ECE4270 Fundamentals of DSP Lecture 27

1. The Fast Fourier Transform
2. DCT

School of ECE
Center for Signal and Information Processing
Georgia Institute of Technology

Overview of Lecture

- The FFT (Chapter 9)
  - Decimation in time
  - Decimation in frequency
- DCT (Discrete-Cosine Transform)

Decimation-in-Time Derivation - I

- We want to compute *X[k]* efficiently. Divide *x[n]* into even- and odd-indexed

$$X[k] = \sum_{n \text{ even}} x[n]W_N^{nk} + \sum_{n \text{ odd}} x[n]W_N^{nk}$$

- Substituting $n=2r$ and $n=2r+1$ into above gives (assume N is even)

$$X[k] = \sum_{r=0}^{(N/2)-1} x[2r]W_N^{2rk} + \sum_{r=0}^{(N/2)-1} x[2r+1]W_N^{(2r+1)k}$$

$$= \sum_{r=0}^{(N/2)-1} x[2r](W_N^2)^{rk} + W_N^k \sum_{r=0}^{(N/2)-1} x[2r+1](W_N^2)^{rk}$$

Decimation-in-Time Derivation - II

- Using the fact

$$W_N^2 = e^{-2j(2\pi/N)} = e^{-j2\pi/(N/2)} = W_{N/2}$$

it follows that

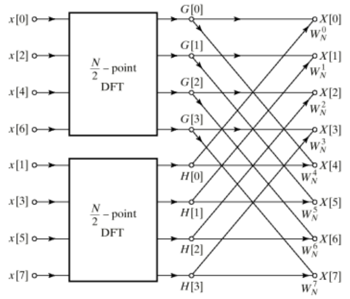
$$X[k] = \sum_{r=0}^{(N/2)-1} x[2r]W_{N/2}^{rk} + W_N^k \sum_{r=0}^{(N/2)-1} x[2r+1]W_{N/2}^{rk}$$

$$= G[k] + W_N^k H[k], \qquad k = 0, 1, \ldots, N-1.$$

In other words, $G[k]$ and $H[k]$ are $N/2$-point DFTs.

$\Rightarrow \mu(N)=N+2\mu(N/2)$

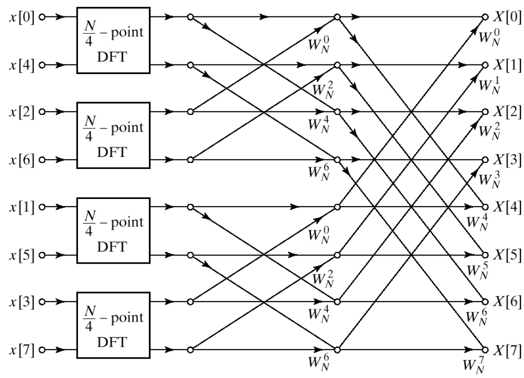
Decimation in Time (First Stage)



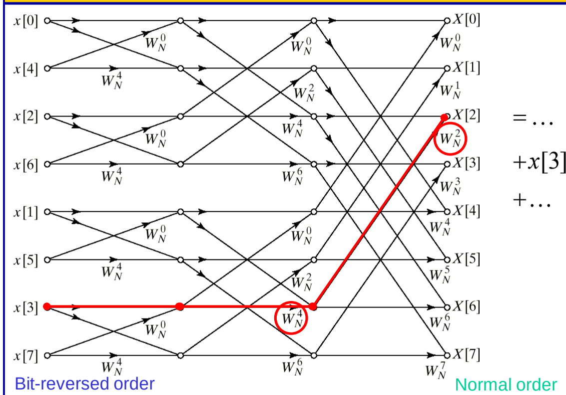
$$X[k] = \sum_{r=0}^{(N/2)-1} x[2r]W_{N/2}^{rk} + W_N^k \sum_{r=0}^{(N/2)-1} x[2r+1]W_{N/2}^{rk}$$

$$= G[k] + W_N^k H[k], \qquad k = 0, 1, \ldots, N-1.$$

Decimation in Time (Second Stage)



Decimation in Time (Third Stage)



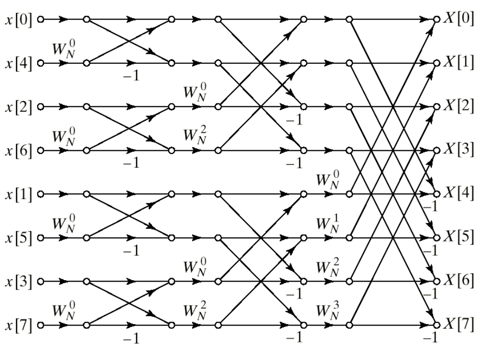
$= \ldots$

$+x[3]W_N^{3\cdot 2}$

$+\ldots$

Decimation in Time (Restructured)

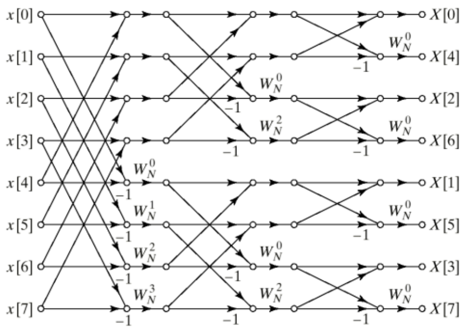
$W_N^{N/2}$
$= e^{-j\pi}$
$= -1$



$\mu(N) = (N/2)\log_2 N$ complex multiplications

Decimation in Frequency

Signal in normal order

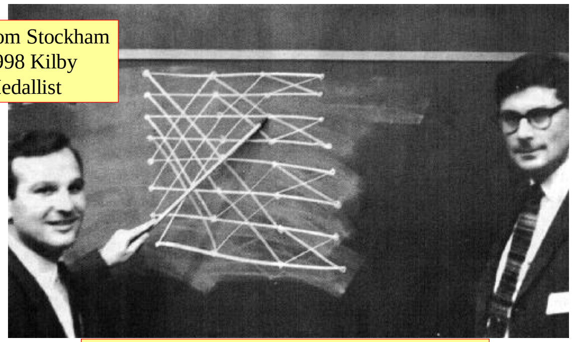


Transform in bit-reversed order

$\Rightarrow \mu(N)=(N/2)\log_2 N$

Tom Stockham and Charlie Rader

Tom Stockham
1998 Kilby Medallist



Inventors of the FFT flow graph representation

FFT Generalizations

- Radix-*R* algorithms:

$$N=R^{\nu}$$

- Mixed-radix algorithms:

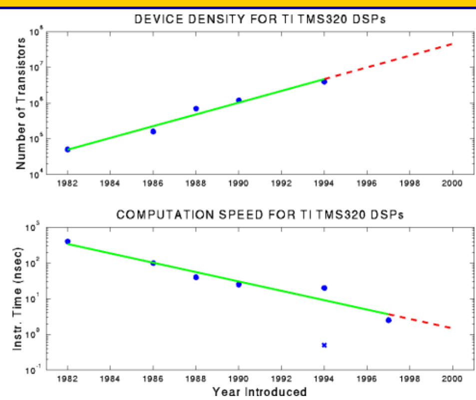
$$N=N_1 N_2 \ldots N_{\nu}$$

- Prime factor algorithms:

$$N=N_1 N_2 \ldots N_{\nu}$$

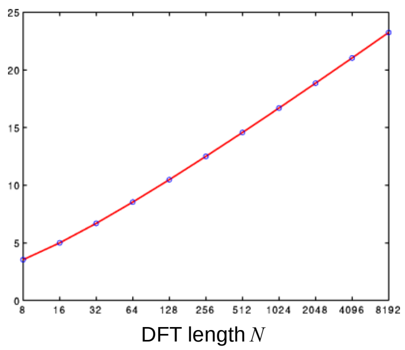
- Winograd and chirp algorithms: based on representing *X[k]* as a convolution

Moore's Law for TI DSPs



Processor Speed vs Algorithm Cleverness

Years of processor development assuming that processor speed doubles every 30 months.



The Discrete Cosine Transform (DCT-2)

- The discrete cosine transform (DCT) is an orthogonal transform (like the DFT) representation that transforms a real finite-length sequence into a real (periodic) sequence

$$\tilde{X}^{c2}[k] = \sqrt{\frac{2}{N}}\tilde{\beta}[k]\sum_{n=0}^{N-1} x[n]\cos\left(\frac{\pi k(2n+1)}{2N}\right), \qquad 0 \le k \le N-1,$$

$$x[n] = \sqrt{\frac{2}{N}}\sum_{k=0}^{N-1} \tilde{\beta}[k]\tilde{X}^{c2}[k]\cos\left(\frac{\pi k(2n+1)}{2N}\right), \qquad 0 \le n \le N-1,$$

$$\tilde{\beta}[k] = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0, \\ 1, & k = 1, 2, \ldots, N-1. \end{cases}$$

Relation of DCT-2 to the DFT

- Form the 2N-point periodic sequence

$$x_2[n] = x[((n))_{2N}] + x[((-n-1))_{2N}]$$

$$X_2[k] = X[k] + X^*[k]e^{j2\pi k/(2N)}, \qquad k = 0, 1, \ldots, 2N-1$$

$X[k]$ is the $2N$ - Point DFT of $x[n]$

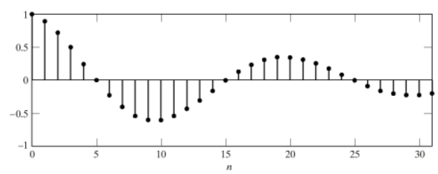
- The DCT-2 in terms of the 2N-point DFT is

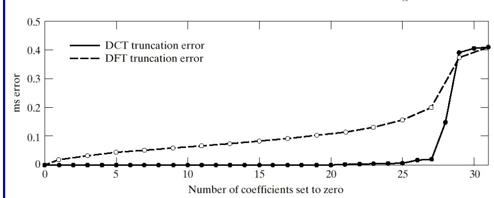
$$X^{c2}[k] = 2\mathcal{R}e\left\{X[k]e^{-j\pi k/(2N)}\right\}, \qquad k = 0, 1, \ldots, N-1$$

$$X^{c2}[k] = e^{-j\pi k/(2N)}X_2[k], \qquad k = 0, 1, \ldots, N-1$$

Energy Compaction in the DCT

Test signal





Truncation error

Comparison of DFT and DCT

Real part of N-point DFT

Imaginary part of N-point DFT

DCT-2