P. Rajesh M.Tech.,
JNTUACEA

Advanced Digital Signal Processing.

1) **Signal:-** The Physical quantity which varies with time space (or) the more independent variables.

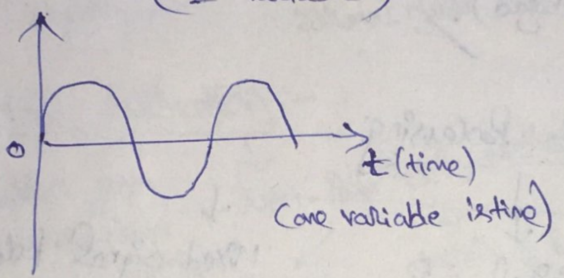
$$x(t) = F(x_1, x_2 \dots x_n).$$

x_i = time, space, temp, etc., → (independent variables)

Eg:- Audio, video, ECG (electro cardio gram), AC power supply ...etc.,

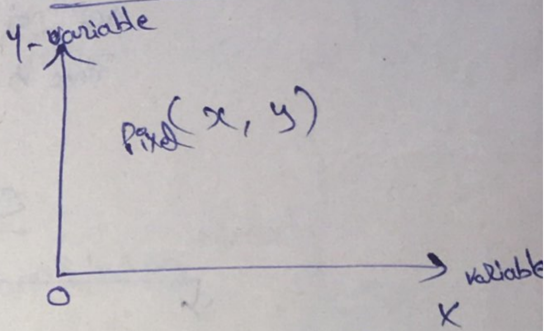
1-Dimensional signal

(1-variable)



Eg:- voice (or) AC signal.

2-Dimensional signal



(2-variables)

Eg:- Picture, video signal.

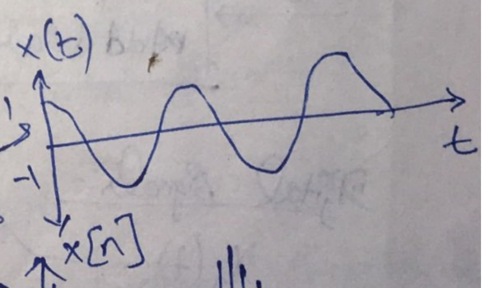
Multi Dimensional signal:-

→ It depends one 1 (or) more variables

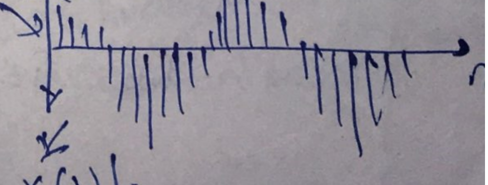
Eg:- speed of winds.

2) **Classifications of signals:-**

ⓐ ⟶ Continous time signals



ⓑ ⟶ Discrete " "



ⓒ ⟶ Digital " "

$x(n) = 0$ (or) 1

$-\infty$ to ∞

$$x(nT) = x(t)\big|_{t=nT}$$

n - integer ranging from $-\infty$ to ∞ called time index

T = sampling interval

(a) ~~Continous time Signals~~ / ~~Analogs~~ Signal..

Signal
- Analog
  - $\boxed{\text{Continous Time}}$
  - $\boxed{\text{Discrete Time}}$ → $\boxed{A/D}$ converter → Digital

(is not Digital signal in which Amplitude is Continum and not discretised and Time is discretized).

- Digital

<u>Signal Processing:-</u>

- Analog S.P
- Digital S.P
- Mixed Signal Processing

<u>Block diagram:-</u>

Analog i/p → $\boxed{\text{sample and hold devices}}$ → $\boxed{\text{ADC}}$ → $\boxed{\text{Digital Processing}}$ → $\boxed{\text{DAC}}$ → $\boxed{\text{Analog Low Pass filter}}$ → Analog o/p

<u>Digital Signals:-</u>

$x_a(t)$ — switch → $x_a(nT) \rightarrow \boxed{A/D} \rightarrow x(n)$

(a & T → lost at digital final o/p)

where $T = \frac{1}{f_s}$ sequence $x(n)$; $\{x(n)\}$

20

3

Frame out

1. Sanjit K Mitra
2. J G Proakis

Advanced Digital Signal Processing.

→ Learn about $\boxed{\text{DSP algorithms}}$ → split Radix FFT, DFT

→ Computation over narrow freq Band

→ fast DFT based on index mapping

→ slide Discrete F.T

→ Linear filtering approach to computation of DFT chirp Z-Transform.

} unit - 1

→ Multi Rate Signal Processing:- → mechanism of converting signal from given rate to different rate is known sample rate conversion

→ filter design & Implementation for sampling rate conversion.

→ Decimation by a factor 'D'

→ Interpolation by factor I

→ sampling rate conversion by rational factor I/D

(Time)

} unit - 2

→ Power spectral Estimation:-

→ spectra estimation from finite duration observation of signals (Energy density spectrum)

→ Non-Parametric methods → Bartlett, welch & Blackmann & Tukey method.

→ Parametric methods for Power Spectrum Estimation:-

→ Relation b/w auto Correlation & model parameters

→ Yule-walker & Burg methods

→ MA & ARMA model for Power Spectrum estimation

} unit - 3

→ Analysis of finite wordlength Effects in fixed Point DSP system:-

→ Fixed, floating Point Arithematic – ADC quantization noise & signal quality – Finite wordlength effect in IIR digital filters.

→ Finite wordlength effect in FFT algorithms

} unit - 4

→ Application of Digital signal Processing:- → over sampling A/D converter & D/A converter

→ musical sound Processing

→ Dual Tone multi freq signal Detection

→ spectral Analysis of sinusoidal signals

→ spectral Analysis of non stationary signals

→ Discrete Time Analytic signal Generation.

$\{1, 2, 3, 4\}$ then
↑
$n=0$

where $n=0 \rightarrow N-1$

Defines finite length of sequences ← $\boxed{DFT}$ (Powerful Computation tool) in Digital Computers / special designed hardware Applications

↓

allows to Evaluate fourier transforms $X(e^{jw})$

↓

continuous & Periodic

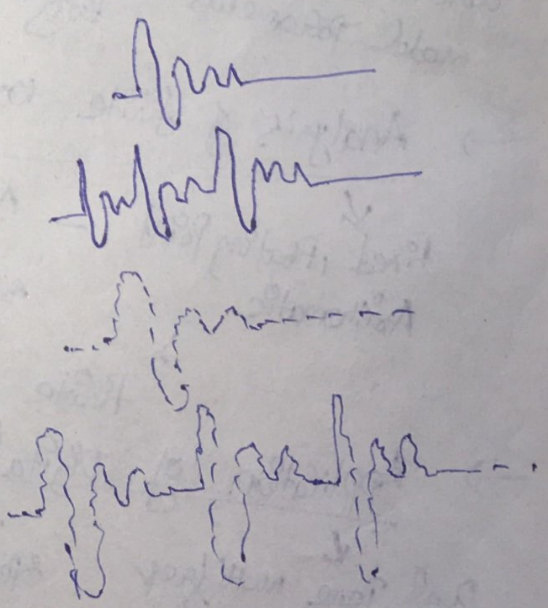
→ DFT obtained by sampling 1-Period of fourier transform @ finite no. of frequency Points.

Decimation mean:- A ~~dramatic~~ drastic reduction in strength of effectiveness of something

interpolation:- the insertion of an intermediate value (or) term into a series by estimating (or) calculating it from surrounding known values.

Signal form Representation

Example signal



(a) fourier Transforms
(signal that are continuous & aperiodic)

(b) fourier series.
(signal that are continuous & Periodic)

(c) Discrete Time fourier Transform (DTFT)
(signal that are discrete & aperiodic)

(d) Discrete fourier Transforms (DFT)
(signal that are discrete & Periodic)

Scanned with CamScanner

Unit - 1 Review 4

Discrete Fourier Transform:-

-> It's a specific kind of Discrete Transform used in Fourier Analysis
-> It transform one func in to another which is freq domain representation. (or) simply DFT.

-> DFT requires i/p is Discrete function, Such i/p are often Created by sampling a continous function such as periodic voice.

-> DFT depends upon the length 'N' - no of sequences that power 2 integers

① **Fast DFT algorithm:-**

If length N is not a power of 2 it is modified by zero padding (reach @ desired length)

Cooley - Tukey FFT algorithm   Prime factor Algorithm

② **Slide Discrete F.T**

Applications involving a very long sequence, it is of interest to determine the spectral properties of subset consisting of fixed no:- of consecutive samples at successive values of time instant 'n'.

③ **chirp Transform:-**

used to Compute a limited no:- of samples of z-Transform of a finite length sequence at point Equally placed in angle over a portion of spiral contour in Z-plane.

③ **DFT Computation over a narrow freq Band:-**

DFT samples over a specified freq range

ⓐ Zoom FFT   ⓑ chirp Transform

used to Compute samples of an N-point DFT X[k] of length, N.- sequence x[n] in small range of values of frequency index.

1). Discrete Fourier Transforms:-

-> Is a Powerful Computation tools allows to Evaluate Fourier transform.

-> It is defined only for sequence of length finite.

where $X(e^{j\omega})$ is Continous & Periodic.

-> DFT is obtained by sampling one Period of Fourier Transform @ a finite no:- of frequency Points, where DFT Plays important role in signal Processing algorithms.

(a) Discrete Fourier Transform (DFT) of Discrete Time Signal:-

-> DTFT of Sequence is Periodic and frequency ranges $0 \rightarrow 2\pi$ and there are infinitly ω in this range.

-> If we use digital computer to Compute N - Equal spaced points over the interval $0 < \omega < 2\pi$ then. 'N' Points should be located at

$$\omega_K = \frac{2\pi}{N} k \; ; \text{ where } K = 0, 1, 2 \ldots N-1$$

$N \rightarrow$ is Equally spaced freq samples of DTFT are known as DFT denoted as $X(K)$.

$$X(K) = X(e^{j\omega}) \Big|_{\omega = \frac{2\pi}{N}.k} \text{ for } K = 0, 1, 2 \ldots N-1$$

where DFT sequence $K = 0$ starts and corresponding to $K = N$.

The no:- of samples for finite duration sequence $x(n)$ of length 'L' should be $N \geq L$ in order aliasing freq spectrum.

$x(n)$ = Discrete Time signal of length 'L'

$X[K] = DFT[x(n)]$

now n-point DFT of $x(n)$ where $N \geq L$ is defined as

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

inverse DFT as

$$x(n) = \frac{1}{N} \sum_{K=0}^{N-1} X(K) \cdot e^{j\frac{2\pi}{N}.k.n} \; ; \; n = 0, 1, 2 \ldots N-1$$

$$x(n) = \frac{1}{N} \sum_{K=0}^{N-1} X(K) \cdot W_N^{-kn} \quad \text{where } W_N = e^{-j\frac{2\pi}{N}}$$

$$\Rightarrow X[K] = \sum_{n=0}^{N-1} x(n) \cdot W_N^{kn} \; //.$$

Scanned with CamScanner

transform

find DFT of sequence $x(n) = \{1, 1, 0, 0\}$ & IDFT of $Y(k) = \{1, 0, 1, 0\}$.

$x(0)\ x(1)\ x(2)\ x(3) \rightarrow x(n)$.

Sol:- $DFT[x(n)] = X(k)$

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

$k = 0, 1, 2, \dots, N-1$ (i.e. $k = 3$)
where $N = 4$ (no of Points)

for $k = 0$

$$X(0) = \sum_{n=0}^{3} x(n) \cdot e^{-j\frac{2\pi}{4}\cdot(0)\cdot n}$$

$$= \sum_{n=0}^{3} x(n) \cdot e^{0} = \sum_{n=0}^{3} x(n)$$

$\therefore$ where $x(n) = x(0)$; $x(n) = x(1)$; $x(n) = x(2)$; $x(n) = x(3)$

then $X(0) = 1 + 1 + 0 + 0 = 2$ //.

for $k = 1$

$$X(1) = \sum_{n=0}^{3} x(n) \cdot e^{-j\frac{2\pi}{4_2}\cdot n\cdot 1}$$

$$= x(0) + x(1) \cdot e^{-j\frac{\pi}{2}} + x(2) \cdot e^{-j\frac{\pi}{2}(2)(1)} + x(3) \cdot e^{-j\frac{\pi}{2}(3)\cdot(1)}$$

$$= 1 + 1 \cdot e^{-j\frac{\pi}{2}} + 0 + 0$$

$$= 1 + \left[\cos\frac{\pi}{2} - j\sin\frac{\pi}{2}\right] + 0 + 0 = 1 + [0 - j] + 0 + 0 = 1 - j.$$

for $k = 2$.

$$X(2) = \sum_{n=0}^{3} x(n) \cdot e^{-j\frac{2\pi}{4_2}\cdot n\cdot 2}$$

$$= x(0) + x(1) \cdot e^{-j\pi(1)} + x(2) \cdot e^{-j\pi(2)} + x(3) \cdot e^{-j\pi(3)}$$

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

$$= 1 - 1 = 0.$$

for $k = 3$

$$X(3) = \sum_{n=0}^{3} x(n) \cdot e^{-j\frac{2\pi}{4_2}\cdot n\cdot 3}$$

$$= x(0) + x(1) \cdot e^{-j\frac{3\pi}{2}\cdot(1)} + x(2) \cdot e^{-j\frac{3\pi}{2}(2)} + x(3) \cdot e^{-j\frac{3\pi}{2}(3)}$$

$$= 1 + 1\cdot\left\{\cos\left(\frac{3\pi}{2}\right) - j\sin\left(\frac{3\pi}{2}\right)\right\} + 0 + 0 = 1 + 1\{0 - j(-1)\}$$

$$= 1 + j \text{ //}.$$

$$\therefore X\{k\} = \{2, 1-j, 0, 1+j\}. \text{ //}.$$

IDFT :- $y[k] = \{1, 0, 1, 0\}$

WKT $y(n) = \frac{1}{N} \sum_{n=0}^{N-1} y(k) \cdot e^{j\frac{2\pi}{N} \cdot kn} \Big|_{n=0,1,2 \dots N-1}$

for $n = 0$ (ie $n = 0, 1, 2, 3$)

$$y(0) = \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{2\pi}{4} \cdot k \cdot (0)}$$

$$= \frac{1}{4} \sum_{n=0}^{3} y(k) \Rightarrow \frac{1}{4}\left\{1 + 0 + 1 \cdot e^{j\frac{2\pi}{4} \cdot (3) \cdot 0} + 0\right\}$$

$$= \frac{1}{4}[1+1] = \frac{2}{4} = 0.5$$

$$y(1) = \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{2\pi}{4} \cdot k \cdot (1)}$$

$$= \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{\pi}{2}k}$$

now :- $\frac{1}{4}\left[1 \cdot e^{j\frac{\pi}{2}(0)} + 0 + 1 \cdot e^{j\frac{\pi}{2} \cdot 2} + 0\right]$

$$= \frac{1}{4}\left[1 + 0 + (\cos(\pi) + j\sin\pi) + 0\right]$$

$$= \frac{1}{4}[1 + 0 - 1 + 0] = 0$$

$$y(2) = \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{2\pi}{4} \cdot k \cdot (2)}$$

$$= \frac{1}{4}\left[1 \cdot e^{j\pi(0)} + 0 + 1 \cdot e^{j2\pi} + 0\right]$$

$$= [1 + 0 + (\cos 2\pi + j\sin 2\pi) + 0] = \frac{1}{4}\{1 + 0 + 1 + 0\} = \frac{2}{4} = 0.5$$

$$y(3) = \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{2\pi}{4} \cdot k \cdot (3)}$$

$$= \frac{1}{4} \sum_{n=0}^{3} y(k) \cdot e^{j\frac{3\pi}{2}k}$$

$$= \frac{1}{4}\left[1 + 0 + 1 \cdot e^{j\frac{3\pi}{2} \cdot (2)} + 0\right]$$

$$= \frac{1}{4}\{1 + 0 + (\cos 3\pi + j\sin 3\pi) + 0\}$$

$$= \frac{1}{4}\{1 + 0 - 1 + 0\} = 0$$

$\therefore y(n) = \{0.5, 0, 0.5, 0\}$

Fast DFT algorithm Based on index mapping :-

-> Previous " " are for sequence of N-length that are power of 2 integer.

-> If N length is not a power of 2 integer it can be modified by zero padding.
   ↳ To arrive at modified sequence of length that is power of 2-integer.

-> i.e zero padding also increases DFT length as a result of computational complexity

-> For case when length of 'N' is a composite no:- that is expressible as product of integers.
and so it possible to develop computational fast DFT algorithm via index mapping approach in 'n' and 'k' are mapped in 2-D indices.

(a) General form of Cooley-Tukey FFT algorithm :-

Let consider sequence $x[n]$ of length N that product of two integers N_1 & N_2

$$N = N_1 \times N_2 \rightarrow (1)$$

Time domain index 'n' can be represented as function of two indices n_1 & n_2

$$n = n_1 + n_2 N_1 \quad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases} \rightarrow (2)$$

Freq domain index k of DFT sequence $X[k]$ can be represented as function of indeces k_1 & k_2

$$k = k_1 N_2 + k_2 \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases} \rightarrow (3)$$

from eq (2) index 'n' ranges from $0 \rightarrow N-1$ and represented n_1 & n_2 and the index mapping effectively maps 1-D sequence $x[n]$ of length N in 2-D sequence $x[N_1\ N_2]$ of size $N_1 \times N_2$ (contains N_1-rows & N_2-columns).

Similarly:- from eq (3) index 'K' ranges $0 \rightarrow N-1$ and represented uniquely as k_1 & k_2 and index mapping effectively maps (2-D) of length 'N' $X[k]$ sequency as $X[k_1, k_2]$ of size $N_1 \times N_2$ in 2-D sequence.

∴ DFT samples are as

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot W_N^{nk} \quad ; \quad 0 \le k \le N-1 \rightarrow (4)$$

as

$$X[k] = X[N_2 k_1 + k_2] = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x[n_1 + n_2 N_1] \, W_N^{[n_1 + N_1 n_2][N_2 k_1 + k_2]} \rightarrow (5)$$

$$X[k] = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x[n_1 + N_1 n_2]\, W_N^{n_1 N_2 k_1} \cdot W_N^{n_1 k_2} \cdot W_N^{N_1 n_2 k_1 N_2} \cdot W_N^{N_1 n_2 k_2} \rightarrow (6)$$

where $0 \le k_1 \le N_1$ & $0 \le k_2 \le N_2$

Now:-

$$W_N^{n_1 N_2 k_1} = W_{N_1 N_2}^{n_1 N_2 k_1} = W_{N_1}^{n_1 k_1} \rightarrow (7)$$

$$W_N^{N_1 n_2 k_1 N_2} = W_{N_1 N_2}^{N_1 n_2 k_1 N_2} = 1 \rightarrow (8)$$

$$W_N^{N_1 n_2 k_2} = W_{N_1 N_2}^{N_1 n_2 k_2} = W_{N_2}^{n_2 k_2} \rightarrow (9)$$

Now:- Eq (6) Can be written as

$$X\{N_2 k_1 + k_2\} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x[n_1 + n_2 N_1]\, W_{N_1}^{n_1 k_1} W_N^{n_1 k_2} \cdot W_{N_2}^{n_2 k_2} \rightarrow (10)$$

$$= \sum_{n_1=0}^{N_1-1} \left[\left[\sum_{n_2=0}^{N_2-1} x(n_1 + n_2 N_1) \cdot W_{N_2}^{n_2 k_2}\right] \cdot W_N^{n_1 k_2}\right] \cdot W_{N_1}^{n_1 k_1}, \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases} \rightarrow (11)$$

let define General form.

$$G[n_1, k_2] = \sum_{n_2=0}^{N_2-1} x(n_1 + n_2 N_1) \cdot W_{N_2}^{n_2 k_2}, \quad 0 \le k_2 \le N_2 - 1 \rightarrow (12)$$

for each value of index n_1 the quantity $G[n_1, k_2]$ Can be considered as an Point DFT of length N_2 — sequences.
→ Row numbered of 2-D array defined by $x[n_1 + N_1 n_2]$.

by substituting Eq (12) in Eq (11) then

$$X\{N_2 k_1 + k_2\} = \sum_{n_1=0}^{N_1-1} \left(G[n_1, k_2]\, W_N^{n_1 k_2} \right) \cdot W_{N_1}^{n_1 k_1} \rightarrow (13)$$

$$= \sum_{n_1=0}^{N_1-1} \hat{G}[n_1, k_2] \cdot W_{N_1}^{n_1 k_1} \rightarrow (14)$$

where. $\hat{G}\{n_1, k_2\} = G[n_1, k_2] \cdot W_N^{n_1 k_2} \rightarrow (15)$

from Eq (15) the set of N_1 - sequences $\hat{G}\{n_1, k_2\}$ of length N_2 Each Can be obtained multipling the row DFT $G[n_1, k_2]$ by the twiddle factor $W_N^{n_1 k_2}$

Similarly:- From Eq (14) defines an N_1 point DFT of length N_1 - sequence in Column No:- k_2 of 2D-array defined $\hat{G}[n_1, k_2]$ 11.

(10).

For $N = 15$ Compute DFT using Cooley-Tukey algorithm

Choose $N_1 = 3$, $N_2 = 5$

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

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

$$\Rightarrow n = n_1 + 3n_2 \begin{cases} 0 \le n_1 \le 2 \\ 0 \le n_2 \le 4 \end{cases} \quad \Big| \quad k = k_2 + 5k_1 \begin{cases} 0 \le k_1 \le 2 \\ 0 \le k_2 \le 4 \end{cases}.$$

$x[k_2 + N_2 k_1]$

$$\therefore x\{k_2 + 5k_1\} = \sum_{n_1=0}^{2} \sum_{n_2=0}^{4} x(n_1 + 3n_2) \cdot W_5^{n_2 k_2} W_{15}^{n_1 k_2} \cdot W_3^{n_1 k_1} \quad \begin{cases} 0 \le k_1 \le 2 \\ 0 \le k_2 \le 4 \end{cases}$$

$\therefore$ i/p index mapping results in 2-D representation.

$n_1 \backslash n_2$	0	1	2	3	4
0	$x(0)$	$x(3)$	$x(6)$	$x(9)$	$x(12)$
1	$x(1)$	$x(4)$	$x(7)$	$x(10)$	$x(13)$
2	$x(2)$	$x(5)$	$x(8)$	$x(11)$	$x(14)$

when $N_1 \times N_2 = 3 \times 5 = 15$,
$N_1 = 0 \to 2$
$N_2 = 0 \to 4$
$\Rightarrow (n = n_1 + N_1 n_2)$

above results 2D array as

$\rightarrow$ the 5-point DFT of each one of 3-rows given

$G[n_1, k_2]$

$n_1 \backslash k_2$	0	1	2	3	4.
0	$G(0,0)$	$G(0,1)$	$G(0,2)$	$G(0,3)$	$G(0,4)$
1	$G(1,0)$	$G(1,1)$	$G(1,2)$	$G(1,3)$	$G(1,4)$
2	$G(2,0)$	$G(2,1)$	$G(2,2)$	$G(2,3)$	$G(2,4)$

These above $G[n_1, k_2]$ DFT are multiplied by twiddle factor $W_{15}^{n_1 k_2}$ leading 2D array

$n_1 \backslash k_2$	0	1	2	3	4
0	$G(0,0)$	$G(0,1)$	$G(0,2)$	$G(0,3)$	$G(0,4)$
1	$G(1,0)$	$G(1,1)$	$G(1,2)$	$G(1,3)$	$G(1,4)$
2	$G(2,0)$	$G(2,1)$	$G(2,2)$	$G(2,3)$	$G(2,4)$

$\therefore$ 3-Point DFT of each column of array $G[n_1, k_2]$ are carried out leading desired 15-Point DFT $X[k]$

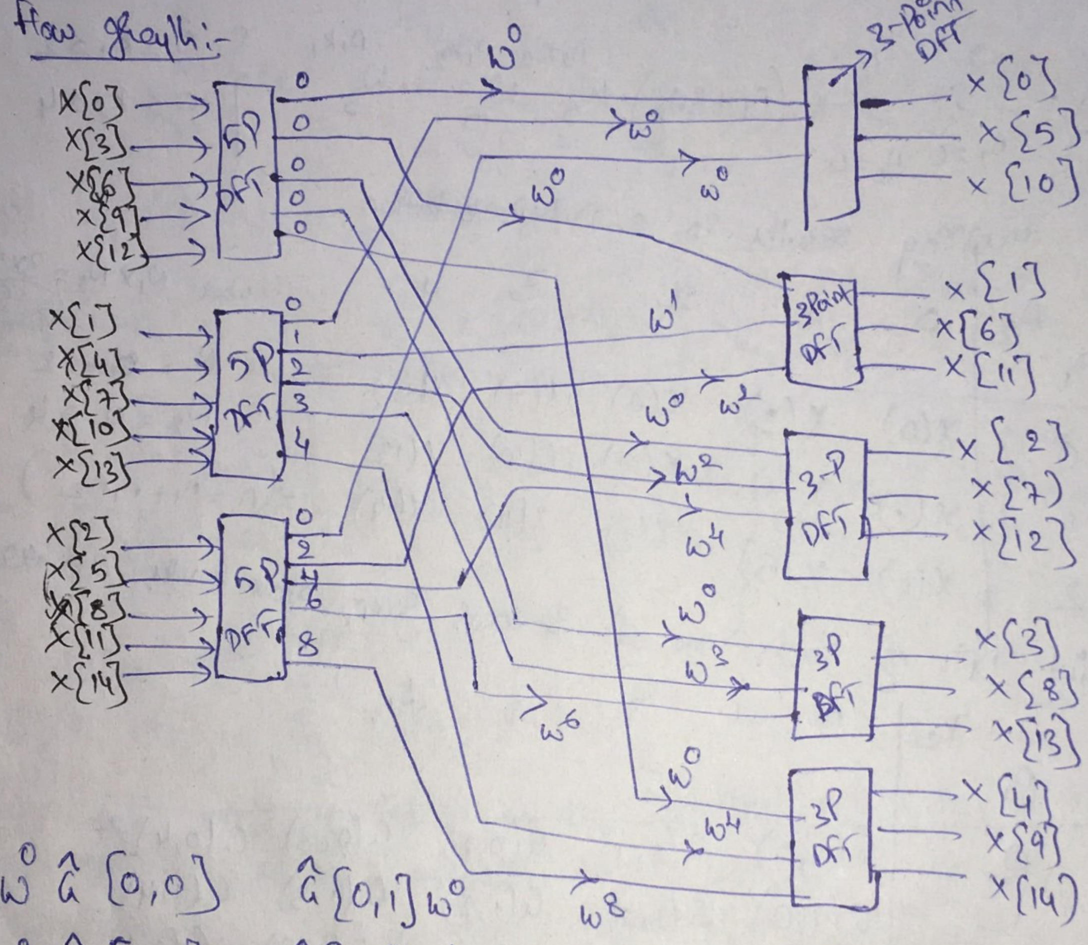
Scanned with CamScanner

k_1 \ k_2	0	1	2	3	4
0	x{0}	x{1}	x{2}	x{3}	x{4}
1	x{5}	x{6}	x{7}	x{8}	x{9}
2	x{10}	x{11}	x{12}	x{13}	x{14}

(ie $k = N_2 \cdot k_1 + k_2$)
$N_2 = 5$

Flow graph representation of above 15 point FFT algorithm is

Flow graph:-



$W^0 \hat{G}[0,0]$ $\quad \hat{G}[0,1] W^0$

$W^0 \hat{G}[1,0]$ $\quad \hat{G}[0,1] W^1$

$W^0 \hat{G}[2,0]$ $\quad \hat{G}[2,1] W^2$

for eq:- if $N_1 = 2$; $N_2 = N/2$ the index mapping of eq (1) & eq (2) lead to 1st stage of decimation in Time FFT algorithm.

where if $N_1 = \frac{N}{2}$ & $N_2 = 2$ the index mapping of eq (1) & (2) lead to 1st stage desimation in freq FFT algorithm

$$n = N_2 n_1 + n_2 \qquad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases} \longrightarrow (16)$$

$$k = k_1 + N_1 k_2 \qquad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases} \longrightarrow (17)$$

12

(b) Prime factor algorithms:-

Efficient DFT algorithm can be developed if the multiplication by the twiddle factors are eliminated by choosing at appropriately the index mapping

$$n = \langle An_1 + Bn_2 \rangle_N \quad , \quad \begin{cases} 0 \le n_1 \le N_1 - 1 \\ 0 \le n_2 \le N_2 - 1 \end{cases} \rightarrow (18)$$

$$k = \langle Ck_1 + Dk_2 \rangle_N \quad , \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases} \rightarrow (19)$$

where $\langle l \rangle_N$ - denotes modulo N

for $A = D = 1$; $B = N_1$ & $C = N_2$ the above index mappings reduce to the mapping give in Eq (2) & (3)

similly for $A = N_2$; $B = C = 1$ & $D = N_1$ for either of these choices the values of indices n and k remain within range $(0, N-1)$ as a result modulo operation is not needed.

using above index mapping, DFT sample of Eq (24).

$$X[k] = X[\langle Ck_1 + Dk_2 \rangle_N]$$

$$= \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x\{\langle An_1 + Bn_2 \rangle_N\} W_N^{[\langle An_1+Bn_2\rangle_N][\langle Ck_1+Dk_2\rangle_N]} \quad , \quad \begin{cases} 0 \le k_1 \le N_1 - 1 \\ 0 \le k_2 \le N_2 - 1 \end{cases} \rightarrow (20)$$

now:-

$$W_N^{[\langle An_1+Bn_2\rangle_N][\langle Ck_1+Dk_2\rangle_N]} = W_N^{\langle ACn_1k_1\rangle_N} \cdot W_N^{\langle ADn_1k_2\rangle_N} \, W_N^{\langle BCn_2k_1\rangle_N} \cdot W_N^{\langle BDn_2k_2\rangle_N} \rightarrow (21)$$

Twiddle factor can be eliminated if N_1 & N_2 are "relatively prime"

$$\therefore \langle AC \rangle_N = N_2 \, , \, \langle BD \rangle_N = N_1 \, , \, \langle AD \rangle_N = \langle BC \rangle_N = 0 \rightarrow (22)$$

then factor on the right hand side of Eq simply to

$$W_N^{\langle ACn_1k_1\rangle_N} = W_{N_1N_2}^{N_2n_1k_1} = W_{N_1}^{n_1k_1} \, ,$$

$$W_N^{\langle ADn_1k_2\rangle_N} = W_N^{0} = 1$$

$$W_N^{\langle BCn_2k_1\rangle_N} = W_N^{0} = 1$$

Scanned with CamScanner

$$W_N^{\langle BD n_2 k_2 \rangle_N} = W_{N_1 N_2}^{N_1 n_2 k_2} = W_{N_2}^{n_2 k_2}$$

from Eq (21) reduces to

$$W_N^{[\langle An_1 + Bn_2 \rangle_N][\langle Ck_1 + Dk_2 \rangle_N]} = W_{N_1}^{k_1 n_1} \cdot W_{N_2}^{k_2 n_2} \rightarrow (23)$$

thus eliminating of twiddle factors.

one possible choice for the constants satisfying is given by from Eq (22)

$$A = N_2 \;;\; B = N_1 \;;\; C = N_2 \langle N_2^{-1} \rangle_{N_1} \;;\; D = N_1 \langle N_1^{-1} \rangle N_2 \rightarrow (24)$$

where $\langle N_a^{-1} \rangle_{N_b}$ denotes the multiplicate inverse of N_a reduced modulo N_b.

if $\langle N_1^{-1} \rangle_{N_2} = \alpha$ then $\langle N_1 \alpha \rangle_{N_2} = 1$ (or) $N_1 \alpha = N_2 \beta + 1$

where β is any integer.

Similly $\langle N_2^{-1} \rangle_{N_1} = \gamma$ then $\langle N_2 \gamma \rangle_{N_1} = 1$ (or) $N_2 \gamma = N_1 \delta + 1$ where δ is any integer

Eg:- Determination of the Multiplicative Inverse of Two Relatively Prime factors:

Let $N_1 = 3$; $N_2 = 5$ then $(3^{-1})_5 = 2$ as $(2 \cdot 3)_5 = 1$ ~~like~~ $\langle 2 \cdot 5 \rangle_3 = 1$

like wise $(5^{-1})_3 = (2 \cdot 5)_3 = 1$. $\boxed{\langle 3^{-1} \rangle_5 = 2 \Rightarrow \frac{1}{3} \times 5 = 2 \Rightarrow \langle 2 \cdot 3 \rangle_5 = 1}$ $\langle 5^{-1} \rangle_3 = 1$ & $2 \cdot 5 \times 3 = 1$, $2 \cdot 5 = \frac{1}{3}$, $\langle 3^{-1} \rangle_5 = 2$

which requires

There are several other choices for constants A, B, C, & D the applications of Chinese Remainder Theorem. from Eq (24)

$$\langle AC \rangle_N = \langle N_2 \cdot (N_2 \langle N_2^{-1} \rangle_{N_1}) \rangle_N = \langle N_2 (N_1 \delta + 1) \rangle_N = \langle N_2 N_1 \delta + N_2 \rangle_N = N_2$$

$$\langle BD \rangle_N = \langle N_1 \cdot (N_1 \langle N_1^{-1} \rangle_{N_2}) \rangle_N = \langle N_1 (N_2 \beta + 1) \rangle_N = \langle N_1 N_2 \beta + N_1 \rangle_N = N_1$$

$$\langle AD \rangle_N = \langle N_2 (N_1 \langle N_1^{-1} \rangle_{N_2}) \rangle_N = \langle N \alpha \rangle_N = 0$$

$$\langle BC \rangle_N = \langle N_1 \cdot (N_2 \langle N_2^{-1} \rangle_{N_1}) \rangle_N = \langle N \gamma \rangle_N = 0$$

from Eq (20) we write

$$x[k] = x[\langle Ck_1 + Dk_2\rangle_N]$$

$$= \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} x\{\langle An_1 + Bn_2\rangle_N\} W_N^{N_2 n_1 k_1} \cdot W_N^{N_1 n_2 k_2}$$

$$= \sum_{n_1=0}^{N_1-1}\left(\sum_{n_2=0}^{N_2-1} x\{\langle An_1 + Bn_2\rangle_N\} \cdot W_{N_2}^{n_2 k_2}\right) W_{N_1}^{n_1 k_1} \quad , \begin{cases} 0 \le k_1 \le N_1-1 \\ 0 \le k_2 \le N_2-1 \end{cases} \rightarrow (25)$$

Substituting

$$G[n_1, k_2] = \sum_{n_2=0}^{N_2-1} x[\langle An_1 + Bn_2\rangle_N] W_{N_2}^{n_2 k_2} \quad , \begin{cases} 0 \le n_1 \le N_1-1 \\ 0 \le k_2 \le N_2-1 \end{cases} \rightarrow (26)$$

in Eq (25) we get

$$x\{k\} = \sum_{n_1=0}^{N_1-1} G\{n_1, k_2\} W_{N_1}^{n_1 k_1} \quad , \begin{cases} 0 \le k_2 \le N_2-1 \\ 0 \le k \le N \end{cases} \rightarrow (27)$$

∴ from Eq (26) it is noted that N_2-Point DFT of n_1^{th} row of 2-D array $x\{\langle An_1 + Bn_2\rangle_N\}$,

Similly Eq (27) it is N_1-Point DFT of k_2^{th} row of 2-D array $G\{n_1, k_2\}$

we can re-write as

$$x\{k\} = \sum_{n_2=0}^{N_2-1}\left(\sum_{n_1=0}^{N_1-1} x\{\langle An_1 + Bn_2\rangle_N\} W_{N_1}^{n_1 k_1}\right) W_{N_2}^{n_2 k_2} \quad , \begin{cases} 0 \le k_1 \le N_1-1 \\ 0 \le k_2 \le N_2-1 \end{cases} \rightarrow (28)$$

∴ The N_1-Point DFT $H\{k_1, n_2\}$ of n_2^{th} colum of 2-D array $x\{\langle An_1 + Bn_2\rangle_N\}$:

$$H\{k_1, n_2\} = \sum_{n_1=0}^{N_1-1} x[\langle An_1 + Bn_2\rangle_N] W_{N_1}^{n_1 k_1} \quad \begin{cases} 0 \le k_1 \le N_1-1 \\ 0 \le n_2 \le N_2-1 \end{cases}$$

∴ use of $H\{k_1, n_2\}$ in Eq (28) we write as

$$x\{k\} = \sum_{n_2=0}^{N_2-1} H\{k_1, n_2\} W_{N_2}^{n_2 k_2} \quad , \begin{cases} 0 \le k_1 \le N_1-1 \\ 0 \le k \le N. \end{cases}$$

Problem:- 15-Point DFT Computation using Prime factor Algorithm

$N = 15$; let $N_1 = 3$ & $N_2 = 5$ which are seen to be relatively Prime

$\langle 3^{-1}\rangle_5 = 2 \Rightarrow \frac{1}{3} \times 5 = 2 \Rightarrow \langle 2 \cdot 3\rangle_5 = 1$

∴ now we write

$$\langle 3^{-1}\rangle_5 = 2 \quad ; \quad \langle 5^{-1}\rangle_3 = 2$$

$\langle 5^{-1}\rangle_3 = 2 \Rightarrow \cancel{\langle 5 \times 3\rangle_5 = 2 \Rightarrow \langle 2 \cdot 3\rangle_5 = 1}$

$N_2 \langle N_2^{-1}\rangle_{N_1} = \langle 2 \cdot 5\rangle_3 = 1$

$\cancel{N_1 \langle N_1^{-1}\rangle_{N_2}} = \cancel{\quad}$;

the value from eq (24) is

$$A = N_2 \; ; \; B = N_1 \; ; \; C = N_2 \langle N_2^{-1}\rangle_{N_1} \; ; \; D = N_1 \langle N_1^{-1}\rangle_{N_2}$$

are

$$A = 5 \; ; \; B = 3 \; ; \; C = 5\langle 5^{-1}\rangle_3 = 10 \; ; \; D = 3\langle 3^{-1}\rangle_5 = 6$$

$\left(\text{i.e } \langle 5^{-1}\rangle_3 = 2 \text{ so } 5 \times 2 = 10\right)$ $C = 2\times5$. $D = 2\times3$ $\left(\text{i.e } \langle 3^{-1}\rangle_5 = 2 \text{ i.e } 3 \times 2 = 6\right)$

Substituting above values of Constants in eq (18) & (19) we arrive at index maps.

$$n = \langle 5n_1 + 3n_2\rangle_{15} \quad , \quad \begin{cases} 0 \le n_1 \le 2 \\ 0 \le n_2 \le 4 \end{cases} \rightarrow (29)$$

$\langle An_1 + Bn_2\rangle_N$

$$k = \langle 10k_1 + 6k_2\rangle_{15} \quad , \quad \begin{cases} 0 \le k_1 \le 2 \\ 0 \le k_2 \le 4 \end{cases} \rightarrow (30)$$

$\langle Ck_1 + Dk_2\rangle_N$

∴ DFT is

$$X\left[\langle 10k_1 + 6k_2\rangle_{15}\right] = \sum_{n_2=0}^{4} \left(\sum_{n_1=0}^{2} x\left[\langle 5n_1 + 3n_2\rangle_{15}\right] W_3^{n_1 k_1} \right) \cdot W_5^{n_2 k_2} \; , \; \begin{cases} 0 \le k_1 \le 2 \\ 0 \le k_2 \le 4 \end{cases} \rightarrow (31)$$

∴ index mapping eq (30) develops the 2D array represents of i/p as

$x(n) = x(5n_1 + 3n_2)$

$n_1 \backslash n_2$	0	1	2	3	4
0	$x[0]$	$x[3]$	$x[6]$	$x[9]$	$x[12]$
1	$x[5]$	$x[8]$	$x[11]$	$x[14]$	$x[2]$
2	$x[10]$	$x[13]$	$x[1]$	$x[4]$	$x[7]$

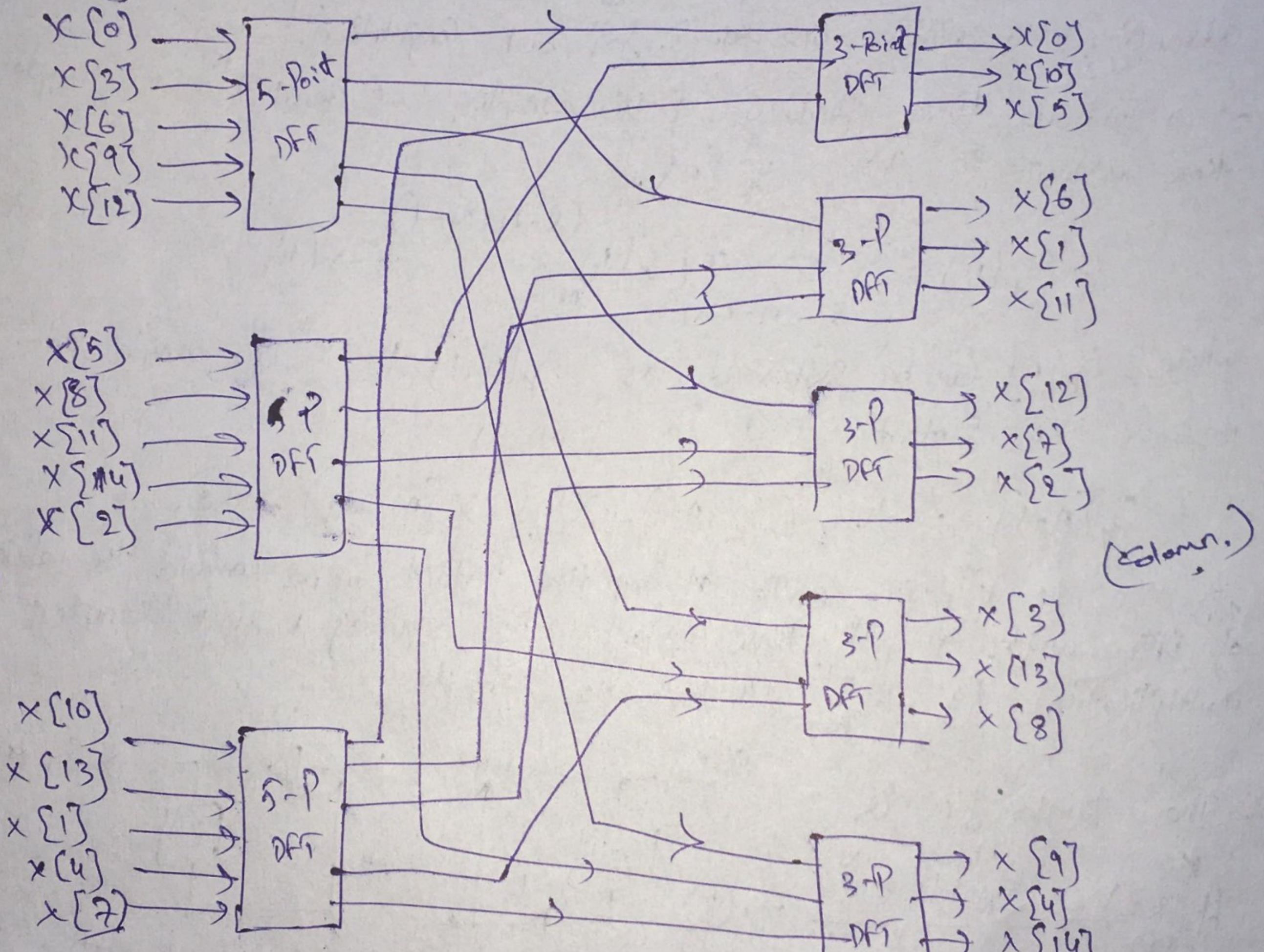
→ row.

(15) (16)

15-Point DFT of Each one of 3-rows given results 2-D array

n_1 \ k_2	0	1	2	3	4
0	G[0,0]	G[0,1]	G[0,2]	G[0,3]	G[0,4]
1	G[1,0]	G[1,1]	G[1,2]	G[1,3]	G[1,4]
2	G[2,0]	G[2,1]	G[2,8]	G[2,3]	G[2,4]

The 3-Point DFT of Each column of array $G[n_1, k_2]$ are carried out leading to desired 15-point X[k] given by.



k_1 \ k_2	0	1	2	3	4
0	X[0]	X[6]	X[12]	X[3]	X[9]
1	X[10]	X[1]	X[7]	X[13]	X[4]
2	X[5]	X[11]	X[2]	X[8]	X[14].

$k = \langle Ck_1 + Dk_2 \rangle_N$

↓ column.

2) Sliding Discrete Fourier Transforms:- (Time instants).

In some applications involving a very long sequence it is of interest to determine the spectral properties of subset consisting of fixed no:- of consecutive sequences at successive values of time instant 'n'.

for Eg:- if 'N' is length of $\subset$ N-Point DFT of finit length segment (subset)

i.e $N \subset$ N-Point DFT

it can be computed inside a length 'N' - sliding window.

→ ∴ The computation is repeated for each increasing value of 'N' - by advancing the window one sample for each computation.

→ To indicate time dependancy of DFT sampling we denot k^{th} DFT sample at time instant in as $S_k[n]$.

$$\therefore S_k[n] = \sum_{l=n-N+1}^{n} x[l] W_N^{(l-n+N-1)} \rightarrow (1)$$

where $S_k[n]$ can be expressed as $S_k[n-1]$ the k^{th} DFT sample at previous time instant $n-1$ is given as

$$S_k[n-1] = W_N^{-k}\left(S_k(n-1) + x\{n\} - x\{n-N\}\right) \rightarrow (2)$$

If once the k^{th} DFT sample at any time instant can be computed the value of DFT sample at all time instants by requiring a single complex multiplication by W_N^{-k} for each new sample

∴ The transfer func is

$$H^{k}_{SDFT}(z) = \frac{W_N^{-k}(1-z^{-N})}{1-W_N^{-k} z^{-1}} \rightarrow (3)$$

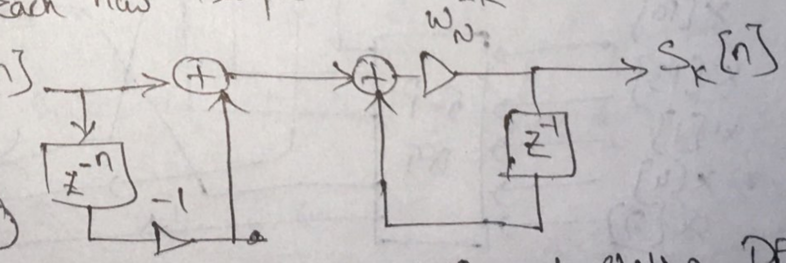


fig:- Recursive implementation of sliding DFT

∴ It is observed that there is a complex pole on unit circle at $z = W_N^{-k}$. Due to quantazation effect the system becomes unstable.

To maintain stability is $W_N^{-k} \rightarrow \gamma . W_N^{-k}$ where 'γ' is slightly smaller than '1' because of this modification error is introduced in the DFT sample being computed. so alternate stabilisation method is used.

3) DFT Computation over a narrow frequency Band : (frequency range)

In applications requiring the computation of the DFT samples over a specified frequency range, the FFT algorithms described earlier are not attractively in particular for sequence of very large length as they compute all samples of DFT.

2 - Computational efficient algorithms for calculation of subset of DFT samples.

(a) Zoom FFT (b) chirp Fourier Transforms.

(a) Zoom FFT :-

Used to compute the samples of an N-point DFT $X[k]$ of length-N sequence $x[n]$ in a small range of values of frequency index k,

$$i \leq k \leq i + K - 1 \quad \text{where} \quad 0 \leq i \leq N - K + 1 \rightarrow (a)$$

if N is an integer multiple of 'K' i.e. $N = K \cdot R$ which is always sastified by zero padding $x[n]$ if necessary.

To develop appropriate expression for samples of DFT in limited range. we use R-band polyphase decomposition to sequence $x[n]$ which develop R-sub sequences

$$x_r[m] = x[r + mR], \quad 0 \leq r \leq R - 1. \rightarrow (b)$$

of length 'K'

note:- $x[r+mR]$ is r^{th} sub-sequence of length 'K' obtained by down sampling $x[n]$ by factor of 'R'.

If $X[z]$ denotes z-transform of $x[n]$ then using a R-band polyphase decomposition

$$X(z) = \sum_{r=0}^{R-1} z^{-r} X_r(z^R) \rightarrow (c)$$

where

$$X_r(z) = \sum_{n=0}^{K-1} x[r + nR] z^{-n} \rightarrow (d)$$

Evaluating Eq (a) on unit circle at 'K' equally spaced points $z = W_N^{-k}$

$$i \leq k \leq i + K - 1$$

$$X[k] = \sum_{r=0}^{R-1} W_N^{kr} X_r[\langle k \rangle_K] \quad ; \; i \leq k \leq i + K - 1 \rightarrow (e)$$

The K-point DFT is periodic in 'l' with a period 'K'.

where Eq (c) represents i^{st} stage of radix-R decimation in time Cooley-Tukey FFT algorithm.

where $X_r[l]$ is K-Point DFT of sub sequence $x[r+mR]$

$$X_r[l] = \sum_{m=0}^{K-1} x[r+mR] \cdot W_K^{lm} \quad ; 0 \le l \le K-1 \rightarrow (f)$$

thus to compute $X[k]$ in the given range of K, we 1^{st} compute K-Point DFT's $X_r[l]$ of each of the R subsequence $x_r[m]$ obtained by down sampling $x[n]$ by factor 'r'.

if sampling rate changes from $\frac{1}{T}$ to $\frac{1}{2}$ then it means that sampling rate is decreased by factor 2. This is known as down sampling.

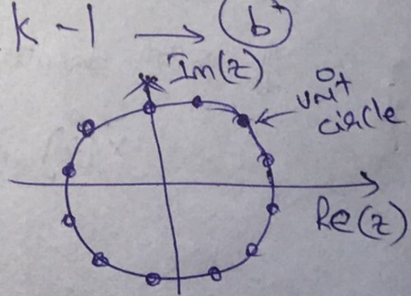
(b) Chirp Fourier Transform :- (or) Chirp Z-Transform :-

The DFT of an N-Point data sequence $x(n)$ may be viewed as Z-Transform of $x(n)$ evaluated at N-Equally spaced points on unit circle and also can be viewed as N-Equally spaced samples of F.T of data sequences

$$X(z) = \sum_{n=0}^{N-1} x(n) \cdot z^{-n} \rightarrow (a) \quad ; k \Rightarrow 0,1,2\ldots N-1$$

the objective is to evaluate $X(z)$ at K-Points z_k $(K<N)$ in z-plane:

$$z_k = r_k \cdot e^{j\omega k} = A \cdot V^{-k} \quad ; 0 \le k < K-1 \longrightarrow (b)$$



where

$$V = V_0 \cdot e^{-j\phi_0} \quad ; \rightarrow (c)$$

$$A = A_0 \cdot e^{j\theta_0} \rightarrow (d)$$

V_0 & A_0 are positive and real numbers

→ The point z_k defined by Eq (b) are located on a spiral contour with $z_0 = A$ being starting point on contour

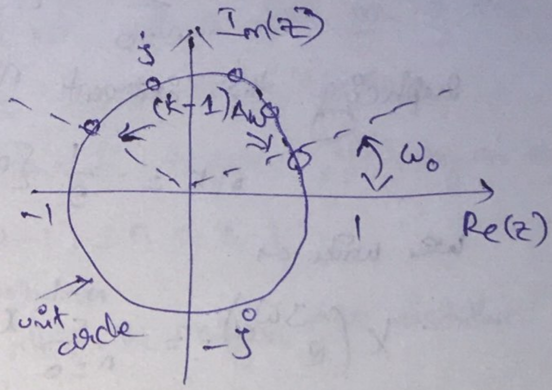
$\therefore$ The k-samples of Z-transform $X(z)$ to be computed are

$$X(Z_k) = \sum_{n=0}^{N-1} x(n) \cdot Z_k^{-n} \rightarrow (e)$$

$$X(Z_k) = \sum_{n=0}^{N-1} x(n) \cdot A^{-n} \cdot V^{nk} \quad ; \quad 0 \le k \le K-1 \rightarrow (F)$$

are called chirp Z-transform of sequence $x[n]$ computed efficiently using a special algorithm.

If $A_0 = V_0 = 1$, the contour is a portion of unit circle and points Z_k are equally spaced points on unit circles



now:- substituting $V_0 = 1$ and $\phi_0 = \Delta\omega$ in eq (c)

$$V = V_0 \cdot e^{-j\phi_0}$$

$$V = e^{-j\Delta\omega} \rightarrow (g)$$

and substituting $A_0 = 1$ and $\theta_0 = \omega_0$ in eq (d)

$$A = A_0 \cdot e^{+j\theta_0}$$

$$A = e^{j\omega_0} \rightarrow (h)$$

then k-point unit circle is given by

$$\omega_k = \omega_0 + k(\Delta\omega) \quad ; 0 \le k \le K-1 \rightarrow (i)$$

where ω_0 — is starting frequency point

$\Delta\omega$ — desired frequency resolution

if $X(Z_k)$ are simply the samples of Fourier transform $X(e^{j\omega})$ then

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j\omega_k n}$$

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j(\omega_0 + k(\Delta\omega))n} \quad ; 0 \le k \le K-1 \rightarrow (j)$$

of sequence $x[n]$ and will be referred to as chirp Fourier transform.

To develop CFT algorithm.

from Eq (g) substitute in Eq (j)

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x[n] \cdot e^{-j\omega_0 n} \cdot e^{-j\Delta\omega nk}$$

$$= \sum_{n=0}^{N-1} x[n] \cdot e^{-j\omega_0 n} \cdot V^{nk} \quad (\because V = e^{-j\Delta\omega}) \rightarrow (k)$$

Replacing the exponent nk in above Eq with the identity

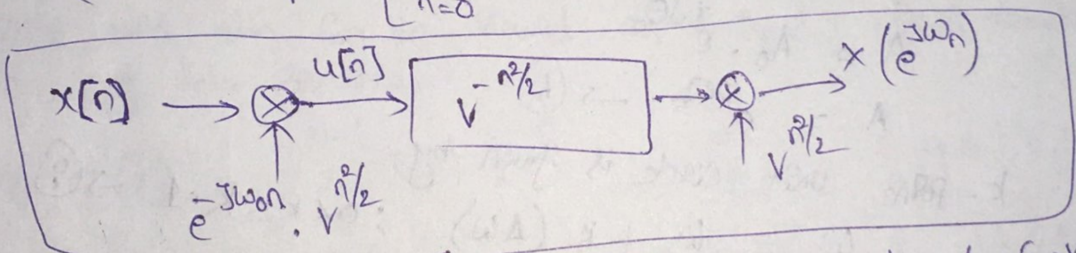
$$nk = \frac{1}{2}\left[n^2 + k^2 - (k-n)^2\right] \rightarrow (l)$$

we write as

$$X(e^{j\omega_k}) = \sum_{n=0}^{N-1} x[n] \cdot e^{-j\omega_0 n} \cdot V^{[n^2+k^2-(k-n)^2]\frac{1}{2}}$$

$$= \sum_{n=0}^{N-1} x(n) \cdot e^{-j\omega_0 n} \cdot V^{n^2/2} \cdot V^{k^2/2} \cdot V^{-(k-n)^2/2}$$

$$X(e^{j\omega_k}) = V^{k^2/2} \cdot \left[\sum_{n=0}^{N-1} x(n) \cdot e^{-j\omega_0 n} \cdot V^{n^2/2} \cdot V^{-(k-n)^2/2}\right] \rightarrow (m)$$



The right hand side of Eq (m) can be expressed in form of convolution

$$u[n] = x(n) \cdot e^{-j\omega_0 n} \cdot V^{n^2/2} \rightarrow (n)$$

on substituting Eq (n) in Eq (m)

$$X(e^{j\omega_k}) = V^{k^2/2} \sum_{n=0}^{N-1} u[n] \cdot V^{-(k-n)^2/2} \quad 0 \le k \le K-1 \rightarrow (o)$$

interchaing the variables k and n we rewrite Eq (o) as

$$X(e^{j\omega_k}) = V^{n^2/2} \sum_{n=0}^{N-1} u[k] \cdot V^{-(n-k)^2/2} \; ; 0 \le n \le K-1 \rightarrow (P)$$

The complex exponential sequence

$V^{-n^2/2} = e^{j(\Delta\omega)n^2/2}$ has linearly increasing frequency and is often referred to as chirp signal.

for Computation of F.T samples the right hand side of Eq (P) can be interpreted as convolution of sequence $u[n]$ with the sequence $V^{-n^2/2}$ multiplied by factor $V^{n^2/2}$

$$X(e^{j\omega_n}) = V^{n^2/2}\left(u[n] \circledast V^{-n^2/2}\right) \rightarrow (q)$$

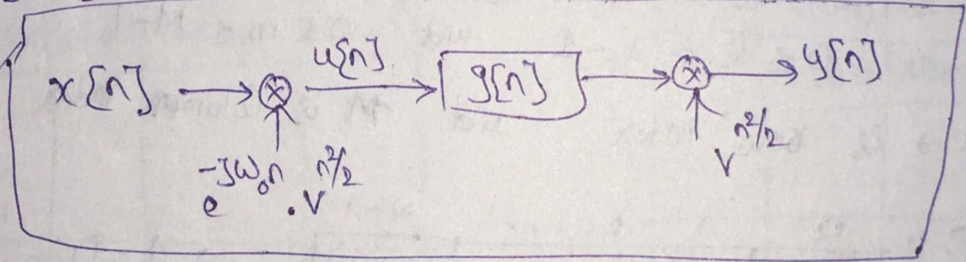
where $u[n]$ is finite length sequence of length N

→ The convolution sum of Eq (q) is carried out over a finit no. of term. To evaluate CFT sample by replacing the infinite length sequence $V^{-n^2/2}$ in Eq (q) with a finite length non-causal sequence $g[n]$ defined as

$$g[n] = \begin{cases} V^{-n^2/2} & -(N-1) \le n \le K-1 \\ 0 & \text{otherwise} \end{cases}$$

which is a LTI finite impulse response (FIR) filter. The modified CFT algorithm is given by

$$y[n] = X(e^{j\omega_n}) = V^{n^2/2}\left(u[n] \circledast g[n]\right) \rightarrow (r)$$



//.

4) **Radix - 4 FFT algorithm:-**

is when the no:- of data points 'N' in the DFT is power of 4 (i.e $N = 4$

offcourse we can use radix - 2 algorithm for computation, but for more efficient computationally to employ a radix - 4 FFT algorithm used.

→ Note:- Divide & conquer approach to computation of DFT:- used to develop of computationally efficient algorithm for DFT is made possible

→ based on decomposition of a N-point DFT into successive smaller DFT's. Computation of an N-point DFT where N can be factored as product of two integers

$$N = LM.$$

now sequences $x(n)$; $0 \le n \le N-1$ can be stored either in 1-D array indexed by 'n'

(or) as 2-D array indexed by 'l' and 'm'

where $0 \le l \le L-1$ and $0 \le m \le M-1$

where $l \to$ is row index and m is column index

$n \to$

0	1	2 . . .	N-1
x(0)	x(1)	. . .	x(N-1)

→ 1-D

m column index →

l (row index) ↓	0 ↓	1 ↓	2 . . . ↓	M-1 ↓
0	x(0,0)	x(0,1)	. . . .	
1	x(1,0)	x(1,1)		
2	⋮		. . . .	
⋮	⋮		. . . .	
L-1		.		

→ 2-D.

for Eg we select mapping $n = Ml + m$ i.e arrangment is 1st row consists M- elements of $x(n)$, second row consists M elements of $x(n)$ and so on, on other hand mapping

$$n = l + mL.$$

1st element of $x(n)$ is first column, next is L elements in 2nd column. soon

Simillar arrangment used to Computed DFT values.
mapping for index k to pair indices (p, q) where

$$0 \le p \le L-1 \; ; \; 0 \le q \le M-1$$

If mapping is $k = Mp + q$
The DFT stored on row wise basis 1^{st} row Contains M elements of DFT $X(k)$, 2^{nd} row Contains next set of M Elements and so on.

If $k = qL + p$
result Column wise of storage $X(k)$ when 1^{st} element L-Element are stored in first Column the second set of L element so on.

Row wise $n = Ml + m$

$l \backslash m$	0	1	2	M−1
0	$x(0)$	$x(1)$	$x(2)$	$x(M-1)$
1	$x(M)$	$x(M+1)$	$x(M+2)$ ...	$x(2M-1)$
2	$x(2M)$	$x(2M+1)$	$x(2M+2)$ ...	$x(3M-1)$
⋮	⋮			
L−1	$x((L-1)M)$	$x((L-1)(M+1))$		$x(LM-1)$

Column wise $n = l + mL$

$l \backslash m$	0	1	2		M−1
0	$x(0)$	$x(L)$	$x(2L)$		$x((M-1)L)$
1	$x(1)$	$x(L+1)$	$x(2L+1)$	. . .	.
2	$x(2)$	$x(L+2)$	$x(2L+2)$		
⋮	⋮	⋮	⋮		
L−1	$x(L-1)$	$x(2L-1)$	$x(3L-1)$	. . . .	$x(LM-1)$

$$X(p,q) = \sum_{l=0}^{L-1} \left\{ W_N^{lq} \left[\sum_{m=0}^{M-1} x(l,m) \cdot W_M^{mq} \right] \right\} W_L^{lp} \rightarrow (i)$$

Radix-4 FFT algorithm:-

let Consider $L = 4$ and $M = N/4$ in divide and Conquer approach.
for choices L & M we have $l, p = 0, 1, 2, 3$.

$$m, q = 0, 1, \ldots, N/4 - 1$$

where $n = 4m + l \longrightarrow (a)$

$$k = \left(\frac{N}{4}\right)p + q \longrightarrow (b)$$

Thus we split (or) decimate N-point i/p sequence into 4-subsequences.

$$x(4n) \; ; \; x(4n+1) \; ; \; x(4n+2) \; ; \; x(4n+3) \quad ; \; n = 0, 1, \ldots \frac{N}{4} - 1$$

from eq (i) we obtain

$$X(p,q) = \sum_{l=0}^{3} \left[W_N^{lq} \, F(l,q) \right] W_4^{lq} \quad ; \; p = 0, 1, 2, 3 \longrightarrow (c)$$

where $F(l,q) = \sum_{l=0}^{(\frac{N}{4}-1)} x(l,m)\, W_{N/4}^{mq}$ $\quad l=0,1,2,3 \rightarrow (d)$

$q = 0,1,2 \ldots \frac{N}{4}-1$

and $X(l,m) = x(4m+l) \rightarrow (e)$

$X(p,q) = X\left(\frac{N}{4}p+q\right) \rightarrow (f)$

Thus four $\frac{N}{4}$ point DFT obtained from Eq (d) are combined to yield N-point DFT

The Eq (e) for combining the $\frac{N}{4}$ point DFT defines a radix-4 decimation in time butterfly expressed as matrix form

$$\begin{bmatrix} X(0,q) \\ X(1,q) \\ X(2,q) \\ X(3,q) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \cdot \begin{bmatrix} W_N^0 F(0,q) \\ W_N^q F(1,q) \\ W_N^{2q} F(2,q) \\ W_N^{3q} F(3,q) \end{bmatrix} \rightarrow (e)$$

The above matrix in right hand side Eq. is defined as.

wkt :- $\boxed{W_N = \frac{2\pi}{N} \cdot N}$.

now :- W_N^{lq} is taken as $l = 0,1,2,3$ (row) represents and $q = 0,1,2,3$ (column) represents $(\because q = 0 \rightarrow \frac{N}{4}-1)$ where $N=4$.

If $l=0$ and $q=0,1,2,3$ then.

$W_N^{lq} = W_4^{(0)(0)} = \underline{1}$; $W_4^{(0)\cdot(1)} = \underline{1}$; $W_4^{(0)\cdot(2)} = \underline{1}$; $W_4^{(0)(3)} = \underline{1}$

If $l=1$ and $q=0,1,2,3$ then

$W_N^{lq} = W_4^{(1)(0)} = \underline{1}$; $W_4^{(1)(1)} = e^{j\frac{2\pi}{4}(1)} = \left(\cos\frac{\pi}{2} - j\sin\frac{\pi}{2}\right) = -j$

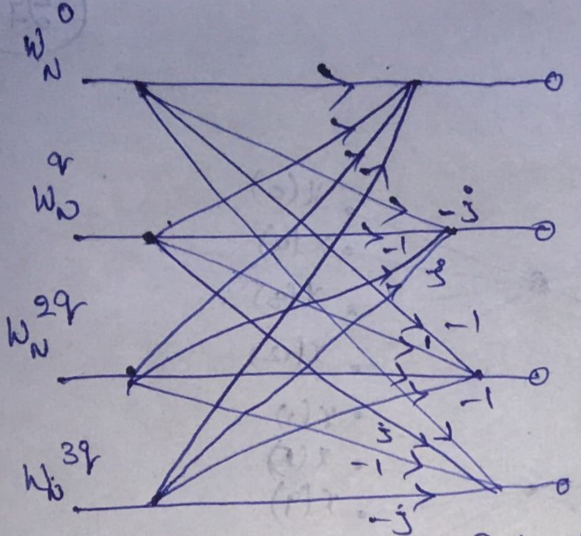
$W_4^{(1)(2)} = e^{j\frac{2\pi}{4}(2)} = (\cos\pi - j\sin\pi) = -1$

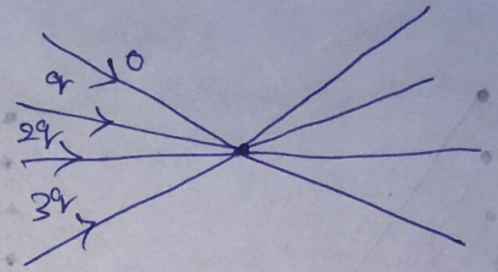
$W_4^{(1)(3)} = e^{j\frac{2\pi}{4}(3)} = \cos\frac{3\pi}{2} - j\sin\frac{3\pi}{2} = j$

Similarly :- $l=2$; $q = 0,1,2,3 \Rightarrow 1, -1, 1, -1$

$l=3$, $q=0,1,2,3 \Rightarrow 1, j, -1, -j$

$\therefore$ The matrix is taken as
$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} .//$$





Basic Butterfly Computation in radix FFT algorithm.

→ The decimation in time Procedure Can be repeated recursively V-times. The resulting FFT algorithm Consists of V-stages where Each stage Contains $\frac{N}{4}$ butterflies, and Computational burden for algorithm is $3v\frac{N}{4} = \left(\frac{3N}{8}\right)\log_2 N$ is complex multiplication and $\left(3\frac{N}{2}\right)\log_2 N$ is Complex additions.

→ the no of multiplications is reduced by 25% but no of additions is increased by 50% from $N\log_2 N$ to $\left(3\frac{N}{2}\right)\log_2 N$

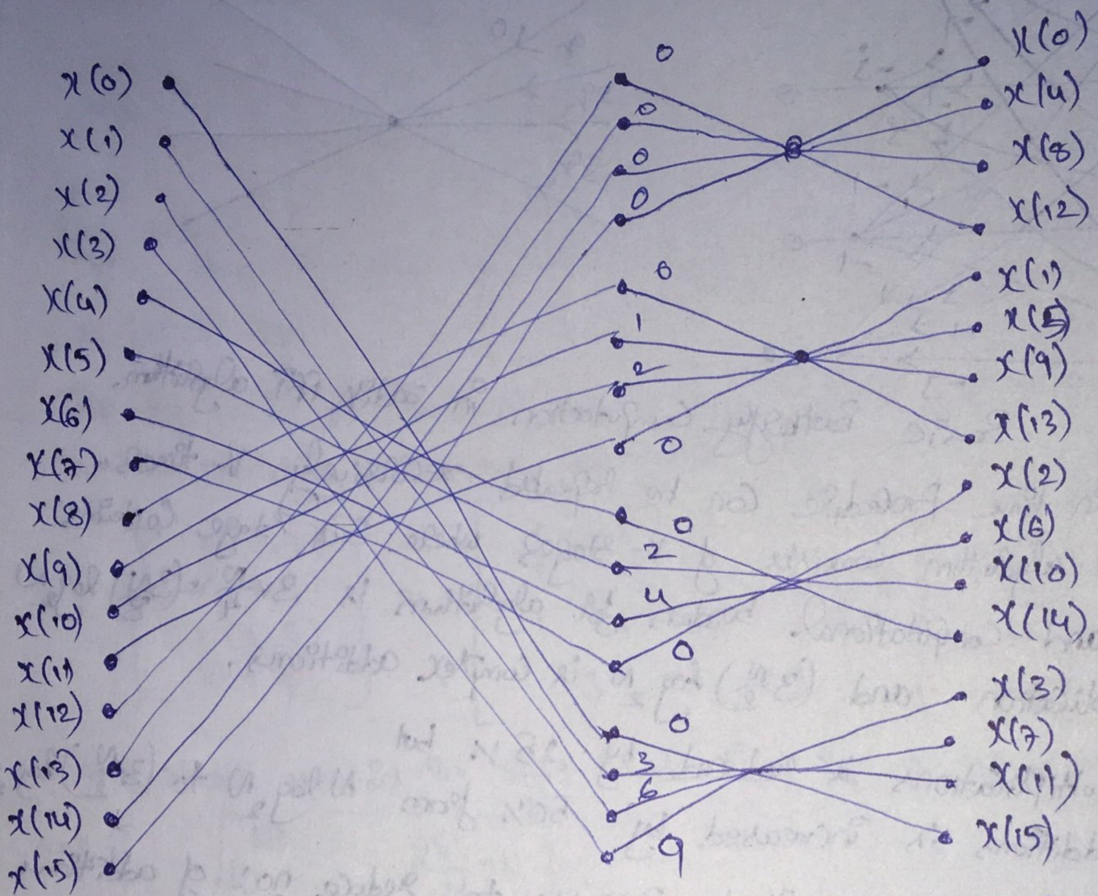
→ By addition of 2-steps it is Possible to reduce no:- of additions Per butterfly from 12 to 8

i.e shown for matrix linear transformation in Eq (e)

$$\begin{bmatrix} X(0,q) \\ X(1,q) \\ X(2,q) \\ X(3,q) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -j \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & j \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} W_N^0 F(0,q) \\ W_N^q F(1,q) \\ W_N^{2q} F(2,q) \\ W_N^{3q} F(3,q) \end{bmatrix}$$

Each matrix multiplication involves 4-additions for total eight additions the total no:- of Complex addition reduced to $N\log_2 N$. which is radix-2 FFT

→ Radix-4 decimation in time FFT algorithm for N=16 is shown below



16-Point radix-4 decimation in time algorithm with i/p in normal order and o/p in digit reversed order.

Split Radix FFT algorithm:-

It uses both radix-2 & Radix-4 decomposition in same FFT algorithm

·) The radix-2 ~~DIT~~ Decimation in frequency FFT algorithm the even no:- samples of N-point DFT are given as

$$X(2k) = \sum_{n=0}^{\frac{N}{2}-1} \left[x(n) + x\left(n+\frac{N}{2}\right)\right] \cdot W_{N/2}^{nk} \quad ; k=0,1,\ldots \frac{N}{2}-1 \rightarrow (a)$$

The odd no:- samples $\{X(2k+1)\}$ of DFT require the pre-multiplication of i/p sequence with twiddle factor W_N^n.

For these samples a radix-4 decomposition produces some computational efficiency bcoz of 4-point DFT has largest multiplication free Butterfly.

→ If we use radix-4 decimation in frequency FFT algorithm for odd numbered samples N-point DFT we obtain following $\frac{N}{4}$-point DFT's

$$X(4K+1) = \sum_{n=0}^{\frac{N}{4}-1} \left\{\left[x(n) - x\left(n+\frac{N}{2}\right)\right] - j\left[x\left(n+\frac{N}{4}\right) - x\left(n+\frac{3N}{4}\right)\right]\right\} \cdot W_N^n \cdot W_{N/4}^{kn} \rightarrow (b)$$

$$X(4K+3) = \sum_{n=0}^{\frac{N}{4}-1} \left\{\left[x(n) - x\left(n+\frac{N}{2}\right)\right] + j\left[x\left(n+\frac{N}{4}\right) - x\left(n+\frac{3N}{4}\right)\right]\right\} \cdot W_N^{3n} \cdot W_{N/4}^{kn} \rightarrow (c)$$

Thus N-point DFT is decomposed into one $\frac{N}{2}$ point DFT without additional twiddle factors and two $\frac{N}{4}$ point DFTs ~~which~~ with twiddle factors.

→ The below flow graph shows 32-point decimation in frequency SRFFT algorithm for N=32, the top 16 points constitute sequences at stage A of the computation

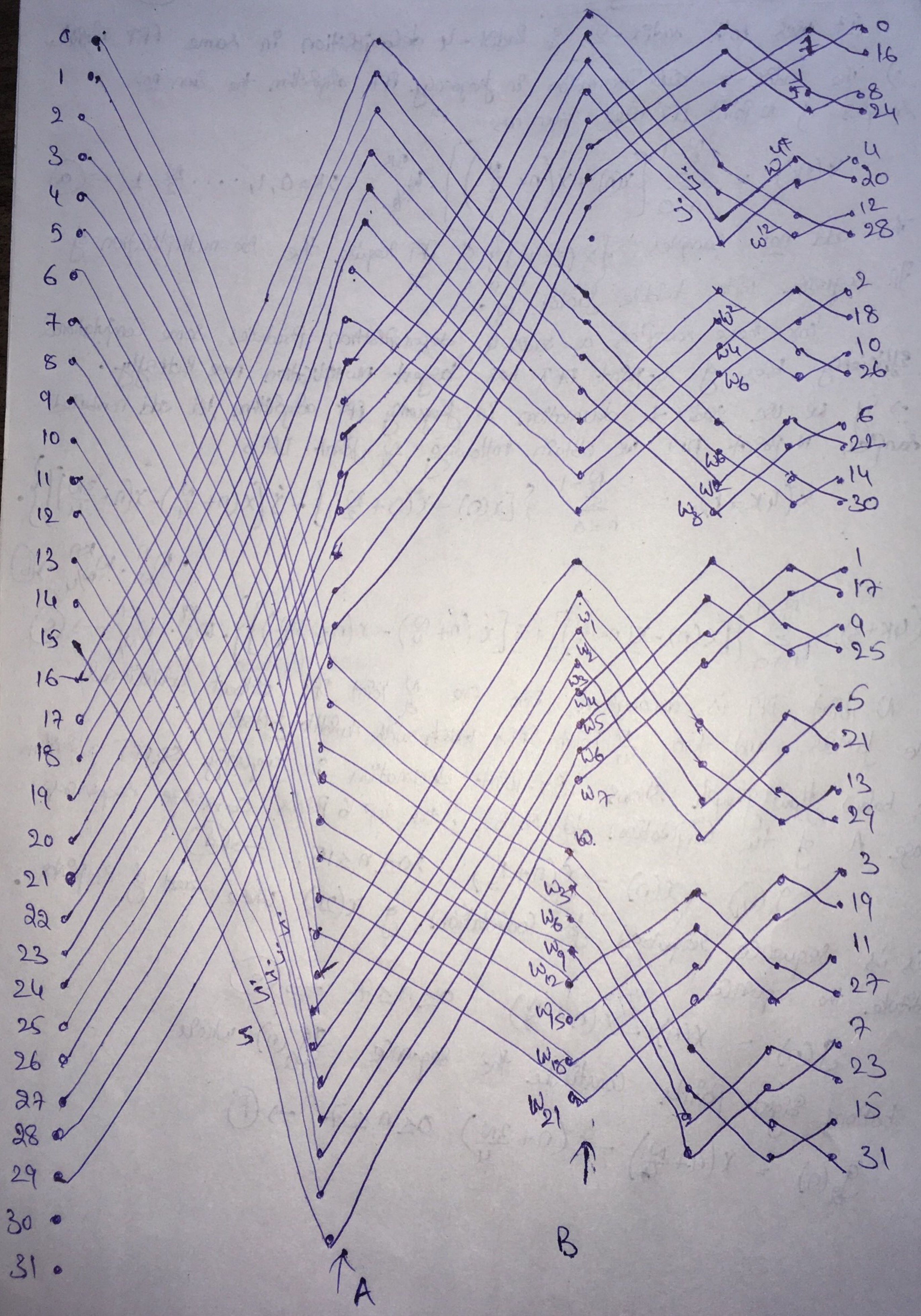
$$g(n) = x(n) = x\left(n+\frac{N}{2}\right) \quad ; 0 \le n \le 15. \rightarrow (d)$$

This is sequence required for computation of $X(2K)$ where rest of 8-points constitute the sequence

$$g_1(n) = x(n) - x\left(n+\frac{N}{2}\right) \quad 0 \le n \le 7 \rightarrow (e)$$

where the bottom eight points constitute the sequence $\{g_2(n)\}$

$$g_2(n) = x\left(n+\frac{N}{4}\right) - x\left(n+\frac{3N}{4}\right) \quad 0 \le n \le 7 \rightarrow (F)$$


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
A
B
0
16
8
24
4
20
12
28
2
18
10
26
6
22
14
30
1
17
9
25
5
21
13
29
3
19
11
27
7
23
15
31
W^4
W^{12}
W^2
$W4$
$W6$
W_1
W_2
W_3
W_4
W_5
W_6
W_7
W_3
W_6
W_9
W_{12}
W_{15}
W_{18}
W_{21}

→ the sequence $g_1(n)$ & $g_2(n)$ are used in the computation of $X(4k+1)$ & $X(4k+3)$ radix-4 $0 \le k \le 7$

At stage B the bottom eight points constitute the computation of $[g_1(n) + g_2(n) j]\, W_{32}^{3n}$ & the next eight points from the bottom constitute the computation of $[g_1(n) - j g_2(n)]\, W_{32}^{n}$ which is used to compute $X(4k+1)$ & $X(4k+3)$ $0 \le k \le 7$

next begining with 16-point at stage A, we decompose the computation into an eight point radix-2 DFT and two 4-point radix-4 DFT

- 8-point constitute sequence $g_0'(n) = g_0(n) + g_0(n+\frac{N}{2})$

→ thus at stage B the top 8-point constitute the two 4-point sequences $0 \le n \le 7$ and next 8-point constitute the two 4-point sequences

$g_1'(n)$ and $j g_2'(n)$

$g_1'(n) = g_0(n) - g(n+\frac{N}{a})$; $0 \le n \le 7$

$g_2'(n) = g_0(n+\frac{N}{4}) - g_0(n+\frac{3N}{4})$; $0 \le n \le 3$

hence each 8-point DFT is decomposed into a 4-point radix-2 DFT and a 4-point radix-4 DFT.//