Digital Signal Processing

Fast Fourier Transform (FFT)

Mifta Nur Farid

Sept. 4st, 2023

ELEKTRO ITK
itk Institut Teknologi Kalimantan

Fast Fourier Transform (FFT)

- very efficient algorithm in computing DFT coefficients $X(k)$
- can reduce a very large amount of computational complexity (multiplications)
- consider the digital sequence $x(n)$ consisting of 2^m samples, where m is a positive integer m = 1, 2, 3, dst N = 2^m
  - $N = 2, 4, 8, 16,$ etc
  - If $x(n)$ does not contain 2^m samples, then we simply append it with zeros until the number of the appended sequence is a power of 2

Fast Fourier Transform (FFT)

- Consider $x(n) = [1, 2, 3]$. Can we do the FFT to $x(n)$?

$N = 3$

Fast Fourier Transform (FFT)

- Consider $x(n) = [1, 2, 3]$. Can we do the FFT to $x(n)$? **No**

$$\underset{2}{\underset{\shortparallel}{2^1}} < N = 3 < \underset{4}{\underset{\shortparallel}{2^2}} \qquad N = 2^m$$

Fast Fourier Transform (FFT)

- Consider $x(n) = [1, 2, 3]$. Can we do the FFT to $x(n)$? **No**
- Because $x(n)$ does not contain 2^m samples

Fast Fourier Transform (FFT)

- Consider $x(n) = [1, 2, 3]$. Can we do the FFT to $x(n)$? **No**
- Because $x(n)$ does not contain 2^m samples
- Solution?

Fast Fourier Transform (FFT)

$2 = 2^1 < N = 3 < 2^2 = 4$

- Consider $x(n) = [1, 2, 3]$. Can we do the FFT to $x(n)$? **No**
- Because $x(n)$ does not contain 2^m samples
- Solution? **Append it with zeros**, $x(n) = [1, 2, 3, 0]$

zero padding.

$x(n) = [1, 2, 3, 4, 5] \longrightarrow x(n) = [1, 2, 3, 4, 5, 0, 0, 0]$

$N = 5 \longrightarrow$ zero padding $N = 8$ $\quad N = 8$

Fast Fourier Transform (FFT)

- we focus on two formats of the radix-2 FFT algorithms:
  1. The decimation-in-frequency algorithm
  2. The decimation-in-time algorithm

Decimation-in-frequency algorithm

- DFT:

banyaknya sample

Twiddle Factor $= e^{\frac{-j2\pi kn}{N}}$

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad \text{for} \quad k = 0, 1, \cdots, N-1,$$

$$N = 2,\ 4,\ 8,\ 16,\ \cdots$$

- can be expanded as

$$\boxed{X(k) = x(0) + x(1)W_N^k + \cdots + x(N-1)W_N^{k(N-1)}}$$

- if we split:

$$X(k) = \boxed{x(0) + x(1)W_N^k + \cdots + x\left(\frac{N}{2} - 1\right)W_N^{k(N/2-1)}}$$

setengah pertama

$$\boxed{+x\left(\frac{N}{2}\right)W^{kN/2} + \cdots + x(N-1)W_N^{k(N-1)},}$$

setengah kedua

Decimation-in-frequency algorithm

awal

akhir

- equation:

$$X(k) = x(0) + x(1)W_N^k + \cdots + x\left(\frac{N}{2} - 1\right)W_N^{k(N/2-1)}$$
$$+ x\left(\frac{N}{2}\right)W^{kN/2} + \cdots + x(N-1)W_N^{k(N-1)},$$

- can be rewriten as a sum of the following two parts:

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n)W_N^{kn} + \sum_{n=N/2}^{N-1} x(n)W_N^{kn}.$$

Decimation-in-frequency algorithm

- equation

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n) W_N^{kn} + \boxed{\sum_{n=N/2}^{N-1} x(n) W_N^{kn}}.$$

- modifying the second term:

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n) W_N^{kn} + \boxed{W_N^{(N/2)k}} \sum_{n=0}^{(N/2)-1} x\left(n + \frac{N}{2}\right) W_N^{kn}$$

-1

Decimation-in-frequency algorithm

$$e^{-j\pi} = \cos\pi - j\sin\pi$$
$$= -1 - j \cdot 0$$
$$= -1$$

- because

$$W_N^{N/2} = e^{-j\frac{\cancel{2}\pi(N/\cancel{2})}{\cancel{N}}} = e^{-j\pi} = -1$$

- then

$$X(k) = \sum_{n=0}^{(N/2)-1} \left(x(n) + (-1)^k x\left(n + \frac{N}{2}\right) \right) W_N^{kn}$$

Decimation-in-frequency algorithm

- $k = 2m$ as an even number:

$$X(2m) = \sum_{n=0}^{(N/2)-1} \left(x(n) + x\left(n + \frac{N}{2}\right) \right) \boxed{W_N^{2mn}}$$

- $k = 2m + 1$ as an odd number:

$$X(2m+1) = \sum_{n=0}^{(N/2)-1} \left(x(n) - x\left(n + \frac{N}{2}\right) \right) W_N^n W_N^{2mn}$$

Decimation-in-frequency algorithm

$W_N^2 = W_{N/2}$
$W_N^3 = W_{N/3}$

- because

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

- then

even $$X(2m) = \sum_{n=0}^{(N/2)-1} a(n) W_{N/2}^{mn} = \text{DFT}\{a(n) \;\; \text{with} \;\; (N/2)\text{points}\},$$

odd $$X(2m+1) = \sum_{n=0}^{(N/2)-1} b(n) W_N^n W_{N/2}^{mn} = \text{DFT}\left\{b(n) W_N^n \;\; \text{with} \;\; (N/2)\text{points}\right\}$$

Decimation-in-frequency algorithm

$$\underset{2m}{X(k)} = \sum_{n=0}^{N-1} x(n) W_{N/2}^{\overset{2m}{kn}}$$

- equation:

$$\rightarrow \quad X(2m) = \sum_{n=0}^{(N/2)-1} a(n) W_{N/2}^{mn} = \text{DFT}\{a(n) \;\text{ with }\; (N/2)\text{points}\},$$

$$\rightarrow \quad X(2m+1) = \sum_{n=0}^{(N/2)-1} b(n) W_N^n W_{N/2}^{mn} = \text{DFT}\{b(n) W_N^n \;\text{ with }\; (N/2)\text{points}\}$$

- where $a(n)$ and $b(n)$ are introduced and expressed as

$$a(n) = x(n) + x\left(n + \frac{N}{2}\right), \quad \text{for} \quad n = 0, 1 \cdots, \frac{N}{2} - 1,$$

$$b(n) = x(n) - x\left(n + \frac{N}{2}\right), \quad \text{for} \quad n = 0, 1, \cdots, \frac{N}{2} - 1.$$

Decimation-in-frequency algorithm

- can be summarized as

FFT

$$\text{DFT}\{x(n) \text{ with } N \text{ points}\} = \begin{cases} \text{DFT}\{a(n) \text{ with} (N/2) \text{points}\} \\ \text{DFT}\{b(n)W_N^n \text{ with} (N/2) \text{points}\} \end{cases}$$

- comparison

$N = 8$

$$\text{Complex multiplications of } \boxed{\text{DFT} = N^2}, \text{ and} \rightarrow 8^2 = 64 \text{ comp. multi.}$$

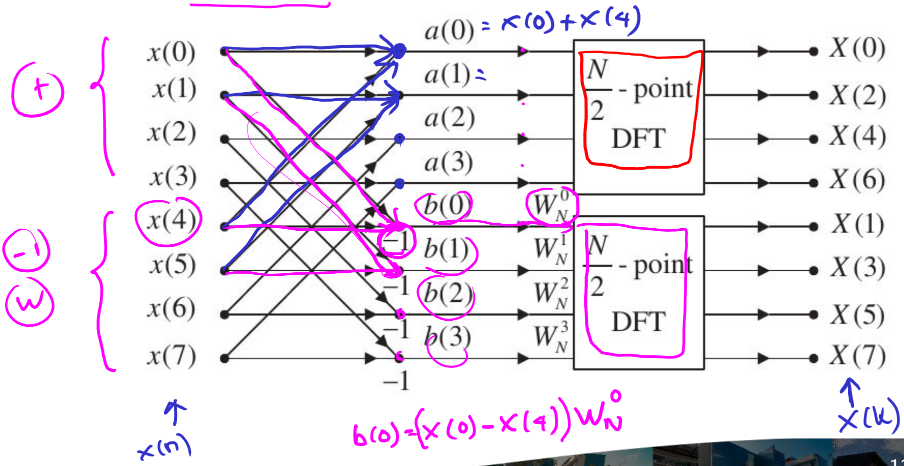
$$\text{Complex multiplications of FFT} = \boxed{\frac{N}{2}\log_2(N).} \rightarrow \frac{8}{2}\log_2(8)$$

$4 \cdot 3 = 12$ comp. mult.

Decimation-in-frequency algorithm

1-stage.

- The first iteration of the eight-point FFT



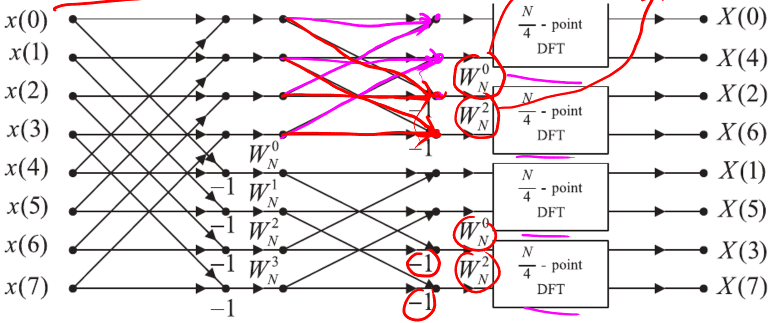
Decimation-in-frequency algorithm

2-stage

$W_{N/2}^{0/2} = W_{N/2}^{0}$

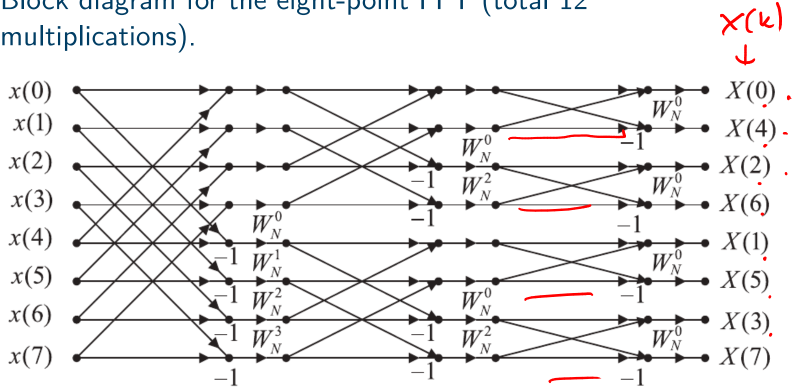
$W_{N/2}^{2/2} = W_{N/2}^{1}$

- The second iteration of the eight-point FFT



Decimation-in-frequency algorithm

- Block diagram for the eight-point FFT (total 12 multiplications).



Decimation-in-frequency algorithm

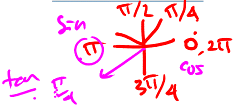
$N=8 \rightarrow 2^{m} = 2^{3}$ bit

Table 4.2 Index Mapping for Fast Fourier Transform

Input Data	Index Bits	Reversal Bits ← x(k)	Output Data
$x(0)$ 0	000	000	$X(0)$
$x(1)$	001	100	$X(4)$
$x(2)$	010	010	$X(2)$
$x(3)$	011	110	$X(6)$
$x(4)$	100	001	$X(1)$
$x(5)$	101	101	$X(5)$
$x(6)$	110	011	$X(3)$
$x(7)$ 7	111	111	$X(7)$

Decimation-in-frequency algorithm

$\frac{5\pi}{4} = \frac{4}{4}\pi + \frac{\pi}{4}$, $\frac{4}{4}\pi = \pi$



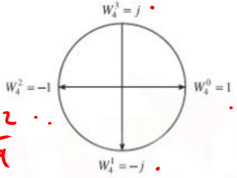
$W_8^5 = e^{-j\frac{2\pi 5}{8}} = e^{-j\frac{5\pi}{4}} = \cos\frac{5}{4}\pi - \sin\frac{5}{4}\pi$

$= -\cos\frac{\pi}{4} + \sin\frac{\pi}{4}\pi$

$=$

$W_4^3 = e^{-j\frac{2\pi 3}{4}} = j$

For a 4-point DFT



(a) 4-point DFT

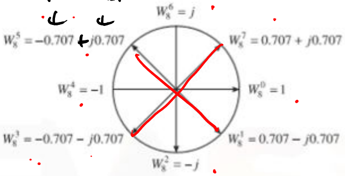
$W_4^2 = e^{-j\frac{2\pi\cdot 2}{4}} = -1$

$W_4^0 = e^0 = 1$

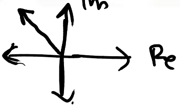
$W_4^1 = e^{-j\frac{2\pi\cdot 1}{4}} = -j$

For an 8-point DFT

Re Im



(b) 8-point DFT



Decimation-in-frequency algorithm

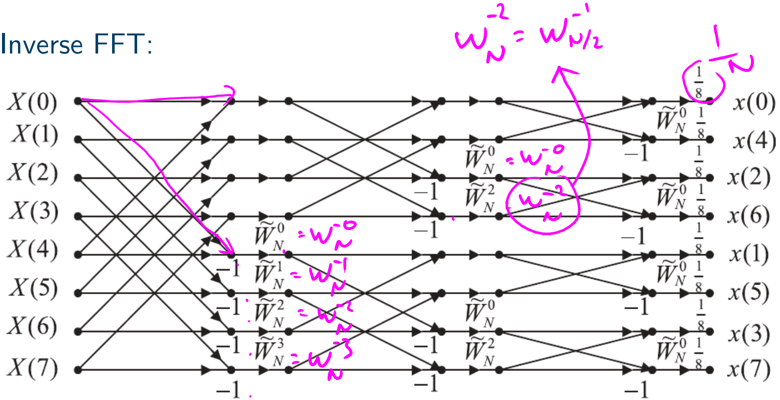
$$\tilde{W}_N^{kn} = W_N^{-kn}$$

- Inverse FFT:

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\boxed{W_N^{-kn}} = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\tilde{W}_N^{kn}, \text{ for } k = 0, 1, \cdots, N-1.$$

Decimation-in-frequency algorithm

- Inverse FFT:



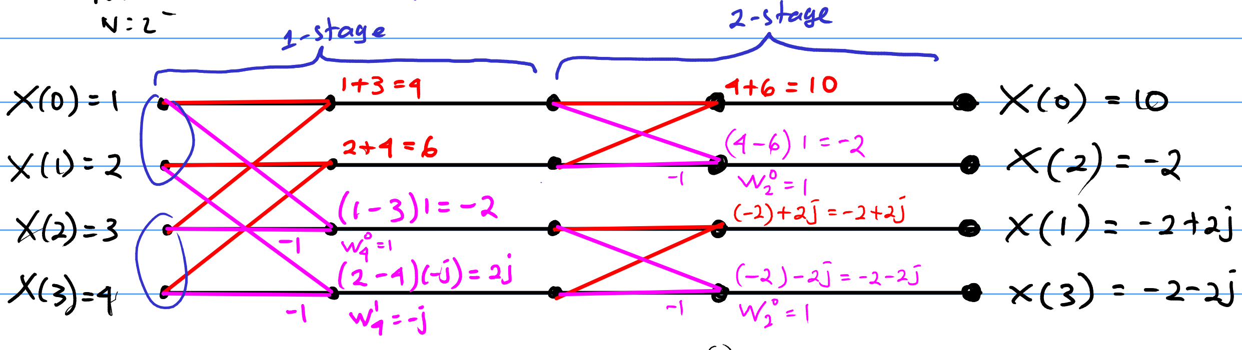
Example 1

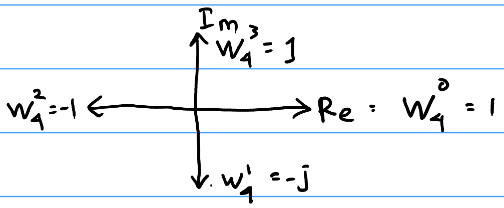
Given a sequence $x(n)$ for $0 \leq n \leq 3$, where $x(0) = 1$, $x(1) = 2$, $x(2) = 3$, and $x(3) = 4$, evaluate its DFT $X(k)$ using the decimation-in-frequency FFT method.

DIF - FFT $\qquad x(n) = [1, 2, 3, 4]$

N=8
N=4
N=2

$x(n) = [1, 2, 3, 4]$: X(k) hasil DIF-FFT





$N=4=2^{(2)}$

index dec	bits	reversal bits	dec
0 →	00	00 →	0
1 →	01	10 →	2
2 →	10	01 →	1
3 →	11	11 →	3

$x(n) = \{1,2,3,4,5\}$; Berapa X(k) dengan menggunakan DIF-FFT?

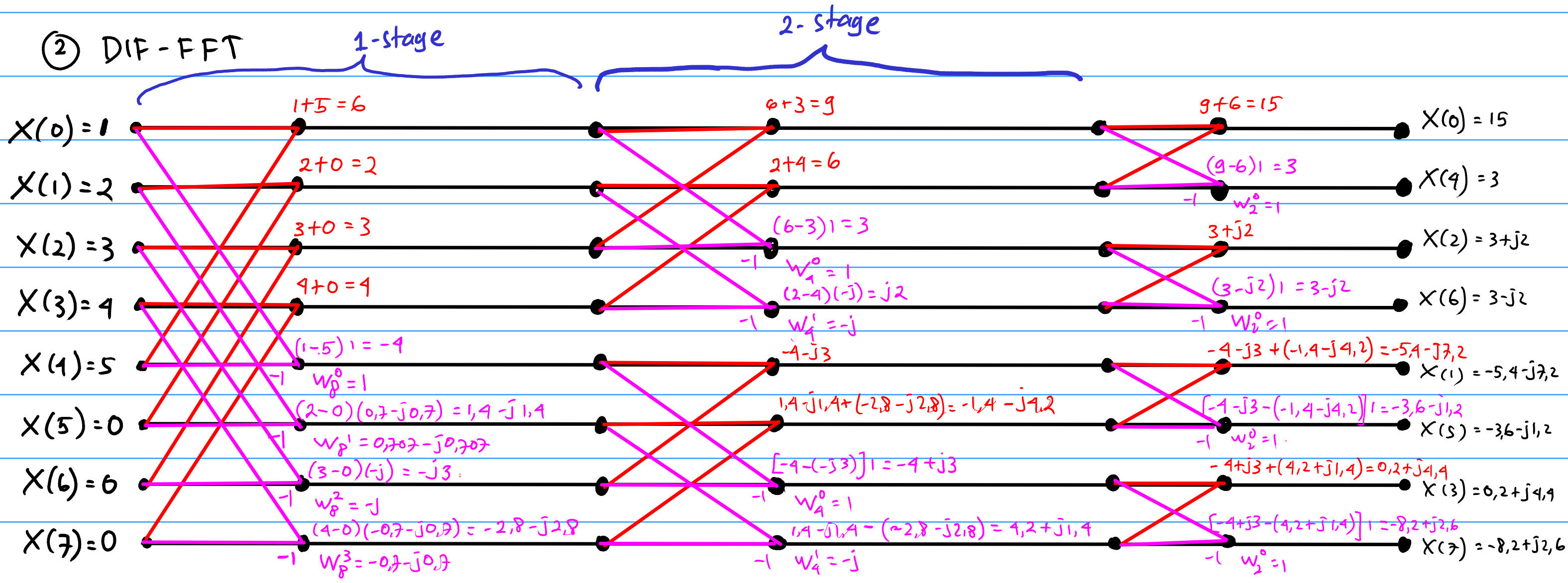
N = 5

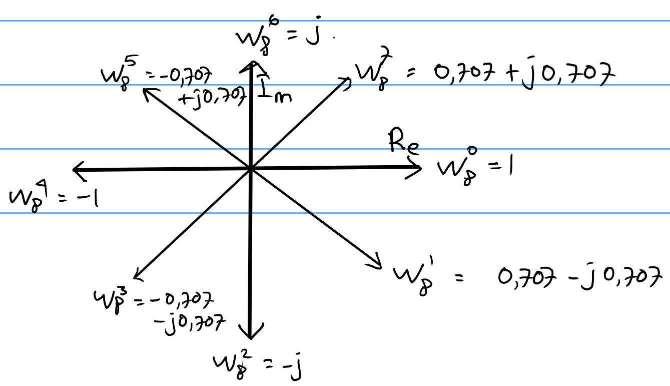
① Zero padding. krn (N=5). utk FFT, $N = 2, 4, 8, 16, 32, \cdots 2^m$

$x(n) = \{1,2,3,4,5,0,0,0\}$

N = 8

② DIF-FFT





index bits		reversal bit	
0 →	000	000 →	0
1 →	001	100 →	4
2 →	010	010 →	2
3 →	011	110 →	6
4 →	100	001 →	1
5 →	101	101 →	5
6 →	110	011 →	3
7 →	111	111 →	7

X(k) = [15; -5,4-j7,2; 3+j2; 0,2+j4,4; 3; -3,6-j1,2; 3-j2; -8,2+j2,6]

Decimation-in-time algorithm

DIT-FFT

- we split the input sequence $x(n)$ into the even indexed $x(2m)$ and $x(2m+1)$, each with N data points

$$X(k) = \underbrace{\sum_{m=0}^{(N/2)-1} x(2m) W_N^{2mk}}_{\text{genap}} + \underbrace{\sum_{m=0}^{(N/2)-1} x(2m+1) W_N^k W_N^{2mk}}_{\text{ganjil}}, \text{for } k = 0, 1, \cdots, N-1$$

- because

$$W_N^2 = W_{N/2}$$

$$W_N^2 = e^{-j2\pi \cdot 2/N} = e^{\frac{-j2\pi \cancel{2}/\cancel{2}}{N/2}} = e^{-j2\pi/N/2} = W_{N/2}$$

- then

$$X(k) = \sum_{m=0}^{(N/2)-1} x(2m) W_{N/2}^{mk} + W_N^k \sum_{m=0}^{(N/2)-1} x(2m+1) W_{N/2}^{mk}, \text{for } k = 0, 1, \cdots, N-1$$

Decimation-in-time algorithm

- equation:

$$X(k) = \sum_{m=0}^{(N/2)-1} x(2m) W_{N/2}^{mk} + W_N^k \sum_{m=0}^{(N/2)-1} x(2m+1) W_{N/2}^{mk}, \text{ for } k = 0, 1, \cdots, N-1$$

- define as new function:

$$G(k) = \sum_{m=0}^{(N/2)-1} x(2m) W_{N/2}^{mk} = \text{DFT}\{x(2m) \text{with} (N/2) \text{points}\},$$

$$H(k) = \sum_{m=0}^{(N/2)-1} x(2m+1) W_{N/2}^{mk} = \text{DFT}\{x(2m+1) \text{with} (N/2) \text{points}\}.$$

Decimation-in-time algorithm

- note that:

$$G(k) = G\left(k + \frac{N}{2}\right), \quad \text{for} \quad k = 0, 1, \cdots, \frac{N}{2} - 1,$$

$$H(k) = H\left(k + \frac{N}{2}\right), \quad \text{for} \quad k = 0, 1, \cdots, \frac{N}{2} - 1.$$

- then:

$$X(k) = G(k) + W_N^k H(k), \quad \text{for} \quad k = 0, 1, \cdots, \frac{N}{2} - 1$$

Decimation-in-time algorithm

- note that:

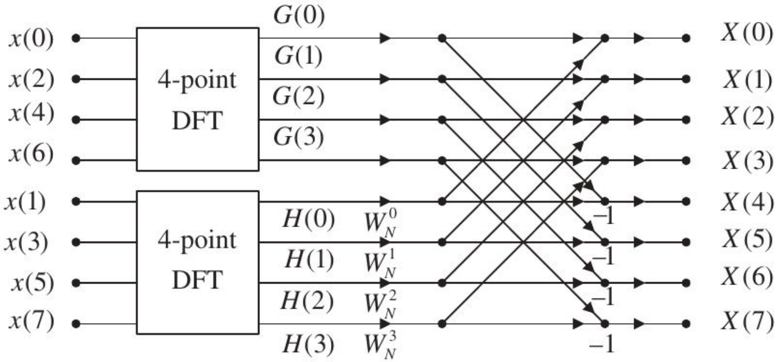
$$W_N^{(N/2+k)} = -W_N^k$$

- then:

$$X\left(\frac{N}{2}+k\right) = G(k) - W_N^k H(k), \text{ for } k = 0, 1, \cdots, \frac{N}{2} - 1$$

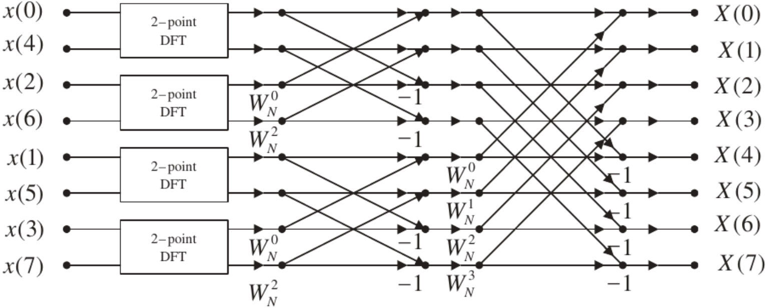
Decimation-in-time algorithm

- the block diagram for the eight-point FFT algorithm (First iteration)



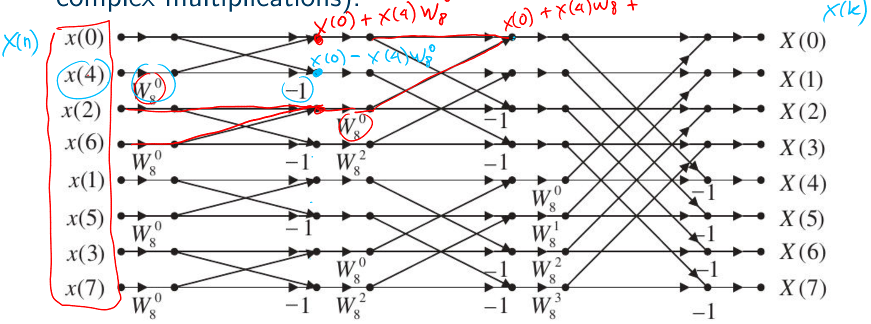
Decimation-in-time algorithm

- the block diagram for the eight-point FFT algorithm (Second iteration)



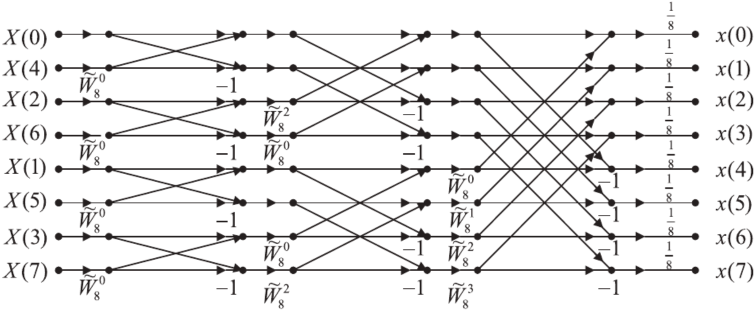
Decimation-in-time algorithm

- Eight-point FFT algorithm using decimation-in-time (12 complex multiplications).



Decimation-in-time algorithm

- The eight-point IFFT using decimation-in-time.



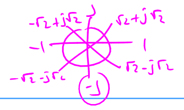
Example 2

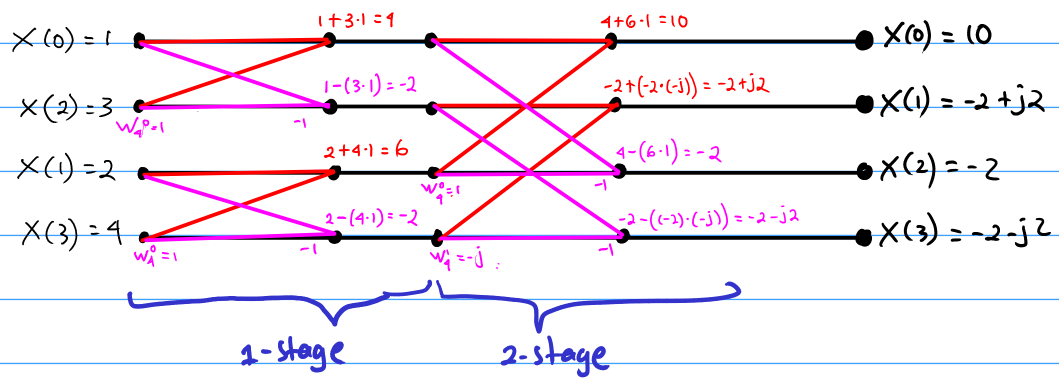
Given a sequence $x(n)$ for $0 \leq n \leq 3$, where $x(0) = 1$, $x(1) = 2$, $x(2) = 3$, and $x(3) = 4$, evaluate its DFT $X(k)$ using the decimation-in-time FFT method.

DIT-FFT

$x(n) = [\underset{\uparrow}{1}, 2, 3, 4]$

index bits	reversal bits
0 → 00	00 → 0
1 → 01	10 → 2
2 → 10	01 → 1
3 → 11	11 → 3





$X(k)=[10\,;\,-2+j2\,;\,-2\,;\,-2-j2]$

$x(n)=[\underset{\uparrow}{1},2,3,4,5]$; Berapa $X(k)$ dari DIT-FFT ?

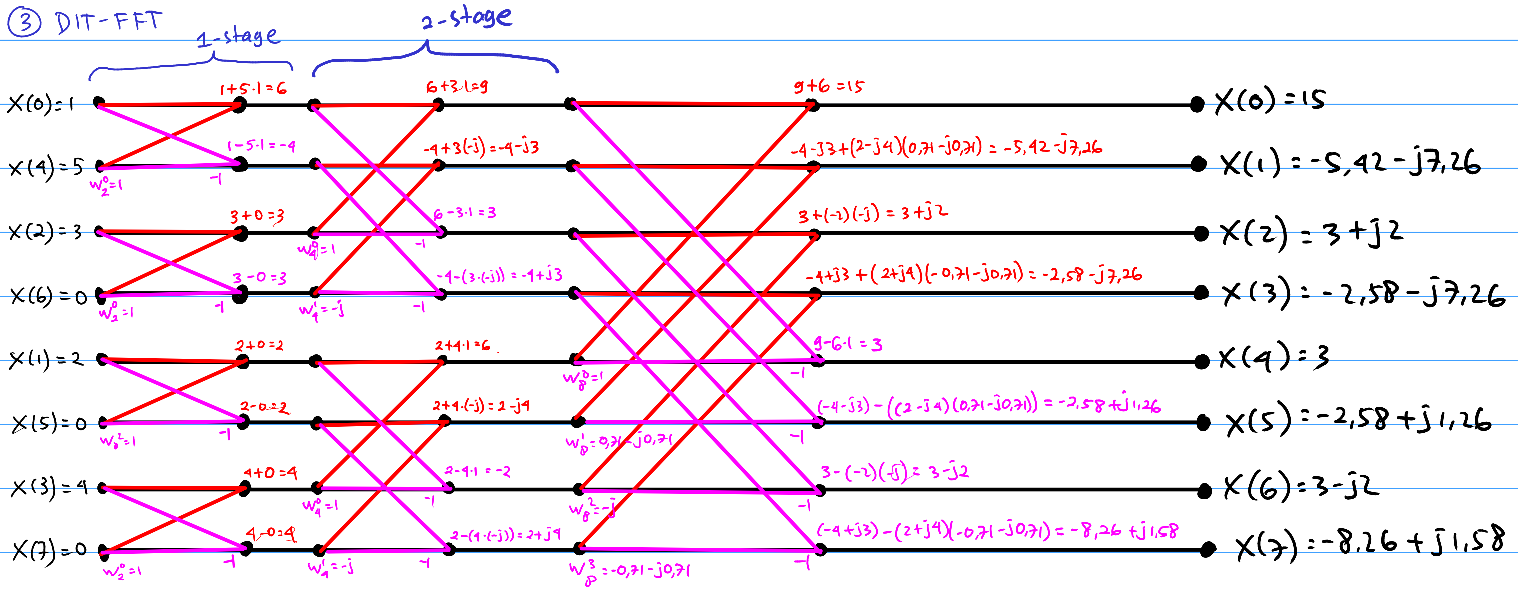
① $N=5 \longrightarrow$ Zero-padding agar $N=8$

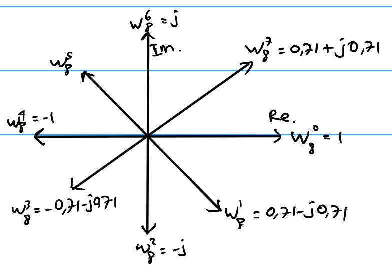
$x(n)=[\underset{\uparrow}{1},2,3,4,5,0,0,0] \rightarrow N=8$

②

index bits	reversal bits
0 → 000	000 → 0
1 → 001	100 → 4
2 → 010	010 → 2
3 → 011	110 → 6
4 → 100	001 → 1
5 → 101	101 → 5
6 → 110	011 → 3
7 → 111	111 → 7

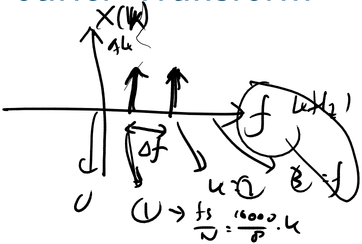
③ DIT-FFT





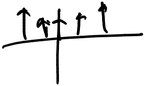
$X(k)=[\underset{\uparrow}{15}\,;\,-5{,}42-j7{,}26\,;\,3+j2\,;\,-2{,}58-j7{,}26\,;\,3\,;\,-2{,}58+j1{,}26\,;\,3-j2\,;\,-8{,}26+j1{,}58]$

Fast Fourier Transform



- frequency resolution, $\Delta f = \frac{f_s}{N}$
- frequency bin, $f = \frac{k f_s}{N}$
- amplitude spectrum,

$$A_k = \frac{1}{N}|X(k)| = \frac{1}{N}\sqrt{(\mathrm{Real}[X(k)])^2 + (\mathrm{Imag}[X(k)])^2},$$
$$k = 0, 1, 2, \cdots, N-1$$



for one-sided amplitude spectrum

$$\bar{A}_k = \begin{cases} \frac{1}{N}|X(0)|, & k = 0 \\ \frac{2}{N}|X(0)|, & k = 1, 2, \cdots, N/2 \end{cases}$$

Fast Fourier Transform

- phase spectrum,

$$\varphi_k = \tan^{-1}\left(\frac{\mathsf{Imag}[X(k)]}{\mathsf{Real}[X(k)]}\right), \; k = 0, 1, 2, \cdots, N-1$$

- power spectrum,

$$P_k = \frac{1}{N^2}|X(k)|^2 = \frac{1}{N^2}\sqrt{(\mathsf{Real}[X(k)])^2 + (\mathsf{Imag}[X(k)])^2},$$
$$k = 0, 1, 2, \cdots, N-1$$

for one-sided power spectrum

positive integer.

$$P_k = \begin{cases} \frac{1}{N^2}|X(0)|, & k = 0 \\ \frac{2}{N^2}|X(0)|, & k = 1, 2, \cdots, N/2 \end{cases}$$

$$x(t) \xrightarrow{\text{Sampling } f_s = 10\,kHz} x(n)$$

$$x(n) = [1, 2, 3, 4]$$

$$X(k) = [10;\ \underline{-2+j2};\ -2;\ -2-j2]$$

① $\Delta f = \frac{f_s}{N} = \frac{10\,kHz}{4} = 2{,}5\ kHz$

② ~~frekuensi~~ ~~bin~~? $f = k.\Delta f$; $k=0 \rightarrow f = 0.2{,}5\ kHz = 0\ kHz$

$k=1 \rightarrow f = 1.2{,}5\ kHz = 2{,}5\ kHz$

$k=2 \rightarrow f = 2.2{,}5\ kHz = 5\ kHz$

$k=3 \rightarrow f = 3.2{,}5\ kHz = 7{,}5\ kHz$

③ Amplitude Spectrum (Ak)

$X(k) = [10;\ -2+j2;\ -2;\ -2-j2]$

$A_k = \frac{1}{N} |X(k)| = \frac{1}{N} \cdot \sqrt{Re_k^2 + Im_k^2}$

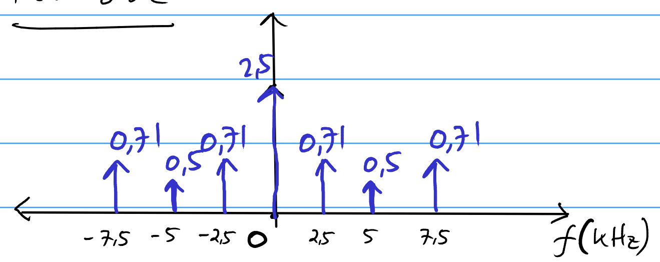
$k=0 \rightarrow A_0 = \frac{1}{4} \cdot \sqrt{10^2 + 0^2} = \frac{1}{4} \cdot 10 = 2{,}5$

$k=1 \rightarrow A_1 = \frac{1}{4} \sqrt{(-2)^2 + 2^2} = \frac{1}{4}\sqrt{8} = \frac{1}{2}\sqrt{2} = 0{,}71$

$k=2 \rightarrow A_2 = \frac{1}{4}\sqrt{(-2)^2 + 0^2} = \frac{1}{4} \cdot 2 = \frac{1}{2} = 0{,}5$

$k=3 \rightarrow A_3 = \frac{1}{4}\sqrt{(-2)^2 + (-2)^2} = \frac{1}{4}\sqrt{8} = \frac{1}{2}\sqrt{2} = 0{,}71$

two - side



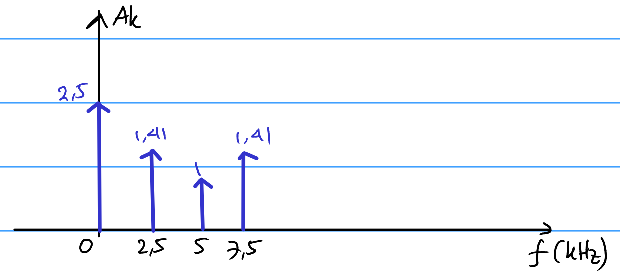
Two - side Amplitude Spectrum :

$A_k = [2{,}5;\ \frac{1}{2}\sqrt{2};\ \frac{1}{2};\ \frac{1}{2}\sqrt{2}]$

One-sided amplitude spectrum

$A_k = [2{,}5;\ \sqrt{2};\ 1;\ \sqrt{2}] = [2{,}5;\ 1{,}41;\ 1;\ 1{,}41]$

one-side Amplitud Spectrum



④ Phase Spectrum $\Phi_k = \tan^{-1}\left(\frac{Im(k)}{Re(k)}\right)$

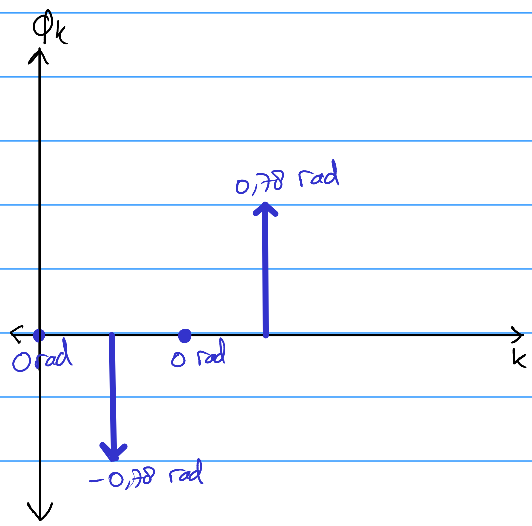
$X(k) = [10;\ -2+j2;\ -2;\ -2-j2]$

$k=0 \rightarrow \Phi_0 = \tan^{-1}\left(\frac{0}{10}\right) = 0\ rad$

$k=1 \rightarrow \Phi_1 = \tan^{-1}\left(\frac{2}{-2}\right) = -0{,}78\ rad$

$k=2 \rightarrow \Phi_2 = \tan^{-1}\left(\frac{0}{-2}\right) = 0\ rad$

$k=3 \rightarrow \Phi_3 = \tan^{-1}\left(\frac{-2}{-2}\right) = 0{,}78\ rad$



⑤ Power Spectrum $P_k = \frac{1}{N^2}|X(k)|^2$

$$= \left(\frac{1}{N}|X(k)|\right)^2$$

$$P_k = (A_k)^2$$

Two-side Power Spectrum :

$$P_k = \left\{\left[2{,}5\,;\ \tfrac{1}{2}\sqrt{2}\,;\ \tfrac{1}{2}\,;\ \tfrac{1}{2}\sqrt{2}\right]\right\}^2$$

$$= [6{,}25\,;\ 0{,}5\,;\ 0{,}25\,;\ 0{,}5]$$

One-side Power Spectrum :

$$P_k = [6{,}25\,;\ 1\,;\ 0{,}5\,;\ 1]$$

two-sided power spectrum