flujo de ejecución.**
  - 4.1. Instrucciones de bifurcación condicional**
    - 4.1.1. Gestión de la Condición
    - 4.1.2. Especificación de la dirección
  - 4.2. Subrutinas**
    - 4.2.1. Control de la dirección de vuelta
    - 4.2.2. Paso de parámetros
    - 4.2.3. Preservación del contexto
    - 4.2.4. Variables locales de la subrutina
  - 4.3. Ejemplos: MC 68000, Pentium y ARM**
- 5. Soporte de las instrucciones de control a las construcciones de alto nivel**
  - 5.1. Construcciones condicionales**
  - 5.2. Construcciones iterativas**

## **Tema 4: Rendimiento del procesador.**

- 1. Medidas del rendimiento de un computador**
  - 1.1. Tiempo de ejecución**
    - 1.1.1. Tiempo de respuesta
    - 1.1.2. Tiempo de CPU
    - 1.1.3. Tiempo de CPU utilizado por el usuario.
    - 1.1.4. Tiempo de CPU utilizado por el S.O. Es el tiempo que el S.O. emplea para
  - 1.2. Otros parámetros de rendimiento**
    - 1.2.1. MIPS (Millones de Instrucciones Por Segundo)
    - 1.2.2. MFLOPS (Millones de operaciones en punto FLOtante Por Segundo)
    - 1.2.3. Productividad (throughput)
    - 1.2.4. Ganancia de velocidad (*speedup*): Ley de Amdahl
    - 1.2.5. Rendimiento medio armónico
- 2. Patrones de medida (Benchmarks)**
  - 2.1. LINPACK
  - 2.2. SPEC: (System Performance and Evaluation Cooperative)
  - 2.3. TPC (Transaction Processing Performance Council)
  - 2.4. Otros benchmarks
    - 2.4.1. Whetstone
    - 2.4.2. Dhrystone
- 3. Influencia en el rendimiento de las alternativas de diseño**
  - 3.1. Tipo de elementos de memoria en la CPU
  - 3.2. Referencias a memoria en instrucciones ALU
  - 3.3. Orden de ubicación de los datos
  - 3.4. Alineamiento de datos
  - 3.5. Direccionamientos
  - 3.6. Datos operando
  - 3.7. Operaciones
  - 3.8. Sentencias de salto condicional
  - 3.9. Llamadas a procedimientos (subrutinas)
  - 3.10. Registros de propósito general: ventanas de registros
- 4. Influencia de los compiladores de lenguajes de alto nivel**
  - 4.1. Optimización de registros
- 5. Procesadores RISC y CISC**
  - 5.1. Características comparativas
  - 5.2. Segmentación
  - 5.3. Dependencias
  - 5.4. Ejemplo: DLX
    - 5.4.1. Formatos de las instrucciones
    - 5.4.2. Repertorio de instrucciones
    - 5.4.3. Eficiencia de los procesadores RISC

## **MODULO III: SUBSISTEMA DE MEMORIA**

### **Tema 5: Organización de la memoria: memoria principal.**

- 1. Características generales de las memorias**
  - 1.1. Método de acceso
  - 1.2. Soporte físico
  - 1.3. Alterabilidad
  - 1.4. Volatilidad con la fuente de energía
  - 1.5. Duración de la información
  - 1.6. Proceso de lectura
  - 1.7. Ubicación en el computador
  - 1.8. Parámetros de velocidad
  - 1.9. Unidades de transferencia
  - 1.10. Jerarquía de las unidades de memoria de un computador
- 2. Organización interna de la memoria principal.**
  - 2.1. Organización 2D
  - 2.2. Organización 3D
- 3. Diseño de memorias**
  - 3.1. Ampliación del número de bits de la palabra de memoria
  - 3.2. Ampliación del número de palabras de memoria
  - 3.3. Ampliación de la longitud y el número de palabras de memoria
  - 3.4. Ubicación en el espacio de direcciones.
    - 3.4.1. Decodificación parcial y total
  - 3.5. Disposición de los módulos de memoria
    - 3.5.1. SIMM (Single In-line Memory Module).
    - 3.5.2. DIMM (Dual In-line Memory Module).
- 4. Detección y corrección de errores.**
  - 4.1. Códigos detectores de errores
  - 4.2. Códigos correctores de errores (Hamming)
- 5. Memoria entrelazada.**
  - 5.1. Entrelazado de orden superior
  - 5.2. Entrelazado de orden inferior
  - 5.3. Memoria entrelazada con organización S
  - 5.4. Memoria entrelazada con organización C

## Tema 6: Memoria Caché.

1. Principios básicos de funcionamiento de la memoria caché
  - 1.1. Localidad de referencia: temporal y espacial
  - 1.2. Tasa de aciertos y tasa de fallos:
2. Elementos de diseño.
  - 2.1. Función de correspondencia
    - 2.1.1. Correspondencia directa
    - 2.1.2. Correspondencia asociativa
    - 2.1.3. Correspondencia asociativa por conjuntos
  - 2.2. Algoritmos de sustitución
    - 2.2.1. Aleatoria
    - 2.2.2. LRU(Least Recently Used)
    - 2.2.3. FIFO (First In First Out)
    - 2.2.4. LFU(Least Frequently Used)
  - 2.3. Política de escritura
    - 2.3.1. Escritura directa o inmediata (*write through*)
    - 2.3.2. Postescritura (*copy back*)
    - 2.3.3. Asignación en escritura (*write allocate*)
    - 2.3.4. No asignación en escritura (*No write allocate*)
  - 2.4. Política de búsqueda
    - 2.4.1. Por demanda
    - 2.4.2. Anticipativa (prebúsqueda)
  - 2.5. Clasificación de los fallos caché
3. Factores que determinan el rendimiento de la memoria caché.
  - 3.1. Reducción de la tasa de fallos
    - 3.1.1. Aumento del tamaño del bloque
    - 3.1.2. Aumento de la asociatividad
    - 3.1.3. Utilización de una caché de víctimas
    - 3.1.4. Cachés pseudoasociativas
    - 3.1.5. Prebúsqueda de instrucciones y datos
    - 3.1.6. Prebúsqueda controlada por el compilador
    - 3.1.7. Optimizaciones del compilador
      - 3.1.7.1. Fusión de *arrays*
      - 3.1.7.2. Fusión de bucles
      - 3.1.7.3. Intercambio de bucles
      - 3.1.7.4. Descomposición en bloques
  - 3.2. Reducción de la penalización de fallos
    - 3.2.1. Prioridad a los fallos de lectura frente a los de escritura
    - 3.2.2. Utilización de sub-bloques dentro de un bloque
    - 3.2.3. Utilización de un segundo nivel de caché
  - 3.3. Reducción del tiempo de acierto
    - 3.3.1. Cachés pequeñas y simples
    - 3.3.2. Evitar traducción de direcciones durante la indexación de las cachés
    - 3.3.3. Escrituras segmentadas para aclarar los aciertos de escritura
4. Ejemplos de sistemas de memoria caché.

## Tema 7: Memoria Virtual.

1. **Gestión de memoria**
  - 1.1. Solapamiento (overlay)
  - 1.2. Reubicación
  - 1.3. Paginación
  - 1.4. Protección
  - 1.5. Compartición
2. **Memoria virtual**
  - 2.1. Memoria virtual paginada
    - 2.1.1. Tablas de páginas de varios niveles
    - 2.1.2. Tabla de páginas invertida HASH
    - 2.1.3. Buffer de traducción anticipada (TLB)
  - 2.2. Interacción entre la memoria virtual y la memoria cache
    - 2.2.1. Acceso paralelo a TLB y caché
    - 2.2.2. Caches con direcciones virtuales
  - 2.3. Múltiples espacios virtuales
  - 2.4. Políticas de búsqueda (*fetch*)
    - 2.4.1. Prebúsqueda
    - 2.4.2. Búsqueda por demanda
  - 2.5. Políticas de sustitución (*replacement*)
    - 2.5.1. Aleatoria
    - 2.5.2. FIFO (First In First Out)
    - 2.5.3. Reloj (FINUFO: First In Not Used First Out)
    - 2.5.4. LRU (Least Recently Used)
    - 2.5.5. Optima (MIN)
3. **Memoria virtual segmentada**
  - 3.1. Políticas de ubicación (*placement*) para memorias segmentadas
    - 3.1.1. Mejor ajuste (*best fit*)
    - 3.1.2. Peor ajuste (*worst fit*)
    - 3.1.3. Primer ajuste (*first fit*)
4. **Memoria con segmentos paginados**
5. **Ejemplo de sistema de memoria virtual: procesador Pentium II**
  - 5.1. Mecanismo de segmentación
  - 5.2. Mecanismo de paginación

## MODULO IV: SUBSISTEMA DE ENTRADA/SALIDA

### Tema 8: Organización de la Entrada/salida

1. **Funciones implicadas en las operaciones de entrada/salida**
2. **Estructura del sistema de E/S: módulos de e/s y controladores de dispositivos**
  - 2.1. Módulos de E/S
  - 2.2. Controlador de dispositivo (periférico)
  - 2.3. Comunicación entre el módulo de E/S y el controlador de dispositivo periférico
3. **Mecanismos básicos de e/s: sincronización**
4. **E/S controlada por programa**
  - 4.1. Ejemplos de E/S controlada por programa
  - 4.2. E/S serie y paralelo
    - 4.2.1. E/S serie asíncrona
    - 4.2.2. E/S serie sincrona
    - 4.2.3. Ejemplo de E/S serie: ACIA (Asynchronous Communicatios Interface Adapter)
    - 4.2.4. Ejemplo de E/S paralelo: VIA (Versatil Interface Adapter)
5. **E/S por interrupción: gestión de interrupciones**
  - 5.1. Tipos de sistemas de interrupciones
    - 5.1.1. Interrupciones autovectorizadas
    - 5.1.2. Interrupciones vectorizadas:
  - 5.2. Prioridades
  - 5.3. Enmascaramiento de interrupciones
  - 5.4. Anidamiento de interrupciones
  - 5.5. Ejemplos de sistemas de interrupciones
    - 5.5.1. Sistema de interrupciones del microprocesador MC 6809
    - 5.5.2. Sistema de interrupciones del microprocesador MC 68000
      - 5.5.2.1. Líneas de interrupción
      - 5.5.2.2. Enmascaramiento
      - 5.5.2.3. Vectores de excepción
      - 5.5.2.4. Ejemplo de conexión de periféricos
    - 5.5.3. Sistema de interrupciones del microprocesador ARM
6. **E/S por acceso directo a memoria (DMA): motivación**
  - 6.1. Estructura y funcionamiento de un controlador de DMA
    - 6.1.1. Bus único (DMA independiente)
    - 6.1.2. Integración de funciones DMA-E/S
    - 6.1.3. Bus de E/S conectado al DMA
7. **Procesadores de E/S: tipos y estructura.**
  - 7.1. Canal multiplexor
  - 7.2. Canal selector
  - 7.3. Canal multiplexado por bloques
  - 7.4. Programas de operación de los procesadores de E/S.

## **Tema 9: Dispositivos periféricos y controladores**

- 1. Discos magnéticos**
  - 1.1. Grabación de información sobre superficies magnéticas**
  - 1.2. Métodos de codificación sobre superficies magnéticas**
  - 1.3. Formato de grabación**
  - 1.4. Comprobación de Redundancia Cíclica (CRC)**
    - 1.4.1. Aritmética módulo 2**
    - 1.4.2. Codificación**
    - 1.4.3. Representación de los errores**
    - 1.4.4. Interpretación polinómica de los códigos CRC**
      - 1.4.4.1. Error simple**
      - 1.4.4.2. Errores dobles**
      - 1.4.4.3. Errores en ráfaga:**
    - 1.4.5. Polinomios generadores estándar**
    - 1.4.6. Implementación hardware**
  - 1.5. Parámetros de rendimiento de un disco magnético**
    - 1.5.1. Tiempo de búsqueda**
    - 1.5.2. Retardo rotacional**
    - 1.5.3. Tiempo de acceso**
  - 1.6. Dispositivo floppy**
    - 1.6.1. Estructura de bloques del dispositivo**
    - 1.6.2. Formato de datos**
    - 1.6.3. Estructura y funcionamiento del controlador de dispositivo**
- 2. Discos ópticos**
  - 2.1. CD-ROM**
    - 2.1.1. Proceso de escritura (grabación)**
    - 2.1.2. Proceso de lectura**
    - 2.1.3. Codificación de la información**
  - 2.2. DVD (Digital Versatil Discs)**
  - 2.3. WORM (Write Once, Read Many times)**
  - 2.4. Discos magnetoópticos o WMRA (Write Many, Read Always)**
    - 2.4.1. Proceso de escritura**
    - 2.4.2. Proceso de lectura**
    - 2.4.3. Proceso de grabación**
- 3. Monitores de tubo de rayos catódicos (CRT)**
  - 3.1. Principios de funcionamiento**
  - 3.2. Sincronismo**
  - 3.3. Generación de caracteres**
  - 3.4. Controlador**
- 4. Pantallas de cristal líquido (LCD-TFT)**
- 5. Monitores de color**
- 6. Unidades de Procesamiento Gráfico (GPU)**
- 7. Impresoras**
  - 7.1. Impresora de martillo**
  - 7.2. Impresora de chorro de tinta**
  - 7.3. Impresora láser**

## 8. Ratón

# Tema 10: Buses de comunicación

## 1. Estructura de un bus

### 1.1. Líneas

- 1.1.1. Líneas de Direcciones
- 1.1.2. Líneas de Datos
- 1.1.3. Líneas de control

### 1.2. Direccionalidad de buses

- 1.2.1. Líneas unidireccionales
- 1.2.2. Líneas bidireccionales

## 2. Protocolos de transferencia

- 2.1. Síncronos
- 2.2. Asíncronos
- 2.3. Semisíncronos
- 2.4. Ciclo partido

## 3. Protocolos de arbitraje

- 3.1. Protocolo de encadenamiento (*daisy chaining*) de dos señales
- 3.2. Protocolo de encadenamiento (*daisy chaining*) de tres señales
- 3.3. Protocolo de encadenamiento (*daisy chaining*) de cuatro señales
- 3.4. Protocolo con concesión por encuesta (*polling*)
- 3.5. Protocolo con señales independientes
- 3.6. Protocolo distribuido

## 4. Jerarquía de buses

- 4.1. Bus del sistema (*backplane*)
- 4.2. Buses locales
- 4.3. Buses de E/S o de expansión

## 5. Buses normalizados

### 5.1. Bus PCI (*Peripheral Component Interconnect*)

- 5.1.1. Estructura
- 5.1.2. Ordenes
- 5.1.3. Transferencia de datos
- 5.1.4. Arbitraje

### 5.2. Bus USB (*Universal Serial Bus*)

- 5.2.1. Arquitectura
- 5.2.2. Protocolo
- 5.2.3. Modos de transmisión

### 5.3. Bus SCSI (*Small Computer System Interface*)

- 5.3.1. Líneas de señal
- 5.3.2. Operación (temporización)
- 5.3.3. Mensajes

### 5.4. Arquitectura del standard InfiniBand

## **Bibliografía:**

Organización y arquitectura de computadores  
W. Stalling, Prentice Hall 2000, 2007

Estructura y diseño de computadores. Interficie circuitería/programación  
D.A. Patterson y J.L. Hennessy, Reverté, 2000

Computer Architecture. A Quantitative Approach  
J.L. Hennessy and D.A. Patterson 2007