Cooley-Tukey FFT Algorithms

- Consider a length-N sequence $x[n]$ with an N-point DFT $X[k]$ where $N = N_1 N_2$
- Represent the indices n and k as

$$n = N_2 n_1 + n_2, \quad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases}$$

$$k = k_1 + N_1 k_2, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases}$$

Cooley-Tukey FFT Algorithms

- Using these index mappings we can write

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}$$

as

$$X[k] = X[k_1 + N_1 k_2]$$

$$= \sum_{n_2=0}^{N_2-1} \sum_{n_1=0}^{N_1-1} x[N_2 n_1 + n_2] W_N^{(N_2 n_1 + n_2)(k_1 + N_1 k_2)}$$

$$= \sum_{n_2=0}^{N_2-1} \sum_{n_1=0}^{N_1-1} x[N_2 n_1 + n_2] W_N^{N_2 n_1 k_1} W_N^{n_2 k_1} W_N^{N_1 n_2 k_2} W_N^{N_1 N_2 n_1 k_2}$$

Cooley-Tukey FFT Algorithms

- Since $W_N^{N_2 n_1 k_1} = W_{N_1}^{n_1 k_1}$, $W_N^{N_1 n_2 k_2} = W_{N_2}^{n_2 k_2}$, and $W_N^{N_1 N_2 n_1 k_2} = 1$, we have

$$X[k_1 + N_1 k_2]$$

$$= \sum_{n_2=0}^{N_2-1} \left[\left(\sum_{n_1=0}^{N_1-1} x[N_2 n_1 + n_2] W_{N_1}^{n_1 k_1} \right) W_N^{n_2 k_1} \right] W_{N_2}^{n_2 k_2}$$

where $0 \le k_1 \le N_1 - 1$ and $0 \le k_2 \le N_2 - 1$

Cooley-Tukey FFT Algorithms

- The effect of the index mapping is to map the 1-D sequence $x[n]$ into a 2-D sequence that can be represented as a 2-D array with n_1 specifying the rows and n_2 specifying the columns of the array
- Inner parentheses of the last equation is seen to be the set of N_1-point DFTs of the N_2-columns:

$$G[k_1, n_2] = \sum_{n_1=0}^{N_1-1} x[N_2 n_1 + n_2] W_{N_1}^{n_1 k_1}, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases}$$

Cooley-Tukey FFT Algorithms

- <u>Note</u>: The column DFTs can be done in place
- Next, these row DFTs are multiplied in place by the twiddle factors $W_N^{n_2 k_1}$ yielding

$$\tilde{G}[k_1, n_2] = W_N^{n_2 k_1} G[k_1, n_2], \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases}$$

- Finally, the outer sum is the set of N_2-point DFTs of the columns of the array:

$$X[k_1 + N_1 k_2] = \sum_{n_2=0}^{N_2-1} \tilde{G}[k_1, n_2] W_{N_2}^{n_2 k_2}, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases}$$

Cooley-Tukey FFT Algorithms

- The row DFTs, $X[k_1 + N_1 k_2]$, can again be computed in place
- The input $x[n]$ is entered into an array according to the index map:

$$n = N_2 n_1 + n_2, \quad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases}$$

- Likewise, the output DFT samples $X[k]$ need to extracted from the array according to the index map:

$$k = k_1 + N_1 k_2, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases}$$

Cooley-Tukey FFT Algorithms

- Example - Let $N = 8$. Choose $N_1 = 2$ and $N_2 = 4$
- Then

$$X[k_1 + 2k_2] = \sum_{n_2=0}^{3}\left[\left(\sum_{n_1=0}^{1} x[4n_1 + n_2]W_2^{k_1 n_1}\right)W_8^{k_1 n_2}\right]W_4^{k_2 n_2}$$

for $0 \le k_1 \le 1$ and $0 \le k_2 \le 3$

Cooley-Tukey FFT Algorithms

- 2-D array representation of the input is

$n_1 \backslash n_2$	0	1	2	3
0	x[0]	x[1]	x[2]	x[3]
1	x[4]	x[5]	x[6]	x[7]

- The column DFTs are 2-point DFTs given by

$$G[k_1, n_2] = x[n_2] + (-1)^{k_1} x[4 + n_2], \quad \begin{cases} 0 \le k_1 \le 1 \\ 0 \le n_2 \le 3 \end{cases}$$

- These DFTs require no multiplications

Cooley-Tukey FFT Algorithms

- 2-D array of row transforms is

$k_1 \backslash n_2$	0	1	2	3
0	$G[0,0]$	$G[0,1]$	$G[0,2]$	$G[0,3]$
1	$G[1,0]$	$G[1,1]$	$G[1,2]$	$G[1,3]$

- After multiplying by the twiddle factors $W_8^{n_2 k_1}$ array becomes

$k_1 \backslash n_2$	0	1	2	3
0	$\tilde{G}[0,0]$	$\tilde{G}[0,1]$	$\tilde{G}[0,2]$	$\tilde{G}[0,3]$
1	$\tilde{G}[1,0]$	$\tilde{G}[1,1]$	$\tilde{G}[1,2]$	$\tilde{G}[1,3]$

Cooley-Tukey FFT Algorithms

- Note: $\tilde{G}[k_1, n_2] = W_8^{n_2 k_1} G[k_1, n_2]$
- Finally, the 4-point DFTs of the rows are computed:

$$X[k_1 + 2k_2] = \sum_{n_2=0}^{3} \tilde{G}[k_1, n_2] W_4^{n_2 k_2}, \quad \begin{cases} 0 \le k_1 \le 1 \\ 0 \le k_2 \le 3 \end{cases}$$

- Output 2-D array is given by

$k_1 \backslash k_2$	0	1	2	3
0	X[0]	X[2]	X[4]	X[6]
1	X[1]	X[3]	X[5]	X[7]

Cooley-Tukey FFT Algorithms

- The process illustrated is precisely the first stage of the DIF FFT algorithm
- By choosing $N_1 = 4$ and $N_2 = 2$, we get the first stage of the DIT FFT algorithm
- Alternate index mappings are given by

$$n = n_1 + N_1 n_2, \quad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases}$$

$$k = N_2 k_1 + k_2, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases}$$