

SISTEMA PARA PROCESAMIENTO DE SEÑALES EN TIEMPO REAL

Azpiroz Leehan J.      Linding Boes M.

Universidad Autónoma Metropolitana-Iztapalapa  
Departamento de Ingeniería Eléctrica  
Área de Ingeniería Biomédica

RESUMEN

Se describe a un coprocesador capaz de convertir a una computadora personal tipo IBM en un sistema de procesamiento de señales biomédicos en tiempo real. Se explican su filosofía de diseño y los componentes esenciales de este sistema. Se mencionan las aplicaciones del mismo.

I.- INTRODUCCION

Los avances en las aplicaciones en el procesamiento digital de señales están ligados íntimamente con los avances en los sistemas de computación donde se aplican estas técnicas. Los algoritmos eficientes para el cálculo de la transformada discreta de Fourier (DFT) fueron aplicados desde antes del desarrollo de la computación digital, pero no fue sino hasta 1965 cuando Cooley y Tukey publicaron su algoritmo para el cálculo de la DFT cuando se comenzaron a desarrollar aplicaciones de la DFT para el procesamiento digital de señales. A partir de este algoritmo surgieron muchos otros similares que colectivamente se denominan la transformada de Fourier (FFT) (1).

En la actualidad el desarrollo de aplicaciones de la FFT dentro del campo de las señales biomédicas tiene la limitación de tiempo de ejecución. Esto se debe principalmente a que los sistemas de compu

tación actuales están orientados a la solución de problemas numéricos que utilizan aritmética de punto flotante en 32 y 64 bits de precisión mientras que en el caso de las señales biomédicas es suficiente utilizar aritmética de precisión fija de 16 bits, ya que la relación señal/ruido en estos sistemas rara vez es mayor a 40 dB y esto contribuye al hecho de que sea inútil efectuar cálculos con una precisión mayor a los 16 bits. (2,3)

El cálculo de la FFT para 512 puntos en una minicomputadora del tipo PDP 11/03 toma aproximadamente 45 segundos. Si queremos hacer el mismo cálculo para 1024 puntos, el tiempo aumenta a más del doble. (4)

La gran mayoría de las aplicaciones del procesamiento digital de señales al ambiente médico y biológico tratan de solucionar el problema del tiempo de procesamiento por medio de rutinas que no siempre son las mejores para señales en el rango biológico. Un ejemplo de esto es el análisis espectral de señales electroencefalográficas por medio de las transformadas de Walsh. Este tipo de procesamiento disminuye, en un factor superior a 2, el tiempo que toma una FFT, de tal manera que permite efectuar un análisis de este tipo en tiempo real, pero su poder discriminatorio es relativamente bajo. (5,6)

Otras aplicaciones como el procesamiento de señales de impedancia cerebral para la detección de hemorragias intraventriculares en prematuros utilizan filtros del tipo de respuesta a impulso infinita para poder tener una señal en tiempo real, a costa del error introducido por la no linealidad de la fase de estos filtros. (7)

En otros casos más, como en el análisis espectral de sonidos respiratorios, se pueden obtener datos muy precisos sobre varias enfermedades pulmonares, pero el procesamiento es lento (del orden de algunos minutos).

(8)

Como se puede ver, en los sistemas de computación de propósito general, es prácticamente imposible efectuar un procesamiento digital de señales biomédicas en tiempo real, especialmente si se quieren calcular los espectros de las señales o si se quieren implementar filtros digitales de orden mayor a 20.

(9)

El análisis espectral es un área especial de las aplicaciones del procesamiento en tiempo real para la cual son adecuadas las computadoras personales. Se han desarrollado varios programas para interconectar a computadoras personales a instrumentos independientes tales como analizadores de espectros, para llevar a cabo un análisis más complejo de los datos generados por el instrumento, pero hay muy pocos programas para permitir que una computadora de este tipo se pueda utilizar como reemplazo de instrumentos tales como el analizador anterior mencionado. Si se utiliza la "ferretería" adecuada, muchas computadoras personales podrían adquirir datos y llevar a cabo análisis espectrales. Algunas de las ventajas que un sistema de este tipo proporcionaría son la habilidad de utilizar una unidad para realizar distintas funciones, la facilidad de transporte, su flexibilidad y programabilidad, además de su costo relativamente bajo.

(10)

Un sistema que sea capaz de efectuar estos análisis en tiempo real permitirá obtener información hasta ahora no disponible prácticamente al mismo tiempo en que se detecta la señal, aportando así -- nuevos conocimientos dentro del area de análisis de señales biomédicas.

## II.- DESCRIPCION DEL SISTEMA

Se ha hecho el diseño de un coprocesador compatible con una computadora personal tipo IBM, capaz de llevar a cabo el análisis espectral, convolución y filtrado de señales biomédicas en tiempo real. Como al hacer la transformada rápida de Fourier (FFT) la mayor parte del tiempo de procesamiento se debe a las multiplicaciones que se deben realizar para cada iteración de la "mariposa", la filosofía de este procesador está encaminada a resolver estas operaciones de manera eficiente, utilizando un multiplicador en paralelo, el cual es capaz de efectuar una multiplicación de 16 bits en 200 nSeg. (MPY 016H). Los demás elementos de este coprocesador son -- una unidad aritmética y lógica (ALU) basada en las ALUs de 4 bits del tipo 74181 y varios registros asociados que se acoplan a una unidad de control basada en una PROM. La utilización de multiplicadores en paralelo para el procesador de señales biomédicas en tiempo real se ha implementado para el análisis espectral de señales de velocidad de flujo sanguíneo utilizando ultrasonido Doppler, pero solo para hacer filtrado (utilizando un analizador de espectros comerciales para efectuar otras funciones).

La arquitectura de este coprocesador está íntimamente ligada a las operaciones que tiene que realizar. Como la operación más compleja entre las utilizadas en el procesamiento de señales es la "mariposa" que es parte del algoritmo de la FFT, ésta operación determina en gran medida las características del sistema. Otros factores que afectan a la arquitectura del sistema son la velocidad de transferencia de datos y la cantidad de componentes utilizados (se quiere tener todo el sistema en una tarjeta de expansión de la IBM PC).

La aritmética que se utilizará es de precisión fija de 16 bits y en complemento a 2. Para almacenar los datos es necesario contar con una memoria aparte, ya que la transferencia de información a través del CPU es muy lenta y los canales de DMA están comprometidos. (11)

La memoria estará en un arreglo lineal orientada por palabras tomando en cuenta que se tendrán 1K de datos complejos.

### III.- CARACTERISTICAS Y DISEÑO DEL SISTEMA

Este coprocesador utiliza un canal interno de 16 bits que se comunica con el 8088. Se han definido 7 instrucciones que el coprocesador llevará a cabo.

Estas son:

No operación

Multiplica y acumula

Salida de datos por 2 registros

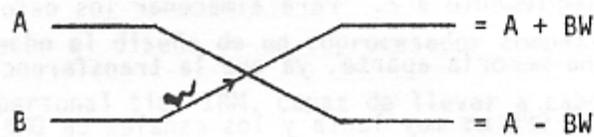
Multiplicación compleja

Mariposa

Magnitud cuadrada

Al ir definiendo estas instrucciones, se observó que era necesario incluir algunos registros de almacenamiento temporal y algunos -- "buffers" de 3 estados para poder ejecutar estas con simplicidad. Como la operación mas compleja es la "mariposa" que se emplea en el cálculo de la FFT, se tomó como base para definir los registros adicionales necesarios (ver figura).

La mariposa se define de la siguiente manera:



A, B y W son números complejos. Se utiliza la multiplicación compleja que se puede ver como:

$$B * W = b_r * w_r - b_j * w_j + j( b_r * w_j + b_j * w_r )$$

donde  $b_r$  es la parte real de B y  $w_r$  es la parte real de W.

La secuencia de microinstrucciones es la siguiente:

Obtención de los datos

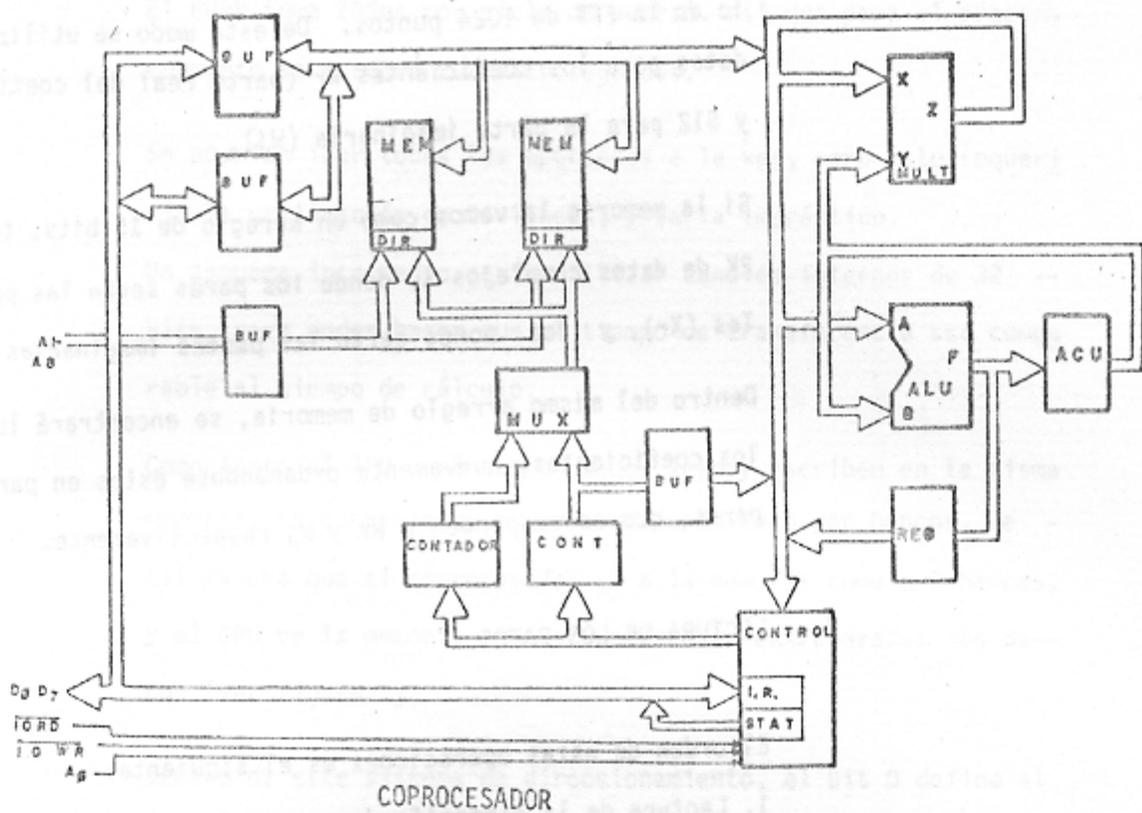
Reacomodo de los mismos

Ejecución de las operaciones

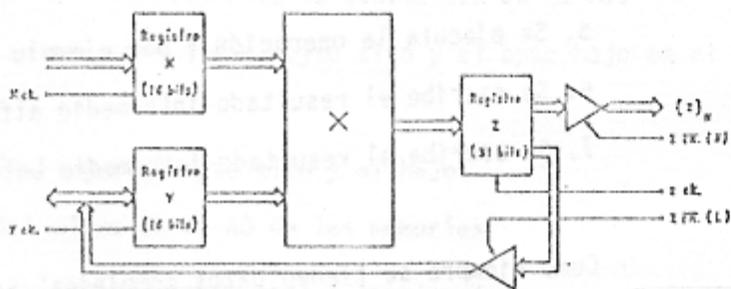
A. OBTENCION DE LOS DATOS.

#### 1.- REQUERIMIENTOS DE MEMORIA PARA EL COPROCESADOR

En la memoria de datos se requieren 1K por 16 bits para cada parte de los datos, es decir, 1K para datos reales y 1K para datos imaginarios. Para la parte de coeficientes, es necesario utilizar unicamente la mitad de este número para el cálculo



Multiplicador MPY 16



lo de la FFT de 1024 puntos. De este modo se utilizarán 512 datos para los coeficientes  $W_r$  (parte real del coeficiente  $W$ ) y 512 para la parte imaginaria ( $W_j$ ).

Si la memoria la vemos como un arreglo de 16 bits, tendremos 2K de datos complejos  $X$ , donde los pares serán las partes reales ( $X_r$ ), y los nones serán las partes imaginarias ( $X_j$ ).

Dentro del mismo arreglo de memoria, se encontrará luego a -- los coeficientes, nuevamente ordenándose estos en pares e impares, que corresponden a  $W_r$  y  $W_j$  respectivamente.

## 2.- LECTURA DE LOS DATOS

El orden de estas operaciones es el siguiente:

1. Lectura de la dirección de un operando.
2. Lectura del operando A (que en realidad son 2 lecturas: partes real e imaginaria), además de calcular la dirección del operando B.
3. Leer operando B y calcular la dirección del coeficiente  $W$ .
4. Se reacomodan los datos
5. Se ejecuta la operación ( por ejemplo la mariposa).
6. Se escribe el resultado intermedio alfa.
7. Se escribe el resultado intermedio beta.

Como siempre se tienen datos complejos, son 2 lecturas o escrituras por transferencia, así que para todo este proceso se toman 11 ciclos de acceso por cada valor.

El 8088 toma 200ns por ciclo de acceso, así que para el proceso completo se tardará 2.2 microsegundos.

Se podrían leer todos los operandos a la vez, pero esto requeriría la utilización de 96 líneas, y sería impráctico.

Un esquema intermedio sería utilizar canales internos de 32 -- bits, para poder hacer que el tiempo de transferencia sea comparable al tiempo de cálculo.

Como tanto el CPU y el coprocesador leen y escriben en la misma memoria, es conveniente hacer direccionamiento por bancos, de tal manera que el coprocesador ve a la memoria como a 2 bancos, y el CPU ve la memoria como si estuvieran entrelazados los datos (ver figura 1).

Dentro de este sistema de direccionamiento, el Bit 0 define el byte (alto o bajo); el bit 1 define la parte real o imaginaria de los datos, y el bit 2 inicia el direccionamiento.

#### NAPA DE MEMORIA

Los 8 bits mas significativos vienen del bus de datos.

A0-A7 del lado del 8031 se convierten en A2-A9.

A0 distingue entre el byte alto y el byte bajo en el buffer del bus.

A1 distingue al byte alto y al bajo.

A1 del micro es el A0 de las memorias

A10 Selecciona parte real o imaginaria

A11 - A14 Selecciona uso de memoria (0000) o comando (1111)

A15 - Selecciona byte alto o bajo

1K	II	H	II-H. Parte imaginaria alta
1K	II	L	II L Parte imaginaria baja
1K	IR	H	R H Parte Real Alta
1K	IR	L	R H Parte de Memoria de datos.

#### Mapa de Memoria de datos

#### B. Reacomodo de Datos.

Como muchas veces es necesario utilizar un mismo dato en una operación compleja y queremos al mismo tiempo simplificar el controlador del sistema y reducir la velocidad, los datos entran a la memoria del coprocesador en un orden dado y por duplicado y éste mismo se encarga de hacer la reordenación para que los datos puedan sacarse de la memoria simplemente por medio de "pops" o incrementos sencillos del sistema direccionador de memoria y sin tener que recurrir a algoritmos complejos - o a ferretería adicional para tenerlos como se requieren.

Como ejemplo, podemos ver que para una mariposa se necesita tener disponibles a br, bj y wr varias veces durante su ejecución, así que si se mandan los datos duplicados y luego se reacomodan, podemos sacarlos fácilmente de la memoria cuando sean necesarias:

Stack inicial	Reacomodo	Stack Final
br	Reg ← br	br
bj	Mem ← br	bj
ar	reg ← bj; acu ← bj	ar
aj	mem ← bj	bj
	reg ← ar	aj
br	mem ← ar	br
bj	reg ← [acu]	aj
ar	acu ← aj	bj
aj	mem ← reg	ar
	mem ← aj	
	reg ← br	
	reg ← [acu]; mem ← [acu]	
	reg ← bj	
	mem ← bj	
	reg ← ar	
	acu ← ar	

donde la notación  $x \leftarrow y$  indica que  $x$  toma a  $y$ ; y  $x \leftarrow [Y]$  indica que  $x$  toma al contenido de  $y$ .

En estas operaciones, cada línea de control de los circuitos integrados que se relacionan con la ejecución de estas, debe tomar los niveles adecuados. Para conocerlos, se construyen tablas para cada operación:

Se muestra la tabla construida para la mariposa, donde se incluyen las líneas que deben controlarse para cada dispositivo y sus valores para cada microinstrucción.

Esta tabla debe de encontrarse en la ROM de microinstrucciones, y al momento en que se acusa la dirección correspondiente a una microinstrucción, en las líneas de datos se encienden o apagan las líneas adecuadas que controlan y están conectadas a los circuitos integrados.

Después de que se recibe el comando Procedente del 8088, éste se separa en 2 componentes: El código de la instrucción (que viene en la parte alta del comando) y un número (parte baja del comando) que indica cuantos ciclos existen en cada instrucción. La parte alta direcciona el bloque de la ROM de microinstrucciones correspondiente, mientras que la parte baja va a un contador que accesa secuencialmente al contenido de estas localidades para ir ejecutando las microinstrucciones paso a paso. Estos circuitos se encuentran dentro de la unidad de control -- (ver figura).

## C.- EJECUCION DE LAS OPERACIONES

La secuencia de ejecución de microinstrucciones también está dado en función del equilibrio entre la velocidad de ejecución y la simplicidad en las líneas de control:

```

Acu ← br
y ← br; x ← wr
z ← brwr; Reg ← brwr
Acu ← bj
z ← bjwj; Acu ← bjwj
Acu ← brwr - bjwj = R (Bw)
Reg ← ar + IR(Bw)
Mem ← ar + R (Bw)
Reg ← ar - IR (Bw)
Mem ← ar - IR(Bw)
acu ← br ; y ← br

z ← brwj ; Reg ← brwj
x ← wr
y ← bj ; acu ← bj
z ← bjwr; acu ← bjwr
Acu ← bjwr - brwj = II(Bw)
Reg ← aj - II(Bw)
Mem ← aj - II(Bw)
Reg ← aj + II(Bw)
Mem ← aj + II(Bw)

```

MARIPOSA

MICRO INSTRUCCIONES



#### IV.- APLICACIONES Y CONCLUSIONES

El sistema anteriormente descrito es susceptible de ser utilizado para el análisis de señales biomédicas en tiempo real.

Las aplicaciones dentro de este campo pueden ser por ejemplo el filtrado tipo FIR de ordenes mayores a 20 o filtrado tipo Wiener para electrocardiografía: el análisis espectral de otras señales -- tales como fonocardiograma, electroencefalogramas, análisis de -- las señales de presión arterial, ruidos respiratorios. y al procesamiento de imágenes médicas.

La alta velocidad de este sistema permitirá realizar nuevos estudios en el área de señales biológicas, que al entrar dentro del campo del tiempo real permitirán que estas nuevas técnicas dejen de ser únicamente utilizadas dentro de centros de investigación y comiencen a emplearse dentro del campo clínico.

#### BIBLIOGRAFIA

1. Oppenheim, Alan V. and Ronald W. Schaffer  
DIGITAL SIGNAL PROCESSING  
Prentice-Hall, Inc. N.Jersey, 1975
2. Strong, P  
BIOPHYSICAL MEASUREMENTS  
Tektronics Inc. Beaverton, Oregon. 1980
3. Usi, S. and Itzhak Amidror  
"Digital Low-Pass Differentiation for Biological Signal Processing"  
IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING Vol. 29 No.19. Oct. 1982.

4. Rabiner, L.R. and Bernard Gold  
THEORY AND APPLICATION OF DIGITAL SIGNAL PROCESSING  
Prentice-Hall. Inc. N.Jersey, 1974
  5. Larsen, H. and David C. Lai  
"Walsh Spectral Estimates with Applications to the classifica-  
tions of EEG Signals"  
IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING Vol.27 No.9,  
Sept.1980.
  6. Jansen, B.H., Bourne, J.R. and Ward, J.W.  
"Spectral Decomposition of EEG Intervals Using Walsh and  
Fourier Transforms  
IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING Vol. 28 No.2.  
Dic. 1981.
  7. Tarassenko, L., Rolfe, P.Brislow, C.J., Wendling, A.M.  
"Use of Digital Techniques to Process Cerebral Electrical  
Impedance Signals in the Newborn"  
MEDICAL & BIOLOGICAL ENGINEERING & COMPUTING Vol. 22 No.1  
Ene. 1984
  8. Chowdry, S.K. & Majumder, A.K.  
"Digital Spectrum Analysis of Respiratory Sound"  
IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING Vol. 28 No.11,  
1981
  9. Rittgers, S.E., Putney, W.W., Barnes, R.W.  
"Real-Time Spectrum Analysis and Display of Directional  
Doppler Ultrasound Blood Velocity Signals"  
IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING Vol.27 No.12. 1980
  10. Darlow, M.S.  
"Using Personal Computers as Spectrum Analyzers"  
ELECTRONICS WEEK, Vol.57 No.27, Oct.15,1984
  11. IBM CORP.  
IBM PC TECHNICAL REFERENCE MANUAL  
Revised Edition 1983  
Boca Raton, Florida.
-