



**Tema:**

**INTRODUCCIÓN A LOS  
PROCESADORES DIGITALES DE SEÑAL.**

**Raúl Mateos Gil**



## ÍNDICE



### ◆ Introducción.

- Generalidades sobre sistemas de tratamiento digital de señal.
- Alternativas de diseño.

### ◆ Características generales de los Procesadores Digitales de Señal.

- Arquitectura de la CPU.
- Juego de instrucciones.
- Arquitectura de memoria.
- Periféricos integrados e interfaces de I/O.

### ◆ Formatos de datos y ancho de palabra de datos.

- Formatos de coma fija y coma flotante.

### ◆ Tipos de Procesadores Digitales de Señal.

- Según el tipo de aritmética utilizada.
- Según el paralelismo del dispositivo.

### ◆ Criterios de selección del DSP.

### ◆ Familias de DSPs más representativas.

### ◆ Aplicaciones.

### ◆ Conclusiones.

### ◆ Bibliografía.



## TÉCNICAS DE TRATAMIENTO DIGITAL vs ANALÓGICAS.

### VENTAJAS.

- Reprogramabilidad.
- Estabilidad, repetitividad y comportamiento previsible.
- Funciones especiales.
- Mayor inmunidad en la transmisión y en el almacenamiento de la información.

### INCONVENIENTES.

- Costes no competitivos en sistemas sencillos.
- Dificultad de implementación en aplicaciones de alta frecuencia.

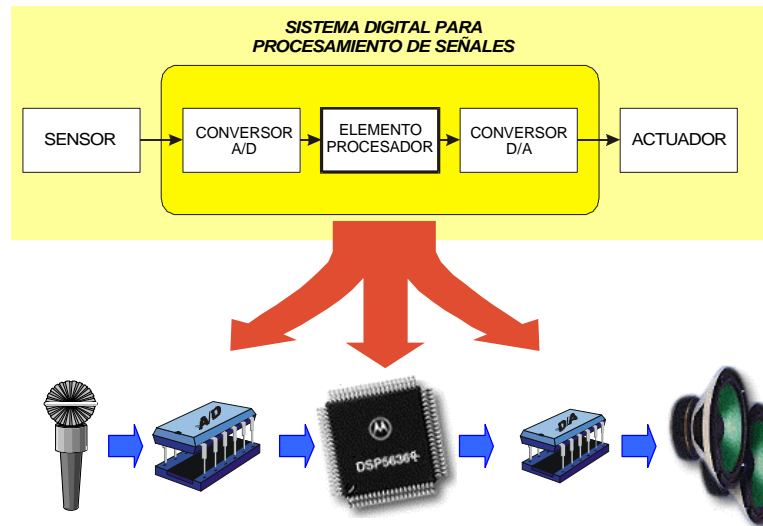
Las técnicas de procesamiento digital de señal plantean grandes ventajas respecto a las analógicas. Entre ellas cabe destacar:

- **Programabilidad** que permite utilizar un mismo sistema para implementar varios algoritmos.
- **Estabilidad, repetitividad y comportamiento previsible.** El comportamiento de un sistema analógico se ve afectado por las variaciones con la temperatura de los valores efectivos de los componentes con que se construyen (resistencias, condensadores, amplificadores operacionales). A esto hay que añadir las derivas por envejecimiento de los componentes, y las tolerancias de éstos. Si se requiere una gran precisión de operación en estos sistemas es necesario realizar un proceso de calibración. Puesto que la salida de un sistema de procesamiento digital de señal no depende de factores ambientales o variaciones en el valor nominal de los componentes, es posible obtener sistemas con una respuesta exacta, conocida y sin variación de un equipo a otro.
- **Funciones especiales.** Existen algunas técnicas de procesamiento de señal que no se pueden implementar con sistemas analógicos. Un ejemplo de estos son los filtros de fase lineal. Sin embargo un filtro FIR digital con simetría conjugada par de sus coeficientes respecto al punto medio es un sistema de fase lineal.
- **Mayor inmunidad al ruido en la transmisión y almacenamiento de la información.** Esto es evidente, basta comparar las prestaciones de un disco de vinilo con un CD.

En cuanto a inconvenientes se pueden citar los siguientes: para aplicaciones sencillas, la complejidad de un sistema analógico es menor, lo que tiene una traducción directa en el coste de éste. En aplicaciones en tiempo real para tratamiento digital de señal, entre muestra y muestra hay que realizar todos los cálculos que impone el algoritmo. Para señales de gran ancho de banda, la frecuencia de muestreo puede ser tan elevada que la potencia de cálculo requerida haga imposible su tratamiento mediante técnicas digitales.



## DIAGRAMA DE BLOQUES.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

3

La estructura típica de un sistema para tratamiento digital de señal responde al diagrama de bloques que aparece en esta transparencia.

Un conversor A/D convierte la señal analógica en una secuencia numérica. Esta señal analógica suele obtenerse de un sensor que transforma una magnitud física en una señal eléctrica. Estas muestras llegan a un elemento procesador en el que se implementa el algoritmo de tratamiento digital. La salida de este procesador se introduce a un conversor D/A para obtener de nuevo una señal analógica. Esta señal analógica de salida puede transformarse de nuevo en una magnitud física por medio de un actuador.

Como ejemplo se podría mencionar un filtro para eliminar ruido de la señal obtenida de un micrófono. Este último elemento transforma en una señal analógica las presiones acústicas de la voz. La salida del micrófono una vez adaptado su nivel se aplica a un conversor A/D. Las muestras de salida del A/D se tratan en un elemento procesador según el algoritmo implementado. El resultado de este procesamiento se aplica a un conversor D/A para obtener de nuevo una señal analógica, que una vez amplificada ataca a un altavoz.

En este tema se realizará un estudio de este elemento procesador, comenzando por las características que debe reunir dicho dispositivo.



## CARACTERÍSTICAS DEL ELEMENTO PROCESADOR (I)

OPERACIÓN BÁSICA EN EL PROCESAMIENTO DIGITAL DE SEÑAL:  
PRODUCTO ACUMULATIVO u OPERACIÓN MAC

$$y[m] = \sum_{k,n} a[k] \cdot b[n]$$

### TRATAMIENTO DIGITAL DE SEÑAL CLÁSICO

- FILTRADO: 
$$y[n] = \sum_{k=0}^N b[k] \cdot x[n-k] + \sum_{k=1}^M a[k] \cdot y[n-k]$$
- ANÁLISIS ESPECTRAL: 
$$X[n] = \frac{1}{N} \sum_{k=0}^{N-1} x[k] \cdot e^{-j \frac{2\pi kn}{N}}$$
- CÁLCULO MATRICIAL: 
$$\mathbf{AB} = \begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & \dots & b_{1r} \\ \vdots & \ddots & \vdots \\ b_{m1} & \dots & b_{mr} \end{bmatrix} = \begin{bmatrix} c_{11} & \dots & c_{1r} \\ \vdots & \ddots & \vdots \\ c_{n1} & \dots & c_{nr} \end{bmatrix} \quad c_{ik} = \sum_{j=1}^m a_{ij} \cdot b_{jk}$$

Las características del elemento procesador vendrán impuestas por los requerimientos de las aplicaciones en que se ha de utilizar.

Considerando la naturaleza de las operaciones que es preciso realizar en aplicaciones de tratamiento digital de señal, la operación básica es el producto acumulativo de dos secuencias. A esta operación se le suele denominar operación MAC (multiply and accumulate).

Esto queda puesto de manifiesto en aplicaciones como: filtrado, análisis espectral, correlaciones, etc. Los factores del producto pueden ser, dependiendo del caso, muestras de una señal, coeficientes de un filtro o constantes precalculadas (tablas de senos y cosenos).

Las expresiones de la salida del sistema para los casos citados, pueden considerarse como una particularización del producto de una matriz fila por una matriz columna. Cada elemento de la matriz producto se obtiene multiplicando las componentes de una fila por una columna.

Además, estas aplicaciones son tareas de tiempo real, por lo que el elemento procesador deberá tener una arquitectura que permita realizar las operaciones MAC a gran velocidad.



## CARACTERÍSTICAS DEL ELEMENTO PROCESADOR (II).

### CONDICIONANTES DE LA APLICACIÓN:

- TIEMPO REAL.
- ALGORITMOS DE CÁLCULO INTENSIVO.
- GRAN CANTIDAD DE DATOS A PROCESAR.
- SISTEMA FLEXIBLE.

### CARACTERÍSTICAS DEL SISTEMA:

- ALTA VELOCIDAD.
- OPERACIONES MAC RÁPIDAS.
- MANEJO CÓMODO DE DATOS.
- PROGRAMABILIDAD.

En resumen, si consideramos los condicionantes que impone este tipo de aplicaciones, las características genéricas que debería reunir el elemento procesador son:

! Deben ser dispositivos de gran velocidad para poder procesar las muestras de entrada en tiempo real.

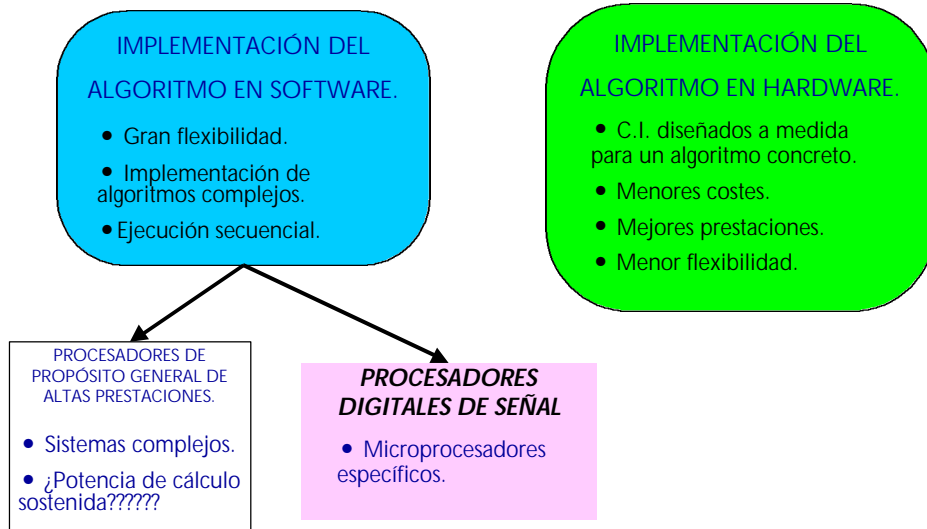
! Los algoritmos de tratamiento digital de señal tienen una elevada carga computacional en los que predomina la operación MAC, por lo que este elemento procesador debe poseer una gran potencia de cálculo.

! Dado el gran volumen de datos a procesar deben permitir un cómodo manejo de éstos.

! Han de ser programables para posibilitar la implementación de distintos tipos de algoritmos de tratamiento digital de señal.



## ALTERNATIVAS DE DISEÑO.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

6

Los sistemas de tratamiento digital de señal se suelen caracterizar por la utilización de un algoritmo que especifica las operaciones a realizar con los datos. La implementación de este algoritmo se puede realizar mediante un sistema programado o en hardware a medida.

En un sistema programado el algoritmo se implementa como un programa que se ejecuta secuencialmente. Esto se traduce en:

- ! Una gran flexibilidad al poder utilizar un mismo sistema para implementar varios algoritmos.
- ! Los algoritmos pueden alcanzar un alto grado de complejidad.
- ! Sin embargo el hecho de que la ejecución del programa se realice de forma secuencial empeora el rendimiento.

Dentro de la implementación del algoritmo en software nos podemos encontrar dos alternativas. La primera de ellas consiste en la utilización de microprocesadores de propósito general de altas prestaciones, como por ejemplo Pentium de Intel, PowerPc de Motorola, etc. Los sistemas basados en estos procesadores son de gran complejidad al requerir toda una serie de periféricos y elementos externos para su implementación. Además, estas altas prestaciones no siempre se traducen en una elevada potencia de cálculo, o al menos en una ejecución rápida de instrucciones de tipo MAC, debido a las restricciones en la ejecución de operaciones en coma flotante.

Sin embargo, a medida que las aplicaciones de tratamiento digital de señal se empiezan a imponer dentro de lo que podríamos considerar informática de consumo, estos microprocesadores han comenzado a adaptarse a la nueva situación. Un ejemplo de esto es la extensión MMX de Intel.

La otra alternativa para la implementación del algoritmo en software son los procesadores digital de señal. Se trata de microprocesadores con una arquitectura específicamente diseñada para estas aplicaciones. Son dispositivos con unas prestaciones superiores y de menor coste que los anteriormente mencionados. Además disponen de herramientas de desarrollo flexibles que permiten acelerar el proceso de diseño de este tipo de aplicaciones. En este tema se realizará un estudio detallado de este tipo de microprocesadores.

La implementación de algoritmos en hardware se realiza mediante circuitos integrados diseñados a medida, integrando en el dispositivo sólo aquellas funciones necesarias para la aplicación concreta a que va destinados, lo que se traduce en un menor coste para grandes volúmenes de producción y unas mejores prestaciones.

El hardware a medida puede tomar múltiples formas. Una de ellas son los dispositivos standard para aplicaciones específicas o ASSPs. Como su nombre indica se trata de circuitos integrados que implementan en hardware un algoritmo concreto. Esto les permite procesar señales de alta frecuencia, pero al mismo tiempo dificulta el diseño del dispositivo por lo que los algoritmos implementados suelen ser sencillos, con una estructura regular, y el número de bits de los datos con que se operan reducidos. Un ejemplo de ASSPs son los filtros digitales programables, los cuales integran un conjunto de multiplicadores hardware que operan en paralelo para implementar un filtro FIR. Debido a la implementación hardware el número de coeficientes del filtro suele ser reducido. Los coeficientes se programan mediante unos registros internos a los cuales se accede mediante un interface con microprocesador. Dispone de sendos puertos para recibir las muestras de entrada y entregar las muestras de salida.

En otras ocasiones, este hardware a medida no es otra cosa que un DSP fabricado con una máscara ROM que contiene un algoritmo concreto.





## ÍNDICE



- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
  - Arquitectura de la CPU.
  - Juego de instrucciones.
  - Arquitectura de memoria.
  - Periféricos integrados e interfaces de I/O.
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**



## CARACTERÍSTICAS GENERALES DE LOS DSPs.

- ARQUITECTURA DE LA CPU.
  - ◆ Ejecución rápida de operaciones MAC.
  - ◆ Modos de direccionamiento especializados para manejo de buffers.
- JUEGO DE INSTRUCCIONES.
  - ◆ Control eficiente de bucles para algoritmos DSP iterativos.
- ARQUITECTURA DE MEMORIA.
  - ◆ Múltiples accesos simultáneos a memoria.
- PERIFÉRICOS INTEGRADOS E INTERFACES DE I/O.
  - Gran interacción con el mundo exterior.

Los DSPs reúnen una serie de elementos diferenciales que los caracterizan y diferencian del resto de los procesadores. Estas características se manifiestan en:

! La arquitectura de la CPU, que dispone de recursos que posibilitan la realizaciones de las operaciones MAC de forma rápida y la utilización de modos de direccionamiento especiales para el manejo de buffers.

! Un juego de instrucciones optimizado para aplicaciones de tratamiento digital de señal, entre las que destacan aquellas que le permiten realizar un control eficiente de bucles.

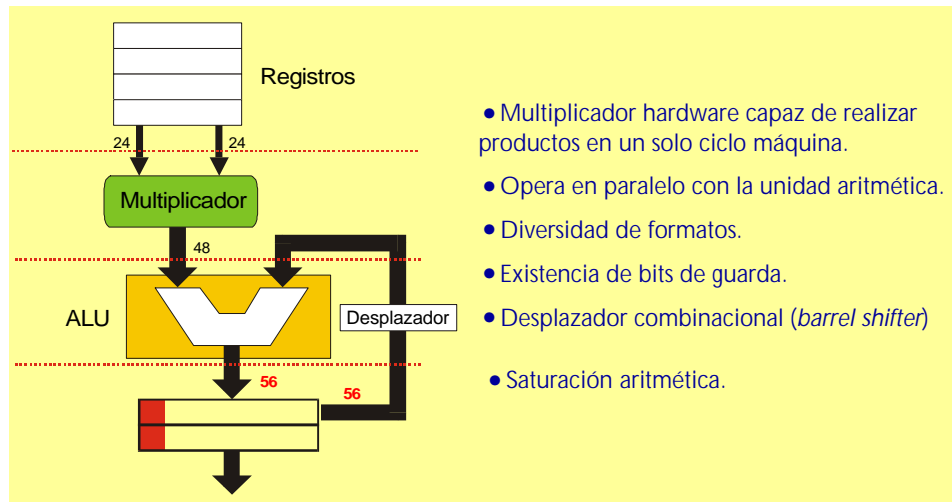
! Una arquitectura de memoria que le permita obtener las instrucciones y datos a procesar al ritmo que los demanda la CPU.

! La inclusión en el propio dispositivo de un conjunto de periféricos que le permitan comunicarse con el exterior.



## ARQUITECTURA DE LA CPU (I).

### UNIDAD MAC



- Multiplicador hardware capaz de realizar productos en un solo ciclo máquina.
- Opera en paralelo con la unidad aritmética.
- Diversidad de formatos.
- Existencia de bits de guarda.
- Desplazador combinacional (*barrel shifter*)
- Saturación aritmética.

Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

10

Para incrementar la velocidad de trabajo y optimizar la implementación de los algoritmos de tratamiento digital de señal, la CPU de los DSPs dispone de unidades computacionales específicas que pueden trabajar en paralelo. Entre ellas cabe destacar las unidades MAC, que posibilitan la ejecución de este tipo de operaciones en un solo ciclo de instrucción.

El elemento básico de esta unidad MAC es el multiplicador hardware (multiplicador cableado) que es capaz de realizar el producto de dos números en un solo ciclo máquina. La acumulación de los productos se realiza en la ALU, también en un solo ciclo máquina. Para realizar la operación MAC ambos elementos operan en paralelo, siguiendo un proceso segmentado en el que mientras el multiplicador realiza el producto de dos operandos la ALU acumula el resultado del producto anterior. Esta unidad MAC puede operar con números enteros o en coma flotante.

Al repetir las operaciones de acumulación de los productos se puede dar una situación de rebose obteniéndose resultados erróneos. El rebose se debe a la imposibilidad de representar el resultado de salida con el número de bits disponibles. Para evitar esta situación se sobredimensiona el número de bits de la ALU añadiendo a los necesarios para realizar la suma de los términos producto un cierto número de bits extras denominados **bits de guarda**. Así, por ejemplo, en un DSP que opera con datos de 24 bits el producto de dos números proporciona un resultado de 48 bits que será acumulado con los productos anteriores. Para evitar situaciones de rebose, se sobredimensiona la ALU añadiendo 8 bits de guarda, con lo que su tamaño final será de 56 bits. Este sobredimensionamiento de la ALU se aplica también al registro o registros donde se almacena el valor acumulado.

Otra estrategia para evitar las situaciones de rebose consiste en realizar el escalado en los operandos de entrada o los resultados intermedios antes de sumarlos al valor acumulado. Este escalado se realiza mediante un desplazamiento aritmético a derechas de un determinado número de bits del producto obtenido. Algunos DSPs que no disponen de bits de guarda suplen esta deficiencia mediante la capacidad de desplazar el contenido del registro donde se almacena el producto, antes de sumarlo, sin necesidad de ciclos de instrucción adicionales. Si bien este desplazamiento supone una pérdida de precisión en los cálculos, su incidencia es despreciable a menos que se realice un fuerte escalado o bien el número de productos a acumular sea muy elevado.

Una característica distintiva de los DSPs es que estos desplazadores están implementados mediante lógica combinatorial (a esta estructura se le denomina barrel shifter) frente a la estructura clásica de un registro de desplazamiento secuencial de otros tipos de procesadores. Esto permite realizar desplazamientos en un solo ciclo de instrucción independientemente del número de bits a desplazar.

Este escalado puede provocar que los valores pequeños no se puedan representar con suficiente fidelidad. Otra alternativa consistiría en utilizar técnicas de **saturación aritmética**.

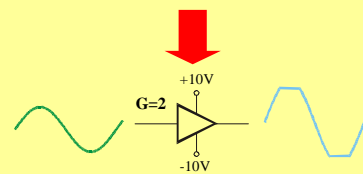
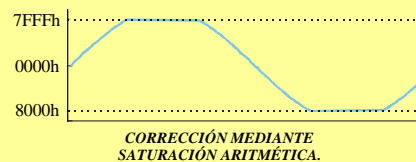
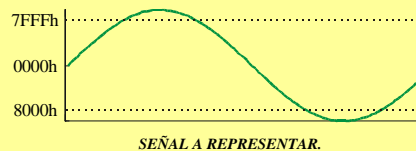


## ARQUITECTURA DE LA CPU (II).

### SATURACIÓN ARITMÉTICA.

Ante situaciones de rebose, muestra a la salida el máximo valor representable, reproduciendo el comportamiento de los sistemas analógicos.

*EJEMPLO DE REPRESENTACIÓN EN C-2 CON 16 BITS.*



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

12

Una situación de rebose se manifiesta cuando al sumar dos números positivos se obtiene un resultado negativo y viceversa. Los DSPs que utilizan técnicas de saturación aritmética incorporan circuitería que al detectar una situación de rebose durante la suma de dos números positivos sustituye la salida errónea por el máximo valor positivo a representar. De igual manera si la situación de rebose se da durante la suma de dos valores negativos proporciona como salida el máximo valor negativo a representar.

El resultado sigue siendo incorrecto pero el error cometido en la operación es menor que en caso de rebose. De esta forma se consigue un comportamiento similar al de un circuito analógico en saturación. La saturación aritmética se puede realizar mediante una instrucción especial o automáticamente. En este último caso, esta facilidad se activa mediante uno de los bits del registro de estado. Algunos fabricantes denominan limitador al módulo que implementa la saturación aritmética.

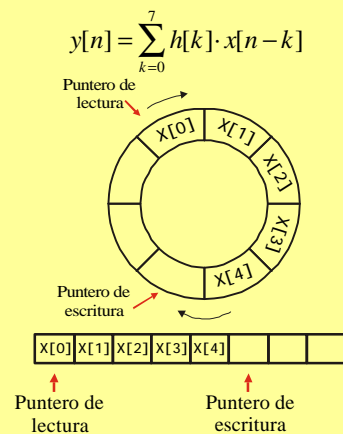


## ARQUITECTURA DE LA CPU (III).

### UNIDAD GENERADORA DE DIRECCIONES DE DATOS (DAG).

ALU ESPECÍFICA PARA GENERACIÓN DE DIRECCIONES DE OPERANDOS QUE TRABAJA EN PARALELO CON EL RESTO DE LAS UNIDADES FUNCIONALES.

- Tiene asociados un gran número de registros para direccionamiento indirecto (PUNTEROS).
- Modos especiales de direccionamiento para procesamiento digital:
  - ◆ Direccionamiento circular (modular).
  - ◆ Direccionamiento para realizar de forma rápida la transformada discreta de Fourier.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

13

Otra característica que permite aumentar la velocidad de procesamiento aritmético en los DSP es la disponibilidad de una o más **unidades generadoras de direcciones de datos**, que realizan los cálculos de la nueva dirección necesaria para el acceso a los operandos. Estas unidades constan de una unidad aritmética específica que opera en paralelo con el resto de las unidades funcionales y un conjunto de registros que proporcionarán la dirección base y el desplazamiento necesarios para el cálculo de la nueva dirección.

Muchas de las aplicaciones de tratamiento digital de señal necesitan gestionar un buffer con estructura FIFO donde se van almacenando las muestras que llegan del exterior o de cálculos anteriores. En la gestión del movimiento de los datos dentro y fuera del buffer el programador mantiene un puntero de lectura y otro de escritura, los cuales se suelen almacenar en registros de direcciones. El puntero de lectura apunta a la posición de memoria que almacena el próximo dato que se va a leer del buffer, mientras que el puntero de escritura apunta a la posición de memoria donde se va a almacenar el siguiente dato de entrada. Cada vez que se realiza una operación de lectura o escritura, el puntero correspondiente avanza una posición, debiendo comprobar el programador si dicho puntero ha llegado a la última posición de memoria del buffer, inicializándose en tal caso para que apunte al comienzo del buffer. Esta comprobación de si un puntero ha llegado al final del buffer y su inicialización en caso afirmativo consume tiempo.

Para solventar este problema las unidades DAG de muchos DSPs realizan de forma automática la comprobación del puntero del buffer y su ajuste relativo a la dirección de comienzo, si es preciso, después de cada cálculo de dirección. Esta característica se denomina **aritmética modular**. El término aritmética modular hace referencia a que el resultado de salida se limita a un determinado rango, de manera que si una operación de suma o resta superará tal rango, se obtendría el valor inicial de dicho rango.

Por último, señalar que las DAGs de muchos DSPs permiten implementar modos de direccionamiento especialmente pensados para realización de la transformada discreta de Fourier de una forma rápida.



## JUEGO DE INSTRUCCIONES.

### Operaciones poco variadas y repetitivas

- Instrucciones sencillas y compactas (RISC):
  - ◆ Ciclo de instrucción rápido.
  - ◆ Precisa menos espacio de memoria.
- Optimizadas para algoritmos de DSP:
  - ◆ Uso del paralelismo interno de la CPU.

MAC (ARO)+, (AR1)+, RO

- Hincapié en el tratamiento de bucles:

MOVE #16, R1

bucle: MAC (ARO)+, (AR1)+, RO

DEC R1

JNE bucle

- Unidades de repetición de instrucciones.

RPT #16

MAC (ARO)+, (AR1)+, RO

El juego de instrucciones es un factor clave a la hora de determinar no sólo las aplicaciones a que se puede destinar un procesador, sino también cuándo su uso es natural y eficiente. Las instrucciones controlan cómo se operan con los datos en la CPU, cómo se leen y almacenan éstos en memoria, etc. Como resultado, el juego de instrucciones de un procesador puede tener una gran influencia en la idoneidad de un procesador para determinadas aplicaciones.

Las operaciones a realizar en tratamiento digital de señal son poco variadas, limitándose en la mayoría de los casos a multiplicaciones y sumas. Los DSPs disponen de un juego de instrucciones optimizado para este tipo de aplicaciones, con un número de instrucciones reducido, implementándose frecuentemente como microprocesadores con arquitectura RISC. Los beneficios de estas instrucciones especiales son dobles: por una parte permiten un código más compacto que requiere menos espacio en memoria; por otra, incrementa la velocidad de ejecución de algoritmos específicos de tratamiento digital de señal.

Estas instrucciones están optimizadas para las aplicaciones mencionadas, haciendo uso del paralelismo interno de la CPU. Así una misma instrucción puede realizar simultáneamente el producto de dos números almacenados en las posiciones de memoria especificadas por sendos registros de direcciones, acumular el resultado con el contenido de otro registro de datos e incrementar los registros de direcciones.

Los algoritmos de DSP más frecuentes (convolución, correlación, multiplicación de matrices, etc ) se realizan mediante la ejecución repetitiva de una misma instrucción o conjunto de instrucciones denominadas bucles internos o núcleos del algoritmo.

La penalización en la ejecución que introducen las instrucciones de salto es especialmente importante en núcleos pequeños. Puesto que la mayoría de estos bucles se ejecutan en un número fijo de veces, el procesador debe utilizar un registro para almacenar el índice del bucle. La CPU debe utilizarse para incrementar o decrementar el índice y comprobar si se verifica la condición de repetición del bucle. Si es así se vuelve al comienzo del bucle mediante una instrucción de salto condicional. Todos estos pasos penalizan la ejecución del bucle y utilizan registros innecesariamente.

Para solventar estos problemas los DSPs utilizan bucles hardware (zero overhead looping). Los bucles hardware son estructuras de control especiales que repiten una sola instrucción o conjunto de instrucciones un determinado número de veces. La diferencia fundamental con los bucles software es que en un bucle hardware no se pierde tiempo en incrementar o decrementar un registro, comprobar si se ha llegado al final del bucle o saltar al origen de éste. Son especialmente efectivos

para el caso de bucles de instrucción única, ya que se precisa traer la instrucción de memoria una única vez, liberando los buses para realizar otras operaciones como el acceso a datos o coeficientes.





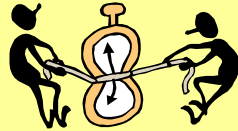
## ARQUITECTURA DE MEMORIA (I).

EJEMPLO DE  
APLICACIÓN.

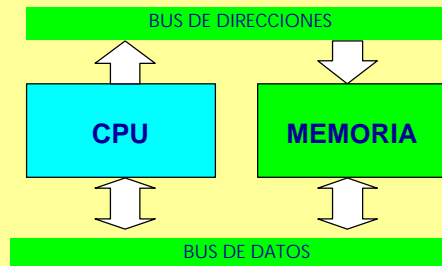
$$y[n] = \sum_{k=0}^{N-1} h[k] \cdot x[n-k]$$

El gran volumen de datos a procesar obliga a que éstos residan en memoria.

- Un acceso para búsqueda de la instrucción.
- Dos accesos para la lectura de los operandos.



### ARQUITECTURA VON NEUMANN.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

16

Las aplicaciones de tratamiento digital de señal procesan un gran volumen de datos. Debido a la limitación del número de registros disponibles en la CPU, normalmente estos datos residen en memoria. Para poder realizar las operaciones MAC en un solo ciclo de instrucción es preciso que los operandos utilizados estén disponibles en el momento de ejecutar dicha instrucción. Normalmente la acumulación de productos se realiza sobre un registro de la CPU, con lo que el almacenamiento del resultado final en memoria no se considera parte del núcleo del algoritmo. Es decir, la ejecución de una instrucción MAC implica la realización de tres accesos a memoria:

- ! Búsqueda del código de la instrucción a ejecutar.
- ! Lectura de los dos operandos del producto.

Teniendo en cuenta estos condicionantes, los DSPs estructuran su memoria de forma que se pueda obtener la instrucción a ejecutar y sus operandos desde memoria al ritmo que los demanda la CPU. De hecho una de las características distintivas de los DSPs es la forma en que organizan la memoria, siendo ésta un factor crítico en la prestaciones del procesador. Dicha arquitectura está orientada a posibilitar la realización simultánea en un solo ciclo de instrucción de los tres accesos anteriormente mencionados.

La arquitectura de memoria utilizada por la mayoría de los microprocesadores convencionales responde al modelo clásico de la arquitectura Von-Neuman. Según este modelo se dispone de un único espacio de memoria en el que se almacena tanto el código a ejecutar como los datos del programa. Los accesos a este espacio de memoria se realizan a través de un único grupo de buses (bus de direcciones y bus de datos). Si bien se trata de una estructura muy simple, el procesador sólo puede realizar un acceso (de lectura o escritura) a memoria durante cada ciclo de instrucción.

En esta situación aunque la CPU del procesador fuese capaz de ejecutar una instrucción MAC en un solo ciclo de instrucción, como es el caso de los DSPs, se requerirían tres accesos secuenciales a memoria para completar su ejecución. Puesto que la duración de cada acceso es de un ciclo instrucción, su ejecución se prolongaría durante tres ciclos.

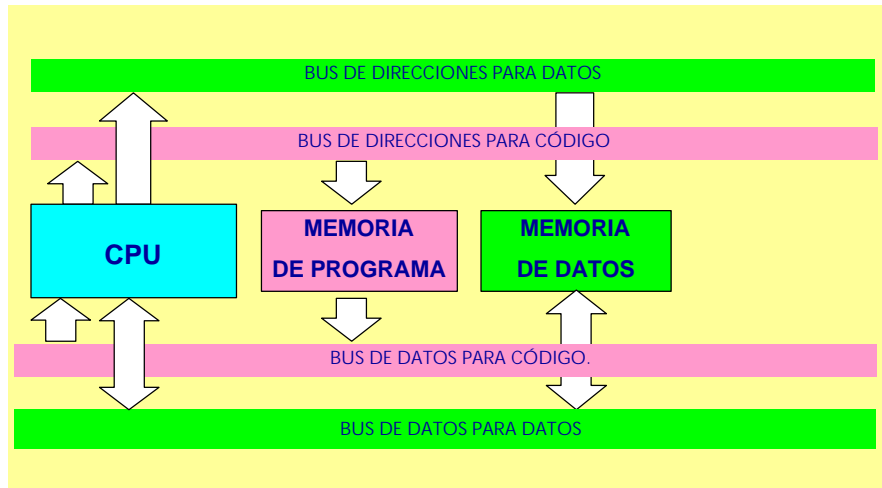
Esta es una de las razones por la que los procesadores convencionales, aun teniendo una elevada potencia de cálculo de pico no son los más indicados para la ejecución de algoritmos de tratamiento digital de señal.

Las mejoras de esta arquitectura van orientadas a aumentar el ancho de banda de los accesos a memoria. Existen distintas alternativas para conseguir este objetivo si bien en el caso de los DSPs más sencillos las limitaciones que introducen dificultan enormemente su programación.



## ARQUITECTURA DE MEMORIA (II).

### ARQUITECTURA HARVARD

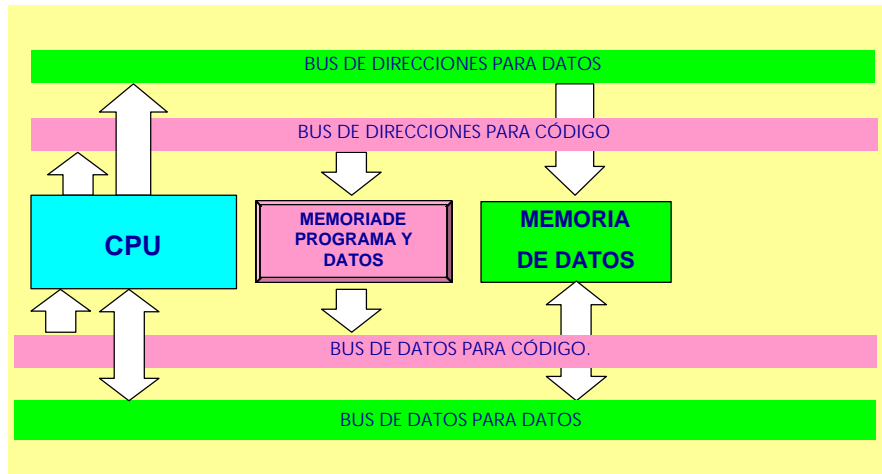


La arquitectura Harvard trata de aumentar el ancho de banda de los accesos aumentando el paralelismo de la memoria. Para ello se dispone de dos espacios de memoria independientes, uno para almacenar el código a ejecutar y el otro para los datos. Cada uno de estos espacios dispone de su propio grupo de buses, con lo cual es posible acceder simultáneamente a ambos espacios. La arquitectura Harvard original restringe el uso a que se destina cada uno de los espacios de memoria (almacenamiento de código o de datos). Esta solución no es del todo adecuada, ya que en instrucciones de tipo MAC es preciso acceder a dos operandos en memoria, manteniéndose por tanto el cuello de botella en los accesos a memoria de datos.



## ARQUITECTURA DE MEMORIA (II).

### ARQUITECTURA HARVARD MODIFICADA.

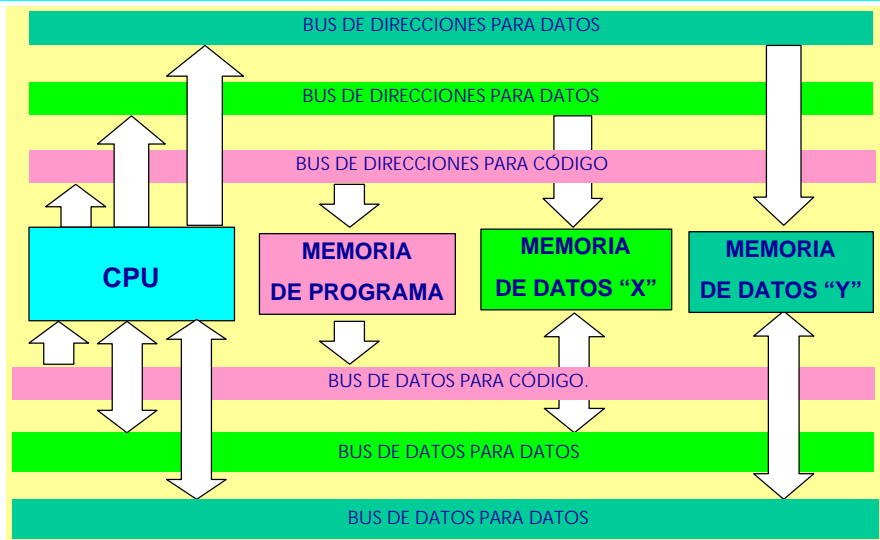


Esta restricción inicial se ha flexibilizado dando lugar a la arquitectura Harvard modificada, en la cual se permite el almacenamiento de código y datos en el espacio de memoria de programa. Puesto que los DSPs suelen disponer módulos de repetición de instrucciones, una vez obtenido el código de la instrucción MAC, el espacio de memoria de programa queda disponible para obtener uno de los operandos. Por tanto, a excepción de la primera vez que se ejecuta dicha instrucción, el resto completan su ejecución en un solo ciclo de instrucción.



## ARQUITECTURA DE MEMORIA (II).

### ARQUITECTURA HARVARD MEJORADA.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

20

Otras aproximaciones eliminan la latencia introducida durante la ejecución de la primera instrucción MAC desarrollando el concepto de la arquitectura Harvard. Para ello dividen a su vez la memoria de datos en dos bancos de memoria, cada uno de los cuales dispone de su propio juego de buses de datos y direcciones. Así se dispone de un banco de memoria de programa y dos bancos de memoria para datos denominados X e Y respectivamente. Estas tres memorias permiten realizar al procesador tres accesos independientes por cada ciclo de instrucción:

- ! Una búsqueda de instrucción.
- ! Un acceso de datos al banco X.
- ! Un acceso de datos al banco Y.

En el caso de los DSPs más recientes las restricciones referentes al uso a que se destina cada uno de los espacios de memoria desaparecen. En su lugar se dispone de un único espacio compuesto por varios de bancos de memoria independientes, cada uno con su propio grupo de buses. Es responsabilidad del programador distribuir cada una de las secciones del programa sobre los distintos bancos para aprovechar al máximo el paralelismo del procesador.

Puesto que el desdoblamiento en múltiples buses de memoria fuera del chip es costoso, los DSPs generalmente proporcionan un único conjunto de buses externo. Los procesadores con múltiples bancos de memoria normalmente proporcionan una pequeña cantidad de memoria interna para cada uno de los bancos. Aunque los bancos de memoria pueden ampliarse externamente, no se pueden realizar accesos externos múltiples en paralelo, debido a la ausencia de un segundo grupo de buses para la memoria externa. Por tanto, si se precisasen múltiples accesos a memoria externa durante la ejecución de una instrucción, la ejecución de esta se prolongaría a lo largo de varios ciclos de instrucción de forma que los accesos a memoria se realizarían de forma secuencial.

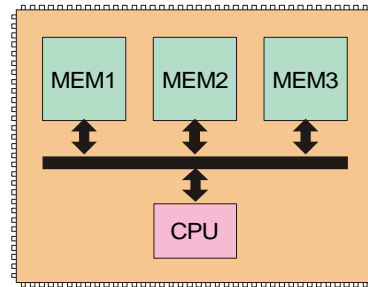
Puesto que el desdoblamiento en múltiples buses de memoria fuera del chip es costoso, los DSPs generalmente proporcionan un único conjunto de buses externo. Los procesadores con múltiples bancos de memoria normalmente proporcionan una pequeña cantidad de memoria interna para cada uno de los bancos. Aunque los bancos de memoria pueden ampliarse externamente, no se pueden realizar accesos externos múltiples en paralelo, debido a la ausencia de un segundo grupo de buses para la memoria externa. Por tanto, si se precisasen múltiples accesos a memoria externa durante la ejecución de una instrucción, la ejecución de esta se prolongaría a lo largo de varios ciclos de instrucción de forma que los accesos a memoria se realizarían de forma secuencial.



## ARQUITECTURA DE MEMORIA (III).

### MEMORIA INTERNA

- Varios bancos de memoria internos.
  - ◆ Accesos multiciclo.
  - ◆ Alta velocidad.
  - ◆ Reducido tamaño.
- Memoria “caché” de programa.



Existen otras alternativas no excluyentes al empleo de la arquitectura Harvard, que permiten aumentar el ancho de banda de la memoria. Una de ellas se basa en el uso de memorias de acceso múltiple. Se trata de memorias lo suficientemente rápidas para permitir varios accesos secuenciales por cada ciclo de instrucción a través de un único grupo de buses, o bien utilizar memorias multipuerto que permitan varios accesos concurrentes a memoria sobre dos o más grupos de buses independientes.

Tanto las memorias rápidas como las multipuerto se integran en el propio encapsulado del DSP. En el caso de las memorias rápidas situarlas en el exterior implicaría la introducción de retardos adicionales que podrían imposibilitar la realización de varios accesos en un solo ciclo; mientras que para las memorias multipuerto, su situación en el exterior significaría elevar el número de terminales de E/S del DSP, lo que incrementaría el coste y tamaño. Independientemente del tipo de implementación la capacidad de los bancos de memoria de acceso múltiple suele ser reducida debido a la limitación del área disponible en el chip.

Para completar en un solo ciclo la ejecución de la instrucción MAC ( incluso el almacenamiento en memoria del resultado de la acumulación) es preciso distribuir adecuadamente las distintas partes del programa (código, operandos y resultados) de forma que no se sobrepase el ancho de banda de cada banco. En caso contrario, se ralentizaría la ejecución del programa.

Otra alternativa utilizada para aumentar el ancho de banda de la memoria consiste en el empleo de memorias caché de programa, que reducen la necesidad de acceder a memoria en la fase de búsqueda de instrucciones. Eliminar este acceso permite un posible acceso a memoria que se empleará para leer o escribir un dato, o bien puede eliminar los retardos asociados con una memoria de programa externa mas lenta.



## ARQUITECTURA DE MEMORIA (IV).

### INTERFACE DE MEMORIA EXTERNA

#### CARACTERIZADO POR:

- Cantidad.
  - ◆ Duplicidad de buses en DSPs de gama alta.
  - ◆ Posibilidad de accesos simultáneos.
- Versatilidad.
  - Generación automática de estados de espera.
  - Facilidad de conexión física.
  - Sistemas multiprocesador.
- Temporización.

Respecto al interface externo con memoria de un DSP, éste puede caracterizarse, básicamente, mediante tres propiedades: el número de puertos disponibles, la sofisticación y flexibilidad de los mismos y los requerimientos temporales.

Incluso aunque cuenten con varios bancos de memoria internos independientes, la mayor parte de procesadores DSP disponen de un único conjunto de buses externo. Es así porque extender los buses al exterior implica encapsulados con un número de terminales demasiado elevado, lo que incrementaría notablemente el precio final. Esto supone la imposibilidad de realizar varios accesos a posiciones externas en un mismo ciclo de instrucción. La duplicidad de buses externos sólo aparece en dispositivos de gama alta.

Los interfaces con memoria externa de los DSPs varían bastante en cuanto a sofisticación y flexibilidad. Algunos son relativamente simples, con tan solo unos pocos terminales de control. Otros son mucho más complejos, permitiendo conectar un mayor rango de dispositivos de memoria externa y buses, sin necesidad de hardware adicional. Algunas características que distinguen estos interfaces son la flexibilidad y granularidad de los estados de espera programables, la disponibilidad de facilidades para implementar entornos multiprocesador (terminales para petición y cesión de bus, selección de acceso a memoria compartida, accesos con interbloqueo para implementar semáforos, etc) y soporte para memorias DRAM en modo página o SRAM síncronas.

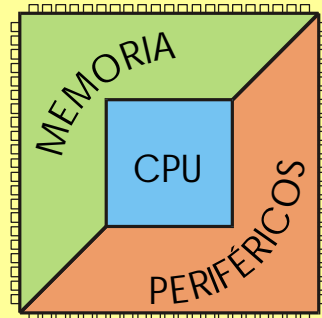
Los accesos a memoria externa no suelen permitir accesos múltiples en un solo ciclo de instrucción. Es más, la temporización varía dependiendo del sentido de la transferencia. Si bien los accesos en modo lectura se pueden completar en un solo ciclo, las escrituras suelen prolongarse durante dos o más ciclos. Por este motivo la memoria interna se suele reservar para almacenar las partes del programa que se pueden modificar durante la ejecución de este (pila y variables) mientras que el código se almacena en memoria externa.





## PERIFÉRICOS INTEGRADOS E INTERFACES DE I/O.

INTEGRACIÓN DE PERIFÉRICOS DE GRAN VERSATILIDAD PARA COMUNICARSE CON EL MUNDO EXTERIOR.



- TIMERS.
- PUERTOS SERIE.
- CONTROLADORES DMA.
- INTERFACE CON UN HOST.
- PUERTOS DE COMUNICACIÓN.
- *CONVERSORES A/D Y D/A.*
- *GENERADORES PWM.*

Dispositivos concebidos como sistemas autónomos



Idóneo para aplicaciones portátiles.

Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

24

La mayor parte de los DSPs integran en el propio encapsulado periféricos de gran versatilidad que le permiten comunicarse con el mundo exterior sin intervención del procesador. Este es un aspecto de gran importancia y que los distingue de los procesadores de propósito general. De entre estos los más frecuentes suelen ser:

**! Temporizadores.** Normalmente, todos los procesadores incluyen estos dispositivos, que suelen admitir como entrada un reloj interno o externo. Se suelen utilizar para la generación de interrupciones periódicas que sincronizan el proceso de muestreo.

**! Puertos serie.** Las características y complejidad de estos interfaces varían de un procesador a otro, pero es habitual su presencia en todos ellos. Suelen utilizarse para comunicarse con circuitos que incluyen conversores A/D y D/A denominados "codecs". Esto permite alejar la parte analógica del procesador eliminando el acoplo de ruido en la primera. De hecho la mayoría de los fabricantes de DSPs diseñan dispositivos de este tipo con un interface que permite la conexión directa con el puerto serie. No es raro encontrar también DSPs que permiten la transmisión del código de arranque en sistemas sin memoria ROM.

**! Controlador DMA.** Este tipo de periférico permite efectuar transferencias de forma rápida sin intervención del procesador. Esto es especialmente útil en aplicaciones en las que el volumen de datos a manejar es considerable como por ejemplo tratamiento digital de imagen. En tal caso el DMA se utiliza para llevar bloques de datos desde una memoria externa de gran capacidad a memoria interna con objeto de que puedan procesar a mayor velocidad.

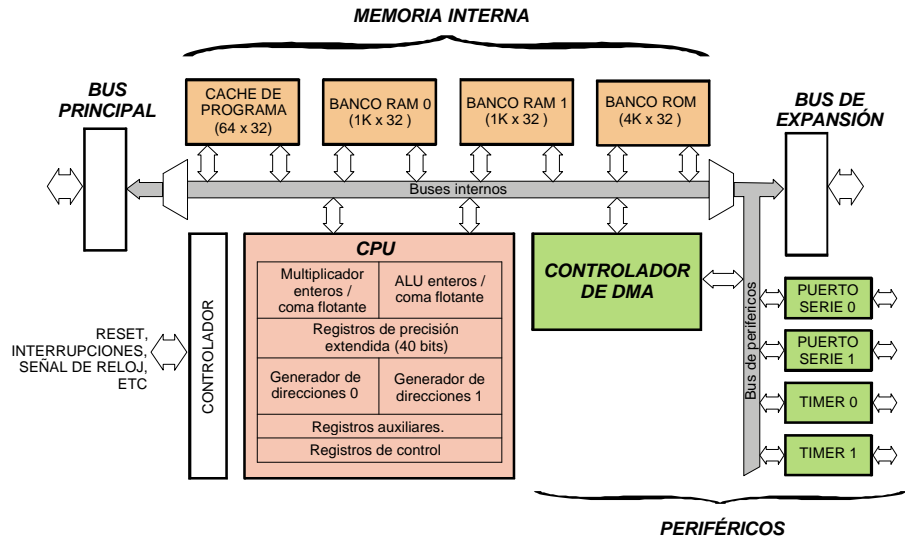
**! Interface con un host.** En algunas casos los DSPs actúan como coprocesadores matemáticos, formando parte de un sistema global controlado por un procesador de propósito general. En algunas casos los DSPs actúan como coprocesadores matemáticos, formando parte de un sistema global controlado por un procesador de propósito general. Para comunicarse con este procesador host algunos DSPs incorporan un puerto paralelo bidireccional de 8 o 16 bits, que podría incluso estar diseñado específicamente para comunicarse con un bus standard, como por ejemplo ISA o PCI. Las comunicaciones a través de este puerto suelen realizarse mediante DMA de manera que posible transferir datos entre memoria y el puerto sin intervención del procesador. El control del puerto se realiza mediante instrucciones específicas. En ocasiones este interface permite incluso el control del funcionamiento del DSP (forzar la ejecución de rutinas de tratamiento de interrupción, acceder a los registros internos del DSP o a su memoria, o incluso realizar la carga del código a ejecutar por el procesador -bootstrapping).

**! Puertos de comunicación.** Se trata de puertos paralelo diseñados para comunicación entre DSPs del mismo tipo que se conectan en red para implementar un sistema multiprocesador. Puesto que la anchura de estos puertos (8 bits) es menor que el tamaño de la palabra de datos de los DSPs que los incluyen (32 bits), los puertos disponen de FIFOs para la fragmentación y reensamblado de los datos que se transmiten a través de ellos. La comunicación a través de estos puertos suele estar asistida por DMA.

Los DSPs diseñados para aplicaciones muy concretas como el control de motores o sistemas de potencia disponen, además, de periféricos específicos para tales aplicaciones como generadores de señal PWM, temporizadores especiales para la implementación de tacómetros digitales, conversores A/D, etc. Estos últimos se suelen incluir conjuntamente con conversores D/A, en DSPs destinados a aplicaciones de tratamiento de voz (teléfonos móviles, contestadores automático). En general suele tratarse de DSPs de gama baja ya que el área de chip utilizada para estos periféricos limita el tamaño de los bancos de memoria interna y la sofisticación de la CPU.



## EJEMPLO: TMS320C3x.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

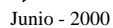
26

A continuación se describe brevemente como se quedan puestas de manifiesto estas características en el DSP que se estudiará con mayor detalle en la siguiente lección.

Se trata de un dispositivo que integra una gran variedad de periféricos: controlador de DMA, puertos serie y timers.

En este diagrama de bloques se intuye la existencia de una arquitectura Harvard. Así aparecen múltiples bancos de memoria interna. Además esta arquitectura se manifiesta también en el exterior del dispositivo con dos buses externos: bus principal y bus de expansión.

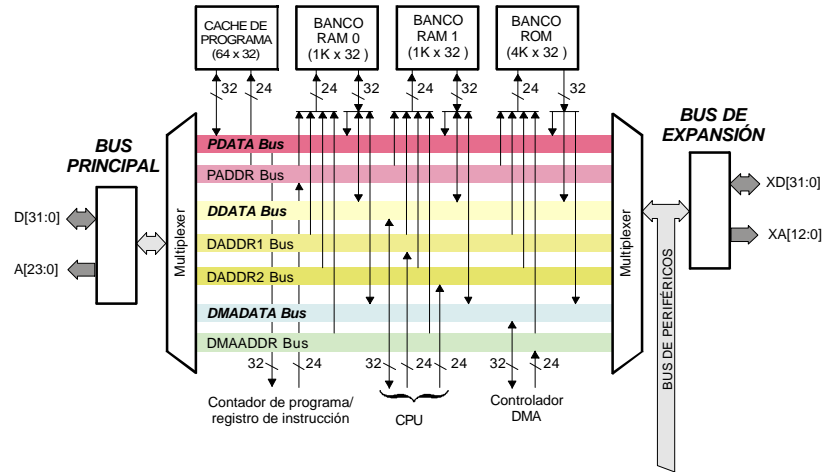
Observaremos a continuación con mayor detalle, la arquitectura de la CPU y los buses internos.



La unidad DAG esta compuesta por dos ALUs específicas (ARAU0 y ARA1) y un conjunto de registros para direccionamiento indirecto (registros auxiliares). Al utilizar direccionamiento circular, el tamaño del buffer se fija mediante el registro BK.



## ARQUITECTURA DE MEMORIA DEL TMS320C3x.



La arquitectura Harvard se aprecia claramente en esta figura. Así existen grupos de buses para:

- Programa (PDATA y PADDR)
- Datos (DDATA, DADDR1 y DADDR2).

En este dispositivo el controlador de DMA tiene un tratamiento diferenciado por lo que se le ha dotado de su propio grupo de buses (DMADATA y DMAADDR)



## ÍNDICE

---



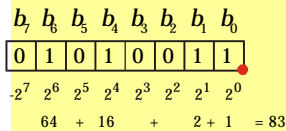
- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
  - Formatos de coma fija y coma flotante.
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**



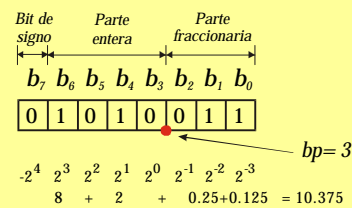
## FORMATOS DE DATOS Y ANCHO DE PALABRA DE DATOS (I).

### NÚMEROS EN COMA FIJA.

#### ENTERO:



#### COMA FIJA:



- Existencia de un punto binario que introduce un factor de escala de  $2^{-bp}$ .

◆ Precisión: Peso del LSB =  $2^{-bp}$ .

- Precisión vs margen de valores a representar.

- Programación laboriosa.
- Emulación software de aritmética en coma flotante no eficiente.

Una de las características más importantes que determinan la idoneidad de un DSP para una aplicación dada es el tipo de formato y número de bits de los datos con que realiza los cálculos matemáticos. Con relación al tipo de formato de datos, los DSPs pueden operar con números en coma fija o en coma flotante.

Algunos DSPs sólo son capaces de operar con números enteros, mientras que las aplicaciones es común manejar números fraccionarios como por ejemplo los coeficientes de un filtro. El formato de coma fija utiliza una representación similar a la de un número entero, salvo que se considera la existencia de un punto binario mediante el cual se escalan los valores enteros para de esta forma, obtener números fraccionarios. Este factor de escala es igual a  $2^{-bp}$  donde "bp" es la posición del punto binario. El DSP realiza las operaciones de suma o multiplicación como si se tratase de números enteros, sin considerar este factor de escala. Es responsabilidad del programador interpretar la posición del punto binario.

Al desplazar el punto decimal a la izquierda, utilizando más bits para la parte fraccionaria, la precisión aumenta, pero disminuye el margen de valores de la representación. Puesto que el tamaño de la palabra de datos es fijo, la situación del punto binario será una situación de compromiso entre la precisión a obtener y el margen de valores a cubrir. El programador debe utilizar el mayor número de bits para la parte fraccionaria (máxima precisión) que permiten representar todo el rango de valores que toma una variable.

Si durante el procesamiento un número en coma fija aumenta demasiado para poder ser representado con el número de bits disponibles para la parte entera, el programador debe realizar un escalado descendente del número mediante un desplazamiento a derechas perdiendo los bits de menor peso y por tanto disminuyendo la precisión. Si por el contrario el número en coma fija disminuye demasiado el número de bits utilizados en la parte fraccionaria puede ser insuficiente. El programador debe realizar un desplazamiento a izquierdas para aumentar la precisión.

En ambos casos el programador debe tomar en consideración como se ha ido desplazando el punto binario restaurando todos los números de coma fraccionaria a una misma escala en una etapa posterior. Esto convierte la programación de aplicaciones en una tarea muy tediosa.

La utilización de rutinas en coma fija que emulan las operaciones en coma flotante (que permiten manejar cómodamente números fraccionarios) es muy costosa en cuanto tiempo de ejecución del código, lo que hace imposible su uso en aplicaciones de tiempo real.





### NÚMEROS EN COMA FLOTANTE.

*mantisa*

0	1	1	0	1	0	0	0	0
---	---	---	---	---	---	---	---	---

$-2^1 \quad 2^0 \quad 2^{-1} \quad 2^{-2} \quad 2^{-3} \quad 2^{-4} \quad 2^{-5} \quad 2^{-6} \quad 2^{-7}$

*exponente*

0	1	1	0
---	---	---	---

$2^{-3} \quad 2^{-2} \quad 2^{-1} \quad 2^0$

$$\text{Valor} = \text{mantisa} \cdot 2^{\text{exponente}}$$

- Un número en coma flotante se compone de:
  - ♦ Mantisa: Número fraccionario en coma fija.
  - ♦ Exponente: Determina la posición de bp.
- Mecanismo más flexible que la coma fija.
  - ♦ Mayor margen de valores.
  - ♦ Mayor precisión.
- El propio hardware realiza el escalado.
- Programación sencilla.

Otros procesadores disponen de una CPU capaz de operar directamente con números de coma flotante. La aritmética en coma flotante es un mecanismo más flexible que la aritmética en coma fija. Con la aritmética en coma flotante, los diseñadores de sistemas tienen acceso a un rango de valores mucho más amplio y a una mejor precisión. Esto facilita la programación ya que no es necesario preocuparse de la realización del escalado. Un número de coma flotante se representa mediante una mantisa y un exponente siendo su valor:

$$\text{valor} = \text{mantisa} \cdot 2^{\text{exponente}}$$

La mantisa es un número fraccionario mientras que el exponente determina la posición del punto binario. En estos procesadores es el propio hardware de la CPU el que realiza los escalados mencionados anteriormente, quedando reflejada la posición del punto binario en el exponente. Esto facilita enormemente la programación de las aplicaciones.



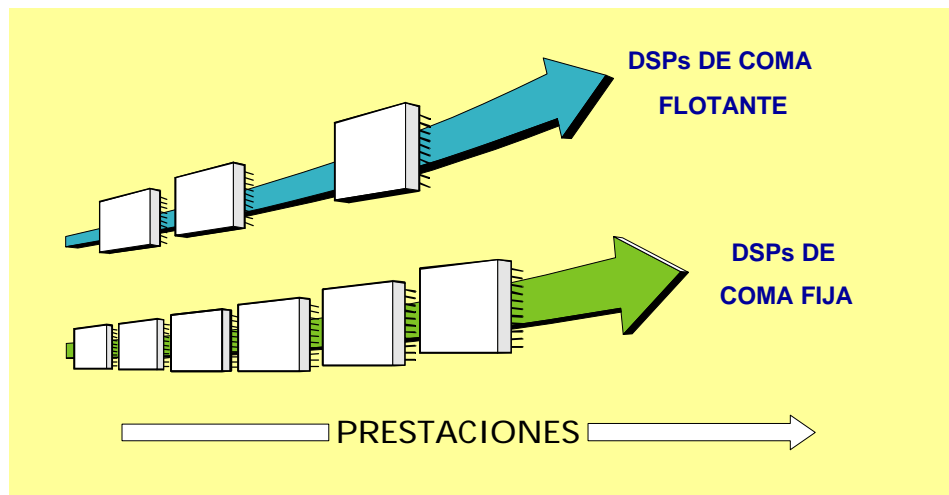
## ÍNDICE

- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
  - Según el tipo de aritmética utilizada.
  - Según el paralelismo del dispositivo.
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**



## TIPOS DE DSPs (II).

SEGÚN EL TIPO DE ARITMÉTICA UTILIZADA.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

34

Normalmente la clasificación de los DSPs se realiza en base al tipo de aritmética que utilizan para realizar los cálculos matemáticos dividiéndose en DSPs coma fija y DSPs de coma flotante. Dentro de cada grupo se clasifican, además, según la anchura de su palabra de datos.

La CPU de los procesadores de coma fija requiere un hardware más simple que la de los procesadores de coma flotante. Esto se traduce en una reducción del coste unitario del DSP haciéndolos idóneos para aplicaciones de gran consumo que no requieran unas prestaciones elevadas. Esta simplicidad de la CPU también reduce el consumo del dispositivo y su tamaño, un aspecto sumamente interesante para aplicaciones portátiles como por ejemplo teléfonos móviles. La utilización de una CPU poco sofisticada permite liberar área del chip para incluir bancos de memoria RAM de mayor tamaño o incluso bancos de memoria EPROM o FLASH, donde grabar el código de la aplicación. Además, suelen disponer de un conjunto de periféricos más variado. De hecho los procesadores destinados a aplicaciones específicas como por ejemplo control de motores, sistemas de tratamiento de voz, etc, son procesadores de coma fija.

La anchura de la palabra de datos puede ser según los casos de 16, 24 o 32 bits. Esto tiene una importante repercusión en el coste, porque influye poderosamente en el tamaño del circuito integrado y en el número de terminales del dispositivo, así como el tamaño de los dispositivos de memoria externa conectados al mismo. Por lo tanto, los diseñadores intentan utilizar el chip con el menor ancho de palabra que su aplicación puede tolerar.

Los DSPs de coma flotante son dispositivos de gama alta, cuya CPU dispone de hardware específico para operar con datos de coma flotante. La anchura de la palabra de datos suele ser de 32 bits si bien el uso de bits de guarda les permite operar en el interior de la CPU con datos de 40 bits. Esto no es óbice para que puedan operar también con datos en coma fija. El área ocupada la CPU de este tipo de DSPs es mayor que en el caso de los de coma fija. Por este motivo, la variedad de los periféricos que integran es menor, tratándose en la mayoría de los casos de periféricos (puertos serie y paralelo, DMA) utilizados en la comunicación con elementos externos (convertidores A/D y D/A). Dentro de este grupo de DSPs, los más sofisticados disponen de puertos de comunicación que permiten facilitar el montaje de redes de DSPs para procesamiento en paralelo.

Cada tipo de procesador es ideal para un rango específico de aplicaciones. Los procesadores de 16 bits de coma fija son adecuados para sistemas de voz, como teléfonos, ya que éstos trabajan con el rango relativamente estrecho de las frecuencias del sonido. Las aplicaciones estéreo de alta fidelidad tienen un rango de frecuencias más amplio, de forma general, los requerimientos mínimos para este tipo de aplicaciones serían un ADC de 16 bits y un procesador de 24 bits de coma fija, de esta forma se proporciona un rango suficientemente amplio para obtener la señal de alta fidelidad y para poder manipular los valores que se obtienen al procesar la señal. El procesamiento de imágenes, gráficos en 3D y simulaciones científicas tiene un rango dinámico mucho más amplio, por lo que precisa procesadores DSP de 32 bits con aritmética de coma flotante.

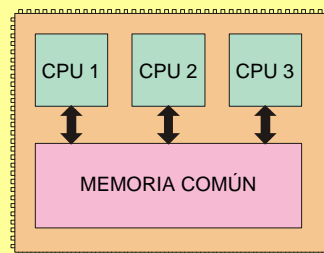


## TIPOS DE DSPs (I).

### SEGÚN EL PARALELISMO DEL DISPOSITIVO.

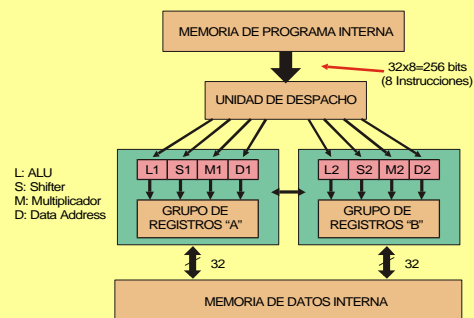
#### • DSPs CON PARALELISMO EXPLÍCITO:

- ♦ VARIAS CPUs EN UN MISMO DISPOSITIVO.
- ♦ DIFICULTAD DE PROGRAMACIÓN.



#### • DSPs CON PARALELISMO IMPLÍCITO:

- ♦ DUPLICIDAD DE UNIDADES FUNCIONALES.
- ♦ EMISIÓN DE MÚLTIPLES INSTRUCCIONES (VLIW).



Otra clasificación de los DSPs se puede realizar atendiendo al paralelismo de éstos entendiendo como tal la posibilidad de ejecutar múltiples instrucciones de forma concurrente, entre los que no se incluyen los DSPs básicos vistos hasta ahora. Este paralelismo puede ser explícito o implícito. En un DSP con paralelismo explícito se integran varias CPUs en un mismo encapsulado que se comunican por medio de una memoria compartida interna. Esta opción se ha convertido en una vía muerta, ya que es responsabilidad del programador el reparto del código a ejecutar por parte de cada uno de las CPUs, y la programación de las rutinas de comunicación, siendo esta tarea muy tediosa.

Un DSP con paralelismo implícito dispone de única CPU con múltiples unidades funcionales (por ejemplo, varias ALUs, multiplicadores y conjuntos de registros) de forma que se pueden ejecutar en paralelo varias instrucciones cada una de ellas sobre una unidad funcional. Para ello disponen de un tamaño de la palabra de instrucción de gran tamaño (Very Long Instruction Word, p.e. 256 bits), en la que se empaquetan varias instrucciones individuales. La planificación de qué instrucciones se van a ejecutar en paralelo en cada momento la realiza el propio compilador siendo un proceso transparente por tanto para el programador. Este tipo de procesadores es sin duda la vanguardia de la tecnología de los DSPs.



## ÍNDICE

---

- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**





## CRITERIOS DE SELECCIÓN (I).

- Tipo de aritmética utilizada y ancho de palabra de datos.
- Velocidad.
  - ◆ Valores de pico:
    - ◆ Coma fija: MIPS.
    - ◆ Coma flotante: MFLOPS.
  - ◆ Test de pruebas (benchmarks).
- Memoria interna.
- Soporte multiprocesador.
- Consumo.
- Coste.
- *Rango dinámico.*

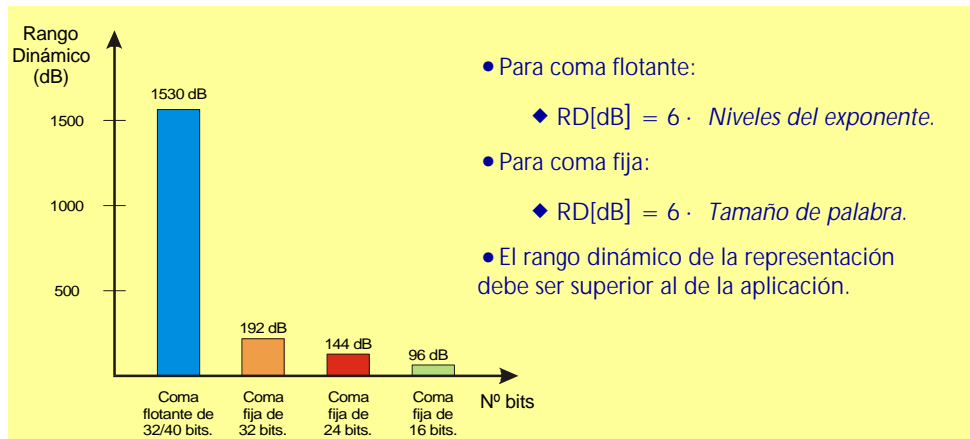
Los factores que se han de considerar a la hora de seleccionar un DSP para una aplicación determinada son:

- **El tipo de aritmética utilizada y el ancho de palabra de datos.** El uso de DSPs de coma flotante simplifica la programación de algoritmos, pero son dispositivos más caros y de mayor consumo. El tamaño de palabra de datos influye en el número de chips de los bancos de memoria externa y por tanto en el coste y consumo del sistema.
- **Velocidad.** Se puede indicar de dos formas. Una de ellas es como valores de pico expresándose en millones de instrucciones por segundo para el caso de los DSPs de coma fija y en millones de operaciones en coma flotante para el caso de los DSPs de coma flotante. Otra forma de indicar la velocidad es mediante bancos de pruebas. Éstos son algoritmos típicos de tratamiento digital de señal que se ejecutan sobre distintos. Midiéndose los tiempos de ejecución en cada dispositivo se puede establecer una comparativa de sus velocidades.
- **Memoria interna.** Es deseable que los DSPs dispongan de la mayor cantidad de memoria interna, ya que los accesos sobre ésta se realizan a mayor velocidad. La disponibilidad de memoria FLASH interna permite reducir la complejidad del sistema.
- **Soporte multiprocesador.** Interesante para aquellas aplicaciones que debido a su elevada carga computacional hace necesario su implementación en sistemas multiprocesador.
- **Consumo.** Este es un factor determinante en aplicaciones portátiles, como por ejemplo teléfonos móviles.
- **Coste.** En aplicaciones de gran consumo este aspecto puede primar sobre otros que inciden más directamente sobre las prestaciones del DSP.
- **Rango dinámico.** Es una figura de mérito que relaciona el tipo aritmética utilizada y el ancho de la palabra de datos.



## CRITERIOS DE SELECCIÓN (II).

**RANGO DINÁMICO.** Relación entre el máximo y el mínimo valor (distinto de cero) a representar.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

39

El rango dinámico es un concepto propio de los procesadores. Se define como la relación existente entre el máximo y mínimo valor (distinto de cero) representables.

En un formato de coma flotante éste viene determinado por el número de niveles del exponente. Si expresamos esta relación en dB:

$$RD[dB] = 6 \cdot \text{Niveles del exponente}$$

Así para el caso típico de un exponente de 8 bits el rango dinámico es de 1530.

En un formato en coma fija este depende del tamaño de palabra de datos según la relación:

$$RD[dB] = 6 \cdot \text{Tamaño de palabra}.$$

El rango dinámico del procesador debe ser superior al de la aplicación. Este último viene determinado por los conversores A/D y D/A utilizados. De esta forma se garantiza que los errores introducidos durante el procesamiento (debidos a truncamientos, redondeos, etc) quedan por debajo del error que introducen los propios conversores.





## ÍNDICE

---

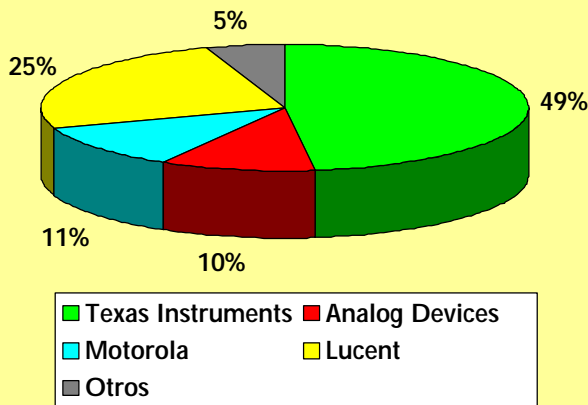
- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**





## DISPOSITIVOS COMERCIALES MÁS REPRESENTATIVOS.

**Mercado mundial de DSPs en 1999: \$4.39 billones.**  
(Fuente: Semiconductor Business News)



- **TEXAS INSTRUMENTS:**

- ◆ TMS320C2x/5x.
- ◆ TMS320C3x/4x.
- ◆ TMS320C6000.

- **ANALOG DEVICES:**

- ◆ ADSP-21xx.
- ◆ Sharc / Tiger Sharc.

- **MOTOROLA:**

- ◆ DSP56xxx.
- ◆ StarCore SC100.

- **LUCENT:**

- ◆ DSP16xx.
- ◆ DSP16000

Los principales fabricantes de DSPs son, por este orden, Texas Instruments, Lucent, Motorola y Analog Devices.

A continuación se enumeran las familias de DSPs más representativas. Cada fabricante suele disponer de familias de DSPs de coma fija y familias de coma flotante. Algunos también disponen de DSPs de emisión múltiple de instrucciones.



## ÍNDICE

---

- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**





## ÁREAS DE APLICACIÓN.

- COMUNICACIONES:
  - ◆ Telefonía móvil digital.
  - ◆ Modems.
  - ◆ LANs.
- ELECTRÓNICA DE CONSUMO:
  - ◆ Cámaras digitales.
  - ◆ Audio/video digital.
  - ◆ Radio y Televisión digital.
- INDUSTRIA:
  - ◆ Control de motores.
- MILITAR/AEROESPACIAL:
  - ◆ Radar y sonar.
  - ◆ Guiado de misiles.
- INSTRUMENTACIÓN:
  - ◆ GPS.
  - ◆ Procesamiento imágenes médicas .

Las ventajas de los DSPs han hecho que cada vez sean mayores las áreas de aplicación. Entre éstas cabe destacar:

! Comunicaciones. Se utiliza en equipos de telefonía móvil, modems, redes de área local, etc.

! Electrónica de consumo. Se utilizan en cámaras digitales para tratamiento de imágenes; en equipos de audio y video para simular efectos sonoros de recintos, técnicas "surround"; en radiodifusión digital tanto de audio como de televisión para implementar los algoritmos de compresión y descompresión.

! Industrial. Cada vez es mayor la utilización de técnicas digitales en sistemas de control de motores eléctricos y equipos de potencia que permiten obtener: menores costes, mejores prestaciones y un menor consumo de energía.

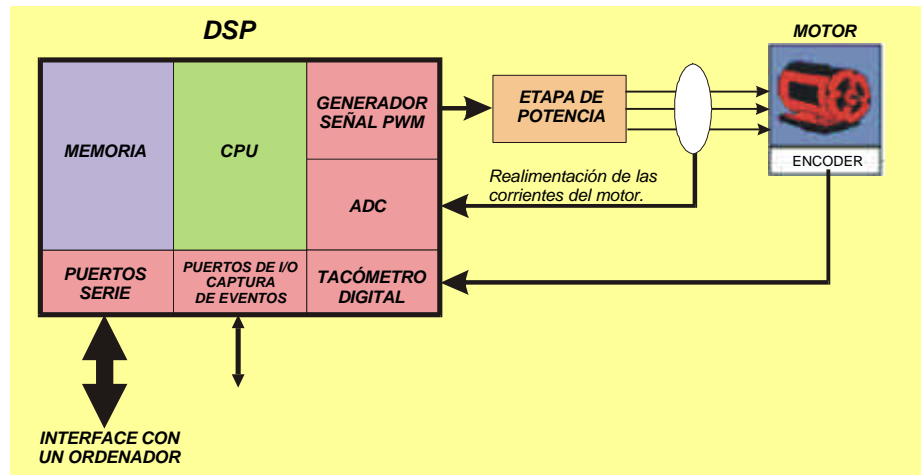
! Militar/aeroespacial. Se usan estas técnicas para tratamiento de señales de radar, sonar, guiado de misiles, etc.

! Instrumentación. Como ejemplo de aplicaciones de este tipo se encuentran los sistemas de posicionamiento (GPS), en los cuales se realiza una correlación entre la señal recibida con un código para obtener los datos de telemetría. Dentro de la instrumentación biomédica se utiliza en resonancia magnética nuclear y ecografía, para realizar el tratamiento de las imágenes obtenidas.



## EJEMPLO DE APLICACIÓN.

### SISTEMA DE CONTROL DE UN MOTOR.



Junio - 2000

"INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE SEÑAL."

44

Como ejemplo de aplicación de los DSPs se muestra la implementación de un sistema de control de la velocidad de giro de un motor. Puesto que los algoritmos utilizados pueden alcanzar gran complejidad se requiere una elevada potencia de cálculo. Además el DSP integra todos los dispositivos necesarios para implementar el sistema.

Las señales de control del motor se obtienen mediante un generador de señal PWM. Estas señales atacan a una etapa de potencia. El lazo de control se puede cerrar de varias formas. Una de ellas sería la obtención de la posición del eje mediante un tacómetro digital recibe las señales en cuadratura provenientes de un encoder solidario al eje del motor. Otra forma de cerrar el lazo sería a partir de las corrientes que fluyen por el devanado del motor, obteniendo el valor de éstas mediante un conversor A/D.

La consigna de velocidad de giro del motor se puede fijar desde un ordenador personal, comunicandose con el DSP a través de un puerto serie. Otro método podría ser mediante un teclado que se conecta a los puertos de entrada/salida del DSP.



## ÍNDICE

---

- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de selección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**





## CONCLUSIONES

- Los algoritmos de tratamiento digital de señal definen las características de los elementos donde se van a implementar.
- Los DSPs son procesadores específicamente diseñados para tales aplicaciones.
- Este diseño específico determina:
  - ◆ La arquitectura de su CPU y la memoria.
  - ◆ Su juego de instrucciones .
  - ◆ La integración de periféricos que le permitan interactuar con el exterior.
- El formato de representación utilizado y la anchura de la palabra de datos determina las aplicaciones en que se puede utilizar un DSP.
- Los DSPs se clasifican atendiendo al tipo de formato de datos para el que han sido diseñados.



## ÍNDICE

---

- ◆ **Introducción.**
- ◆ **Características generales de los Procesadores Digitales de Señal.**
- ◆ **Formatos de datos y ancho de palabra de datos.**
- ◆ **Tipos de Procesadores Digitales de Señal.**
- ◆ **Criterios de elección del DSP.**
- ◆ **Familias de DSPs más representativas.**
- ◆ **Aplicaciones.**
- ◆ **Conclusiones.**
- ◆ **Bibliografía.**







## REFERENCIAS BIBLIOGRÁFICAS

### BIBLIOGRAFÍA BÁSICA.

- "Apuntes de sistemas microprogramados para tratamiento digital de señal" R. Mateos  
"DSP processor fundamentals". P. Lapsley.

### BIBLIOGRAFÍA COMPLEMENTARIA.

- "A simple approach to Digital Signal Processing" M. Ewers .  
"Digital signal processing in VLSI". R.J. Higgings.  
"The Scientist and Engineer's Guide to Digital Signal Processing" S. W. Smith  
"Digital Signal Processing with C and the TMS320C3x" R. Chassing.

### DIRECCIONES DE INTERÉS EN LA WORLD WIDE WEB

<http://www.ti.com/>  
<http://www.analog.com/>  
<http://www.bores.com/courses>  
<http://www.techonline.com/>  
<http://www.ednmag.com/>