* D.F.T. Computation:

FFT (1)

Regular computation $\underline{X} = \underline{W}_N \, \underline{x}$

$$X[0] = \sum_{n=0}^{N-1} x[n] \, e^{-j\frac{2\pi}{N}\cdot 0 \cdot n}$$

complex N multiplications are required

$$X[1] = \sum_{n=0}^{N-1} x[n] \, e^{-j\frac{2\pi}{N}\cdot 1 \cdot n}$$

$$\vdots$$

$$X[N-1] = \sum_{n=0}^{N-1} x[n] \, e^{-j\frac{2\pi}{N}(N-1)\cdot n}$$

Total computational cost to obtain $X[0], X[1], \ldots, X[N-1]$ = N^2 complex multiplications!

Decimation-in-Frequency F.T. Computation Algorithm

(Cooley & Tukey) Divide and conquer. $\underline{\underline{N = 2^P}}$

$$X[k] = \sum_{n=0}^{\frac{N}{2}-1} x[n] \, W_N^{kn} + \sum_{n=\frac{N}{2}}^{N-1} x[n] \, W_N^{kn}, \quad W_N = e^{-j\frac{2\pi}{N}}$$

$$X[k] = \sum_{n=0}^{N/2-1} x[n] \, W_N^{kn} + \sum_{l=0}^{N/2-1} x[l+\tfrac{N}{2}] \, W_N^{kl} \cdot W_N^{\frac{N}{2}k}, \quad n = l + \tfrac{N}{2}$$

$$l \leftarrow n$$

$$X[k] = \sum_{n=0}^{N/2-1} x[n] \, W_N^{kn} + (-1)^k \sum_{n=0}^{\frac{N}{2}-1} x[n+\tfrac{N}{2}] \, W_N^{kn}$$

$$X[k] = \sum_{n=0}^{N/2-1} \left(x[n] + (-1)^k x[n+\tfrac{N}{2}] \right) W_N^{kn}, \quad k = 0, 1, \ldots, N-1$$

Down-sample $X[k]$ into even and odd samples

$k \leftarrow 2l$ $$X[2l] = \sum_{n=0}^{N/2-1} \left(x[n] + (-1)^{2l} x[n+\tfrac{N}{2}] \right) W_N^{2ln}, \quad l = 0, 1, 2, \ldots, \tfrac{N}{2}-1$$

$k \leftarrow 2l+1$ $$X[2l+1] = \sum_{n=0}^{N/2-1} \left(x[n] + (-1)^{2l+1} x[n+\tfrac{N}{2}] \right) W_N^{(2l+1)n}, \quad l = 0, 1, 2, \ldots, \tfrac{N}{2}-1$$

FFT ②

(1*) $$X[2l] = \sum_{n=0}^{N/2-1} g[n]\, W_{N/2}^{ln}, \quad l = 0, 1, \cdots, \frac{N}{2}-1$$

where $g[n] = \left(x[n] + x\left[n + \frac{N}{2}\right]\right)$

So Eq.(1*) is $\frac{N}{2}$-point DFT of $g[n]$, $n = 0, 1, \cdots, \frac{N}{2}-1$

$$G[l] = \sum_{n=0}^{N/2-1} g[n]\, W_{N/2}^{ln}, \quad l = 0, 1, \cdots, N/2 - 1.$$

(2*) $$X[2l+1] = \sum_{n=0}^{N/2-1} \tilde{h}[n]\, W_N^{2ln}\, W_N^{n}, \quad l = 0, 1, \cdots, \frac{N}{2}-1.$$

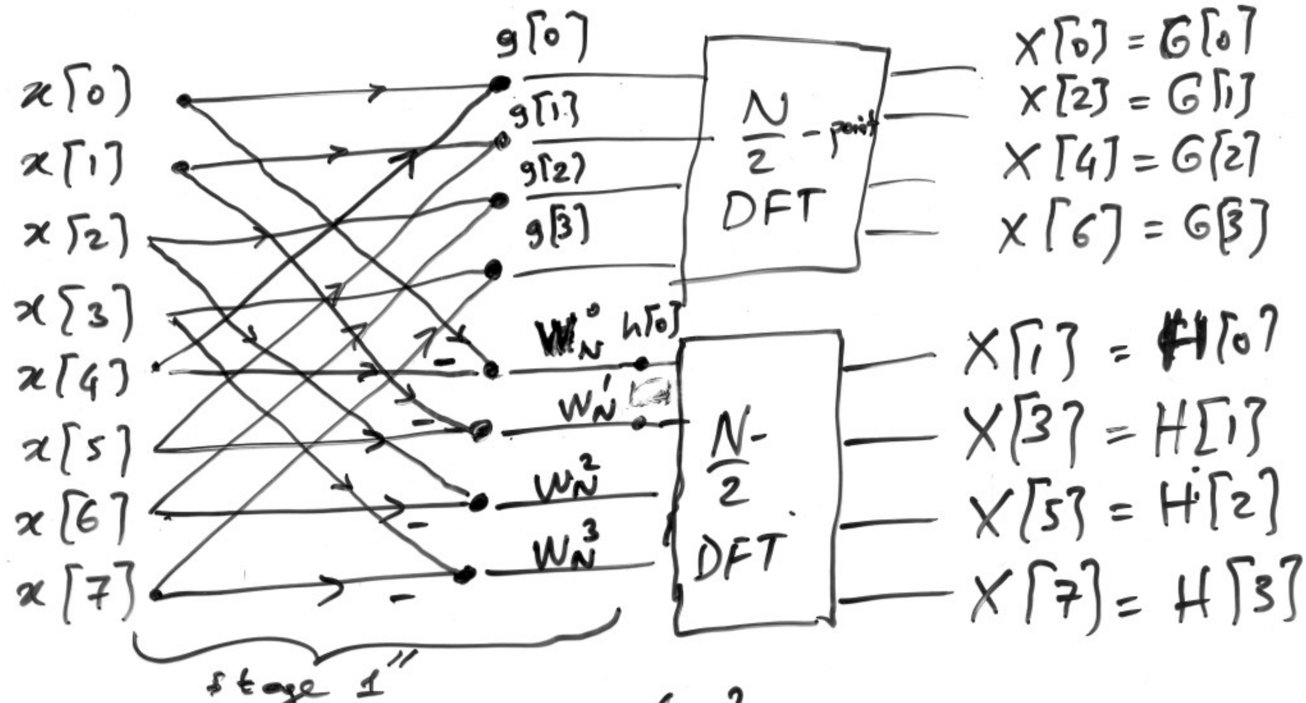
where $\tilde{h}[n] = x[n] - x\left[n + \frac{N}{2}\right]$, $\left((-1)^{2l+1} = -1\right)$.

Eq (2*) is the $\frac{N}{2}$-point DFT of

$$h[n] = \left(x[n] - x\left[n + \frac{N}{2}\right]\right) W_N^{n}, \quad n = 0, 1, \cdots, \frac{N}{2}-1.$$

$$H[l] = \sum_{n=0}^{N/2-1} h[n]\, W_{N/2}^{ln}, \quad l = 0, 1, \cdots, \frac{N}{2}-1.$$

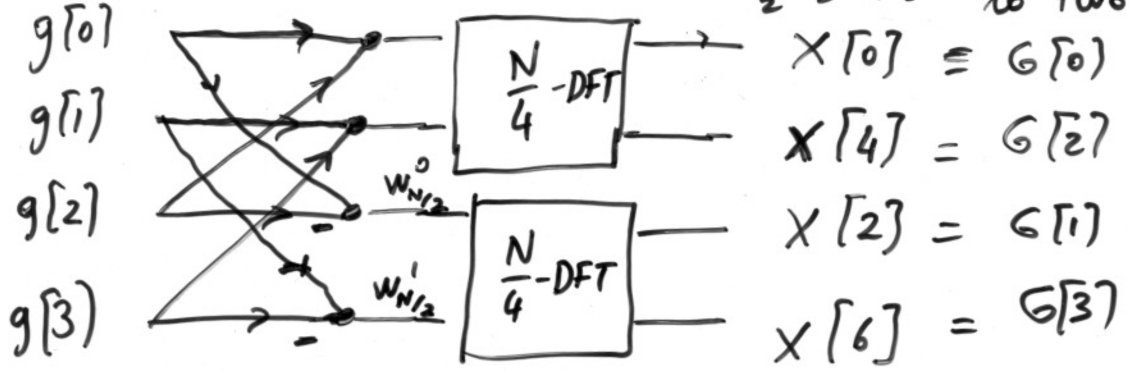
N=8

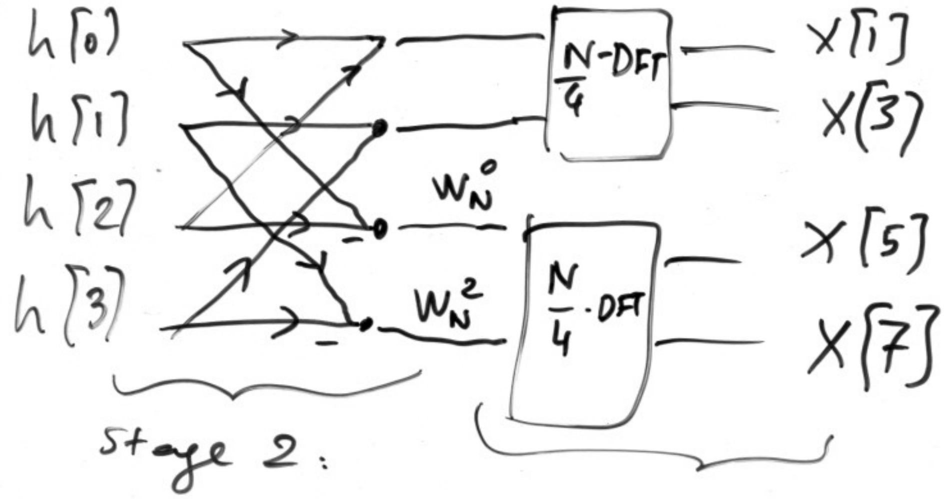


Comp. cost $= \frac{N}{2}$ complex ⊗ $+ \left(\frac{N}{2}\right)^2 \cdot 2 = \frac{N^2}{2} + \frac{N}{2}$ compl ⊗

Use (1*) & (2*) to divide $\frac{N}{2}$-DFT's to two $\frac{N}{4}$ DFT's:

FFT 3





Stage 2:

Comp. cost $\frac{N}{2}$ complex $\otimes$ + $4 \cdot \left(\frac{N}{4}\right)^2$ complex $\otimes$.

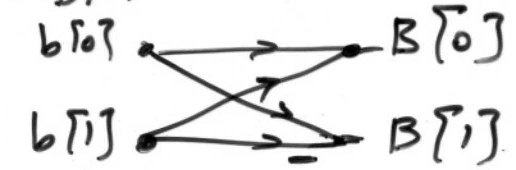
For $N = 8 \Rightarrow \frac{N}{4} = 2$

$N = 2$-point DFT: $B[k] = \sum_{n=0}^{1} b[n]\, W_N^{kn}$, $k = 0, 1$.

$$B[0] = b[0] + b[1] = b[0] + b[1]$$

$$B[1] = b[0]\, W_N^0 + b[1]\, W^1_{N=2} = b[0] - b[1]$$

$\frac{N}{4} = 2$ - DFT

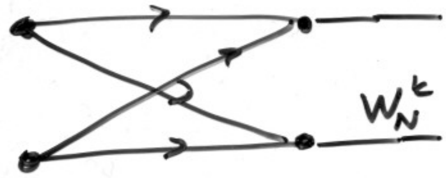


The $N = 8 = 2^3$ point DFT can be computed in $p = 3 = \log_2 8$ stages & in each stage we perform $4 = \frac{N}{2}$ complex $\otimes$.

Computational cost of Decimation-in-freq. algorithm: for $N = 2^p$-point DFT:

$$p \cdot \frac{N}{2} \text{ complex } \otimes = \frac{N}{2} \log_2 N \text{ complex multiplications}$$

* Butterfly's are the building blocks of the radix-2 FFT algorithm.



$N = 2^p$

* In Deci. in. freq algorithm the input vector is in an orderly fashion.

$x[0] = x[000]$ — $X[0] = X[000]$

$x[1] = x[001]$ — $X[4] = X[100]$

$x[2] = x[010]$ — $X[2] = X[010]$

$x[3] = x[011]$ — $X[6] = X[110]$

$x[4] = x[100]$ — $X[1] = X[001]$

$\vdots$

$x[7] = x[111]$ — $X[7] = X[111]$

FFT algorithm

bit reversed output

bit-reversed order

It works for all $N = 2^p$.

* One complex $\otimes$ is equivalent to 4 real $\otimes$.
So the cost of N-FFT $= 2N \log_2 N$ real multiplications for complex input $x[n]$.