

Spectrum Analyzer

Lavoro Professionale Individuale

Naoki Pross

28 maggio 2018

SAM Bellinzona

1. Introduzione
2. Fourier Transform
3. Fast Fourier Transform radix-2
4. Prodotto realizzato
5. Conclusioni

Introduzione

A cosa servono l'analisi spettrale e la FT?

Alcuni esempi pratici

Elettronica Filtri digitali, analisi di segnali

Informatica Audio editing, Riconoscimento sonoro / vocale

Matematica Risoluzione di equazioni differenziali

Fisica Principio di indeterminazione di Heisenberg



$$\langle x | \hat{p} | \psi \rangle = -i\hbar \frac{d}{dx} \psi(x)$$

$$\langle x | \hat{x} | \psi \rangle = i\hbar \frac{d}{dp} \psi(p)$$

Realizzare un circuito di analisi spettrale

Requisiti

- Analisi dello spettro fino a 10 kHz
- Entrate Jack e RCA
- Visualizzazione
- Utilizzo di un PIC18F45K22

Realizzare un circuito di analisi spettrale

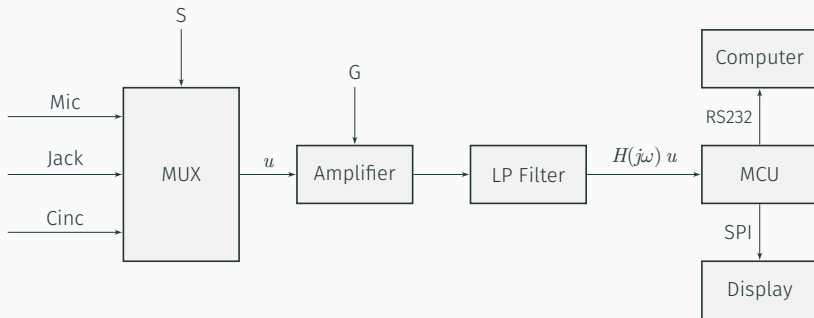
Requisiti

- Analisi dello spettro fino a 10 kHz
- Entrate Jack e RCA
- Visualizzazione
- Utilizzo di un PIC18F45K22

Componenti

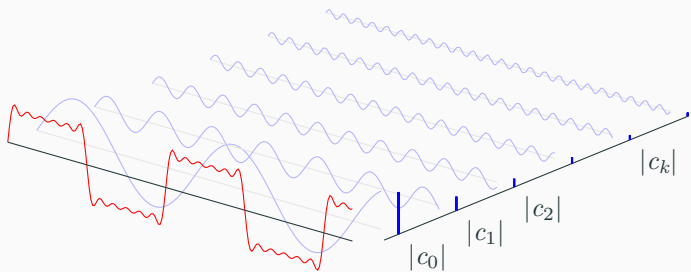
- Circuito di adattamento in entrata
- Design di un PCB
- Software per il uC e per il PC

Schema a blocchi



Fourier Transform

Cos'è la FT?



$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x) \cdot e^{-i\omega x} dx$$

Fast Fourier Transform _{radix-2}

Il problema della DFT

La trasformata di Fourier discreta è

$$\vec{X} = \mathbf{V} \cdot \vec{x}_n$$

In cui

$$\mathbf{V} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & (e^{i\omega})^{-1} & (e^{i\omega})^{-2} & \dots & (e^{i\omega})^{-(N-1)} \\ 1 & (e^{i\omega})^{-2} & (e^{i\omega})^{-4} & \dots & (e^{i\omega})^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & (e^{i\omega})^{-(N-1)} & (e^{i\omega})^{-2(N-1)} & \dots & (e^{i\omega})^{-(N-1)^2} \end{pmatrix}$$

Complessità temporale (analisi asintotica)

Definizione

In informatica, la complessità temporale di un algoritmo quantifica la quantità di tempo impiegata da un algoritmo a essere eseguito.

Complessità temporale (analisi asintotica)

Definizione

In informatica, la complessità temporale di un algoritmo quantifica la quantità di tempo impiegata da un algoritmo a essere eseguito.

Descrivendo il prodotto nelle componenti

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N} \quad 0 \leq k < N$$

Osserviamo che la complessità temporale è $\mathcal{O}(N^2)$.

Complessità temporale (analisi asintotica)

Definizione

In informatica, la complessità temporale di un algoritmo quantifica la quantità di tempo impiegata da un algoritmo a essere eseguito.

Descrivendo il prodotto nelle componenti

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N} \quad 0 \leq k < N$$

Osserviamo che la complessità temporale è $\mathcal{O}(N^2)$.

Come si può migliorare?

Divide et impera (Divide and Conquer)

La FFT appartiene ad una classe di algoritmi chiamata

Divide et Impera

Definizione

1. *Divide* il problema in sottoproblemi
2. *Impera* (conquista) i sottoproblemi
3. *Combina* i risultati dei sottoproblemi

1. Divide in sottoproblemi

Danielson-Lanczos Lemma

$$X_k = E_k + W_N^k \cdot O_k$$

Twiddle factor W_N^k

$$e^{-i2\pi k/N}$$

Termini pari E_k

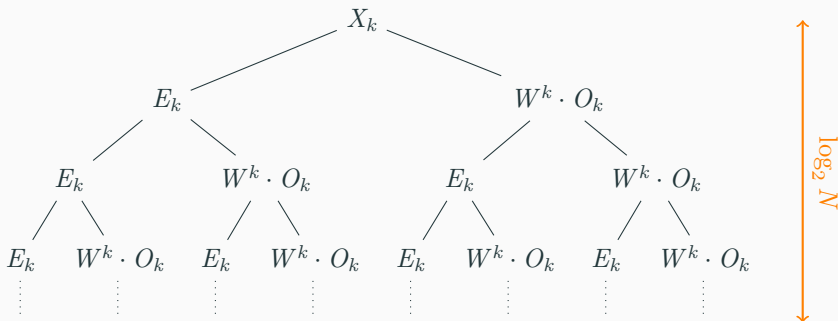
$$\sum_{m=0}^{N/2-1} x_{2m} \cdot e^{-i2\pi mk/(N/2)}$$

Termini dispari O_k

$$\sum_{m=0}^{N/2-1} x_{2m+1} \cdot e^{-i2\pi mk/(N/2)}$$

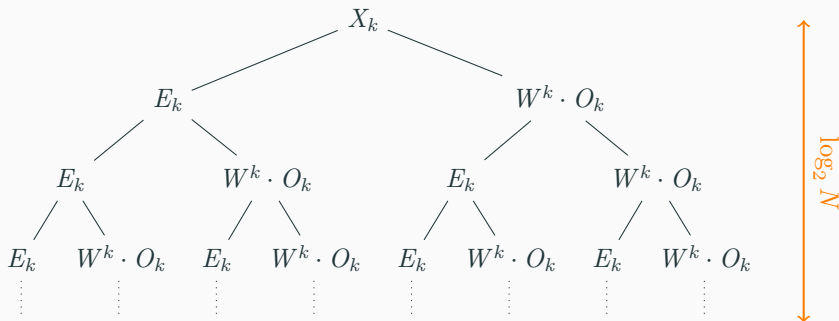
Albero di recursione

E_k e O_k sono delle serie di campioni come $X_k \implies$ recursione!



Albero di recursione

E_k e O_k sono delle serie di campioni come $X_k \Rightarrow$ recursione!



Ogni foglia risulta essere: $\underbrace{W_a^k \cdot W_b^k \cdot W_c^k \dots W_1^k}_{\text{Dipende dalla strada}} \cdot x_m$

2. Impera i sottoproblemi

Ogni coppia di foglie risulta essere

$$E_k + W_1^k \cdot O_k$$

Ossia

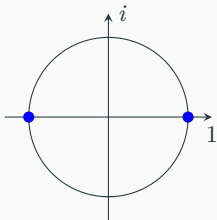
$$\underbrace{W^k \dots W^k}_{\text{Strada}} \cdot x_m + \underbrace{W^k \dots W^k}_{\text{Strada}} \cdot W_1^k \cdot x_n$$

Ridondanza dei twiddle factor

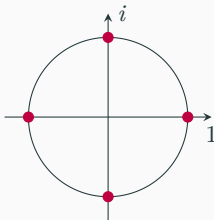
Cosa sono i twiddle factor?

Radici dell'unità $W_N^k = e^{-i2\pi k/N}$

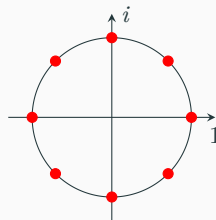
$N = 2$



$N = 4$



$N = 8$

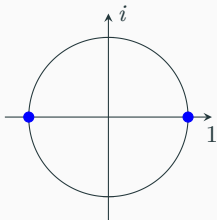


Ridondanza dei twiddle factor

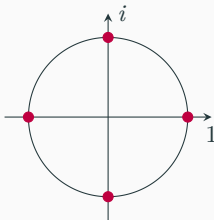
Cosa sono i twiddle factor?

Radici dell'unità $W_N^k = e^{-i2\pi k/N}$

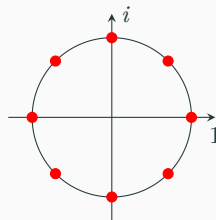
$N = 2$



$N = 4$



$N = 8$



$$W^{k+N/2} = -W^k$$

3. Combina i risultati

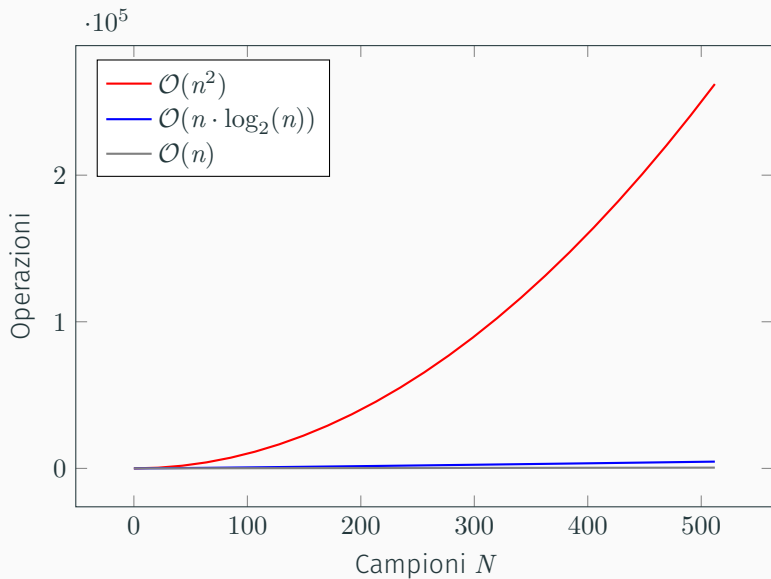
Ciò significa che ad ogni livello di recursione si può trovare

$$\begin{aligned}X_k &= E_k + W^k \cdot O_k \\X_{k+N/2} &= E_k - W^k \cdot O_k\end{aligned}$$

Con l'analisi asintotica si osserva che la complessità temporale è

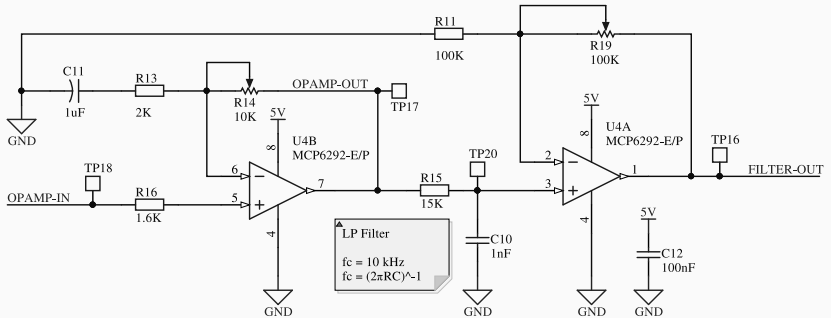
$$\mathcal{O}(N \cdot \log_2(N))$$

Performance



Prodotto realizzato

Signal Adapter



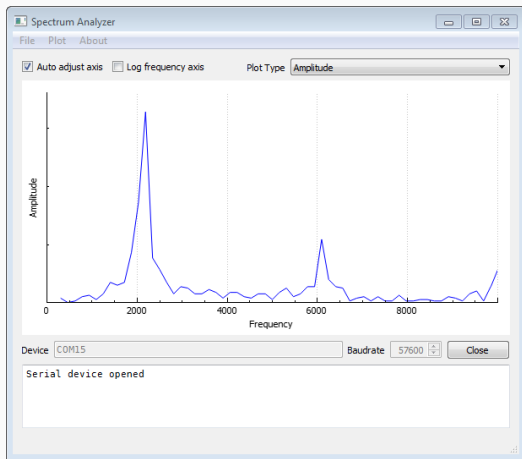
L'implementazione utilizzata si chiama

Fixed-Point In-Place FFT

Fixed-Point Il numero di bit per la mantissa e per l'esponente del valore floating point sono costanti

In-Place I risultati sovrascrivono i dati di partenza

FFT Fast Fourier Transform



Demo!

Conclusioni

Raggiungi

- Analisi spettrale fino a 10 kHz
- Visualizzazione al PC[†]
- Esportare immagini / dati

Raggiungi

- Analisi spettrale fino a 10 kHz
- Visualizzazione al PC[†]
- Esportare immagini / dati

Incompleti

- [†] Visualizzazione delle curve $\Re(z)$ e $\Im(z)$ in un solo grafico
- Visualizzazione mediante la matrice LED

Possibili sviluppi futuri

- Supporto multiplatforma
- Cronologia spettrale

Possibili sviluppi futuri

- Supporto multiplatforma
- Cronologia spettrale

Grazie per l'attenzione