

# Implementación de un filtro FIR

Herramientas de programación para  
procesamiento de señales

# Índice

- Definición FIR
- Conceptos básicos de algoritmia
- Optimización

# FIR (1)

- SLIT: sistema lineal invariante en el tiempo

- BIBO estable

$$y[n] = x[n] * h[n]$$

- Respuesta al impulso

- Respuesta en frecuencia

$$Y(w) = H(w).X(w)$$

- 

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

# FIR (2)

- Causalidad

$$y[n_0] = h[0].x[n_0] + h[1].x[n_0 - 1] + \dots + h[k - 1].x[n_0 - k + 1]$$

- Retardo de grupo

$$\tau = \frac{dfase(w)}{dw}$$

- Constante

$$\tau = \frac{n}{2}$$

# FIR (3)

- Media móvil (promediador, pasabajos)
  - $h[k]=1/K$

# Implementación (versión 1)

**fir.c**

```
for(int n=0;n<SIGNAL_LENGTH;n++)  
    for(int k=0;k<TAP_LENGTH;k++)  
        y[n]+=x[n-k]*h[k];
```

Una suma extra ( $\frac{1}{4}$  total)

- Usa toda la señal almacenada
- Rápido
- Usa mucha memoria
- Simple

# Complejidad (1)

- Objetivo:
  - Medir el costo computacional de un programa
- Medidas:
  - Almacenamiento
  - Tiempo de ejecución
  - Otras: latencia
- Asintótica:
  - Se estima para  $n$  grande  $n \rightarrow \infty$
  - Notación:

# Complejidad (2)

- Teórica, hipótesis:
  - modelo de máquina RAM
  - Operaciones aritméticas y de memoria
  - Costo constante por cada operación (suma, producto, memoria)
  - No tenemos operaciones complejas (SSE, paralelismo)
- Práctica, factores:
  - fuertemente del compilador
  - Fuertemente de la arquitectura del computador



# Complejidad (3)

- Unidad de medida:
  - Ciclos de reloj
  - relativa
- Costo variable vs costo fijo
  - Absoluto: ciclos
  - Relativos:
    - ciclos/muestra
    - ciclos/retardo/muestra
- tamaños pequeños: costo fijo importante.
- Estrategia:
  - Depurar con caso chico
  - Optimizar con caso grande

# Implementación (versión 2)

- Usa sólo la memoria necesaria
- Buffer de estados
- Corrimiento de memoria

No tomar decisiones a priori!!!!

# Implementación (versión 3)

- Usa sólo la memoria necesaria
- Buffer circular de estados
- Sin corrimiento de memoria
- Verificación de condiciones de borde:
  - if else, poco elegante, eficiente
  - Módulo, elegante, ineficiente

# Medidas

- Implementar para varios tamaños y graficar:
  - Para  $K$  cte graficar ciclos vs  $N$ , variando  $N$
  - Verificar costo asintótico