Power System Protection II

Saeed Lotfifard

Washington State university

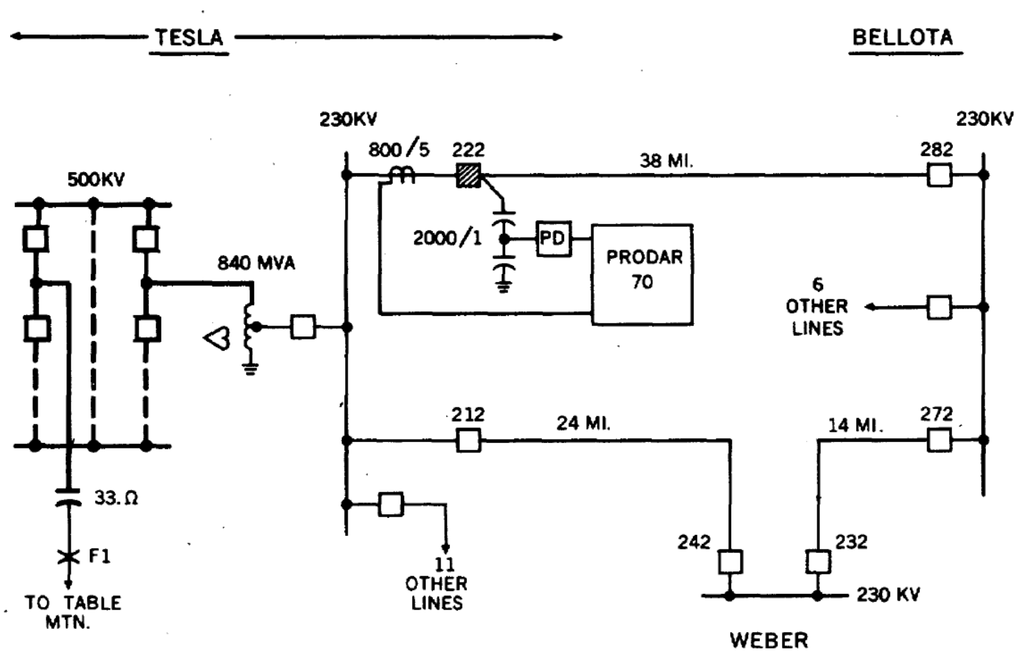
EE 511

Outline

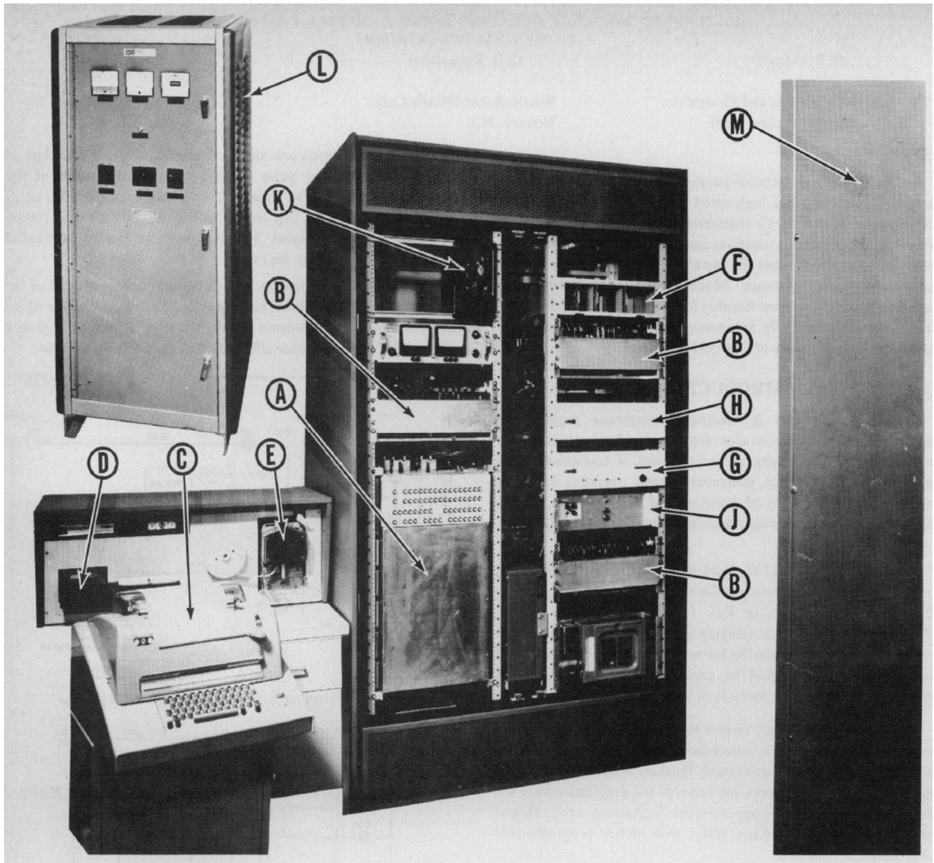
- Short-window algorithm
  - Mann& Morrison
  - Prodar
- Long-window algorithm
  - Fourier algorithm (full-cycle, half cycle)
  - Least squares algorithm

The first digital protective relay in the U.S

G.D. Rockefeller, commissioned in 1971 at Pacific Gas & Electric Company.

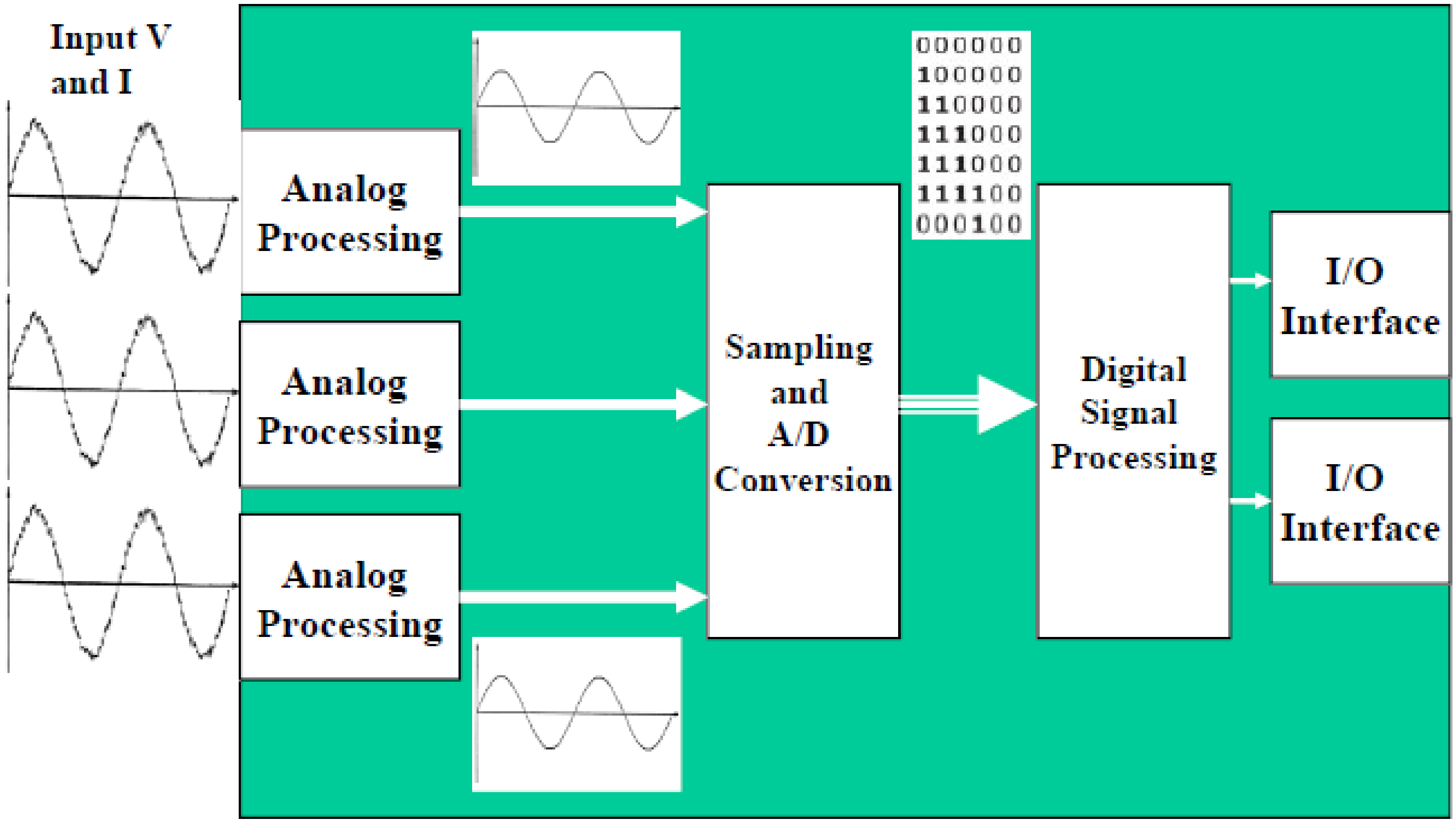


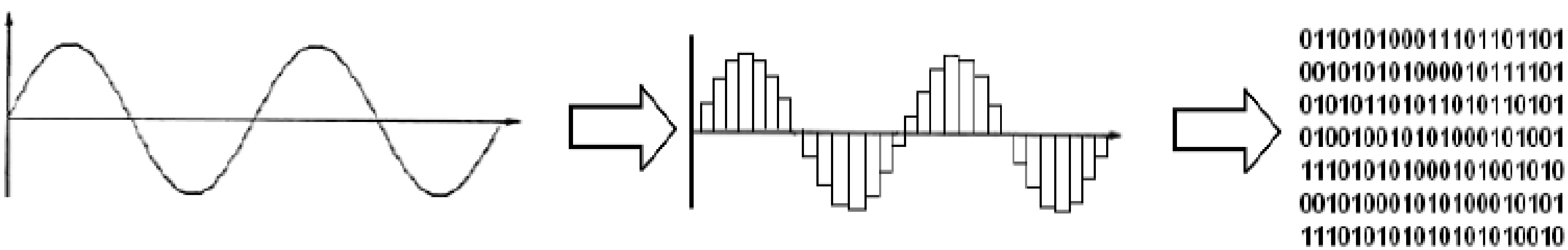
Tesla-Bellota 230 kVline



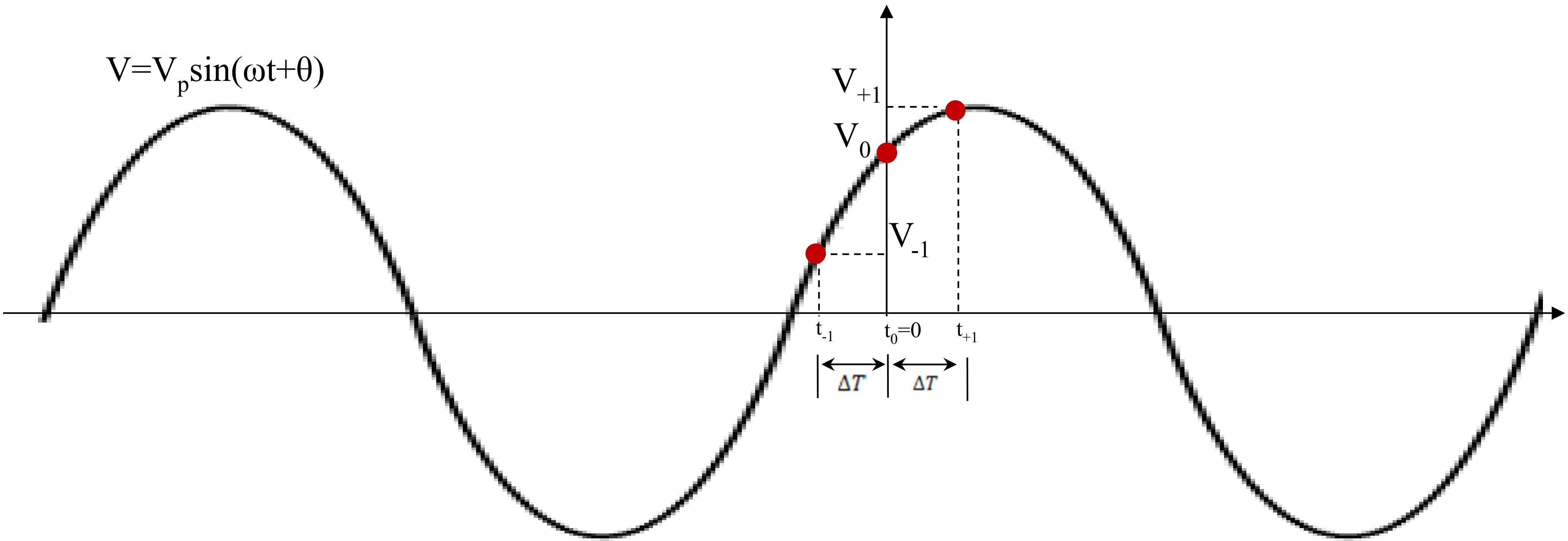
Prodar 70 Digital Protection System based on Westinghouse P-2000 process control computer

The cabinet contains a Westinghouse P-2000 process control computer, the base of the system. The computer main frame (A) is interfaced to peripheral gear via standard I/O panels at points (B). Also included in the standard P'- 2000 system are a Teletype KSR-35 typewriter (C), and a programmer's console having 60 char./sec. paper tape reader (D) and punch (E). Units added for the application include an analog signal conditioning package (ACP) at (F); a medium-speed multiplexer equipment, digitizer (G); an SPM data buffer interface and A/D control unit (H); and a power supply for the latter two items at (J). Also, the static auxiliary relay at (K) performed circuit breaker tripping in lab tests. Inset (L) shows the 3 kVA inverter used to provide continuous system power. (M) is the location of the automatic-start oscillograph used in evaluating system performance.


Input V
and I
Analog
Processing
Analog
Processing
Analog
Processing
000000
100000
110000
111000
111000
111100
000100
Sampling
and
A/D
Conversion
Digital
Signal
Processing
I/O
Interface
I/O
Interface


011010100011101101101
001010101000010111101
010101101011010110101
010010010101000101001
111010101000101001010
001010001010100010101
111010101010101010010

Mann& Morrison algorithm (short window)



$$V_0 = V_p \sin\theta$$

$$V_{-1} = V_p \sin(\theta - \omega_0 \Delta T)$$

$$V_{+1} = V_p \sin(\theta + \omega_0 \Delta T)$$

$$(A\sin\theta)^2 + (A\cos\theta)^2 = A^2$$

$$A = \sqrt{(A\sin\theta)^2 + (A\cos\theta)^2}$$

$$\Delta T = \frac{1}{n \times f}$$

f =60 or 50 and n is number of samples per cycle

Prodar 70 have common mathematical and logical techniques described in Ian Morrison's work at the University of New South Wales, Australia.

Mann& Morrison algorithm (short window)

$$(A\sin\theta)^2 + (A\cos\theta)^2 = A^2$$

$$A = \sqrt{(A\sin\theta)^2 + (A\cos\theta)^2}$$

$$V_0 = V_p \sin\theta$$

$$V_{-1} = V_p \sin(\theta - \omega_0 \Delta T)$$

$$V_{+1} = V_p \sin(\theta + \omega_0 \Delta T)$$

$$V = V_p \sin(\omega_0 t + \theta)$$

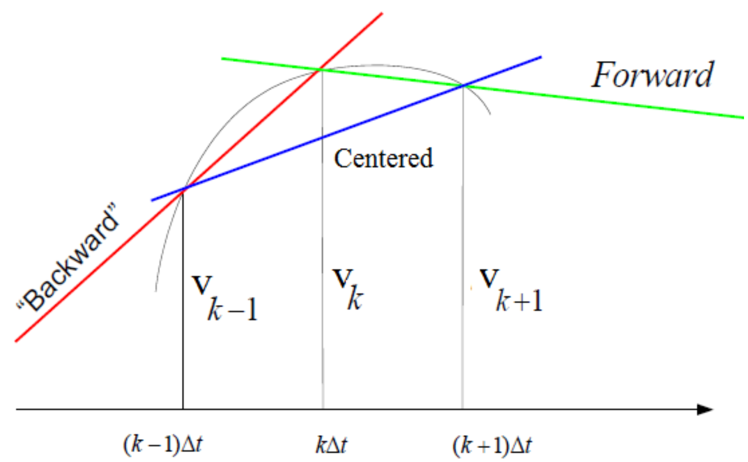
$$\frac{dV}{dt}\Big|_{t=0} = \frac{d(V_p \sin(\omega_0 t + \theta))}{dt}\Big|_{t=0} = V_p \omega_0 \cos\theta$$

$$V_p \cos\theta = \frac{dV}{\omega_0 dt}\Big|_{t=0} = \frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}$$

$$V_p = \sqrt{(V_0)^2 + \left(\frac{dV}{\omega_0 dt}\Big|_{t=0}\right)^2}$$

$$\tan\theta = \frac{V_p \sin\theta}{V_p \cos\theta} = \frac{V_0}{\frac{dV}{\omega_0 dt}\Big|_{t=0}} = \frac{V_0}{\frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}}$$

Different methods for calculation of numerical derivative



$$\textit{Backward} : \frac{V_k - V_{k-1}}{\Delta t}$$

$$\textit{Forward} : \frac{V_{k+1} - V_k}{\Delta t}$$

$$\text{Centered} : \frac{V_{k+1} - V_{k-1}}{2\Delta t}$$

More accurate method

Man& Morrison algorithm (short window)

$$V = V_p \sin(\omega_0 t + \theta)$$

$$V_p = \sqrt{(V_0)^2 + (\frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T})^2}$$

$$\theta = \tan^{-1}(\frac{V_0}{\frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}})$$

$\Delta T = \frac{1}{n \times f}$ f =60 or 50 and n is number of samples per cycle

Mann& Morrison algorithm (short window)

to estimate $V_p cos\theta$

$$V_p cos\theta = \frac{dV}{\omega_0 dt}\Big|_{t=0} = \frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}$$

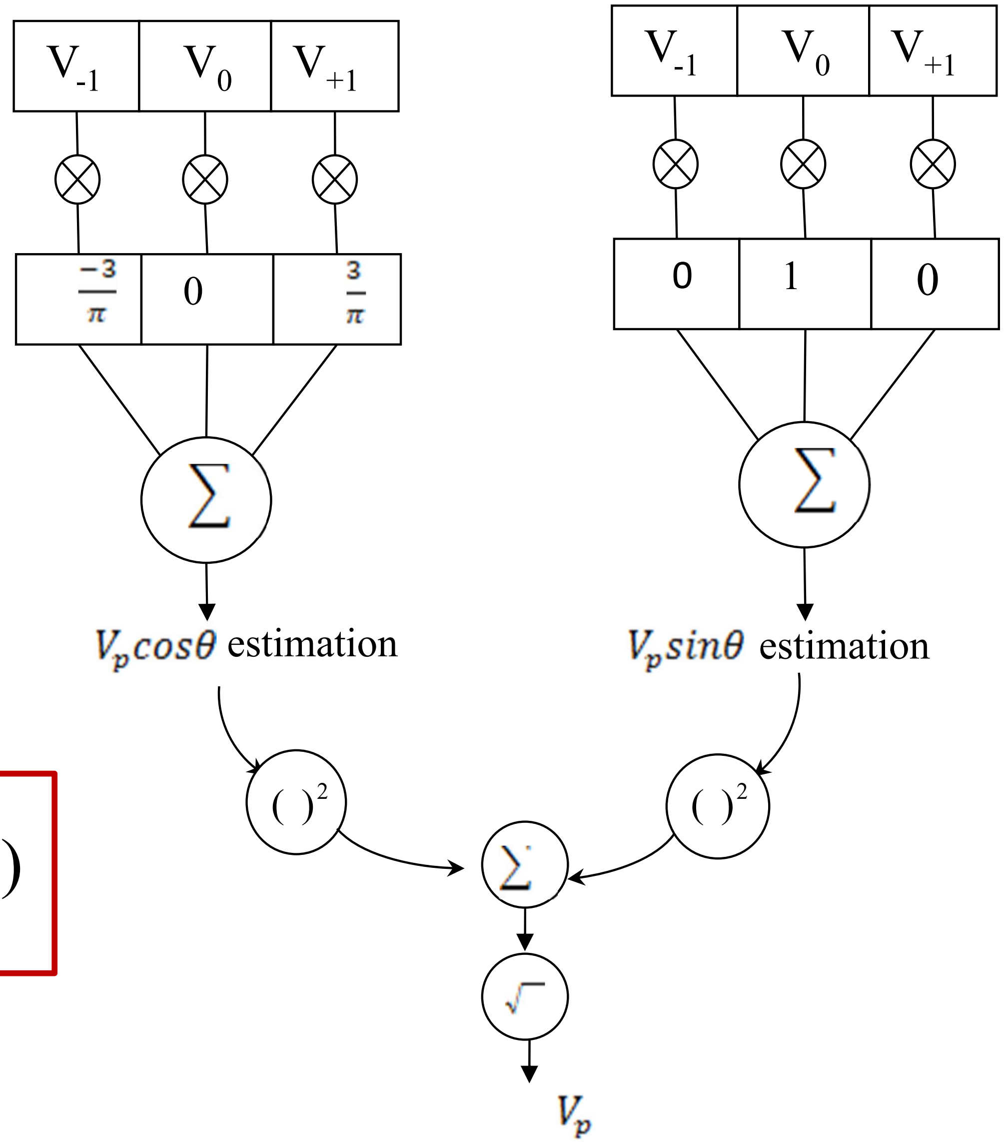
$$\begin{cases} f = 60 \rightarrow \omega_0 = 2\times\pi\times f = 2\times\pi\times 60 \\ n = 12 \rightarrow \Delta T = \dfrac{1}{12\times 60} = \dfrac{1}{720} \end{cases} \rightarrow$$

$$2\times\omega_0\times\Delta T = 2\times(2\times\pi\times 60)\times\frac{1}{720} = \frac{\pi}{3}$$

$$\boxed{V_p cos\theta = \frac{dV}{\omega_0 dt}\Big|_{t=0} = \frac{3}{\pi}\times(V_{+1} - V_{-1})}$$

to estimate $V_p sin\theta$

$$\boxed{V_p \sin\theta = V_0}$$



Frequency response of Man& Morrison algorithm (short window)

$$f(V) = \frac{3}{\pi} \times (V_{+1}) + 0 \times (V_0) - \frac{3}{\pi} \times (V_{-1})$$

$$f(Z) = \frac{3}{\pi} \times (Z^{+1}) - \frac{3}{\pi} \times (Z^{-1})$$

$$Z = e^{J\omega\Delta T}$$

$$H(\omega) = \frac{3}{\pi} \times (e^{J\omega\Delta T}) - \frac{3}{\pi} \times (e^{-J\omega\Delta T})$$

$$e^{J\omega\Delta T} = \cos(\omega\Delta T) + j\sin(\omega\Delta T)$$

$$= \frac{3}{\pi} \times [\cos(\omega\Delta T) + j\sin(\omega\Delta T) - \cos(\omega\Delta T) + j\sin(\omega\Delta T)] = \frac{6}{\pi} \times j\sin(\omega\Delta T)$$

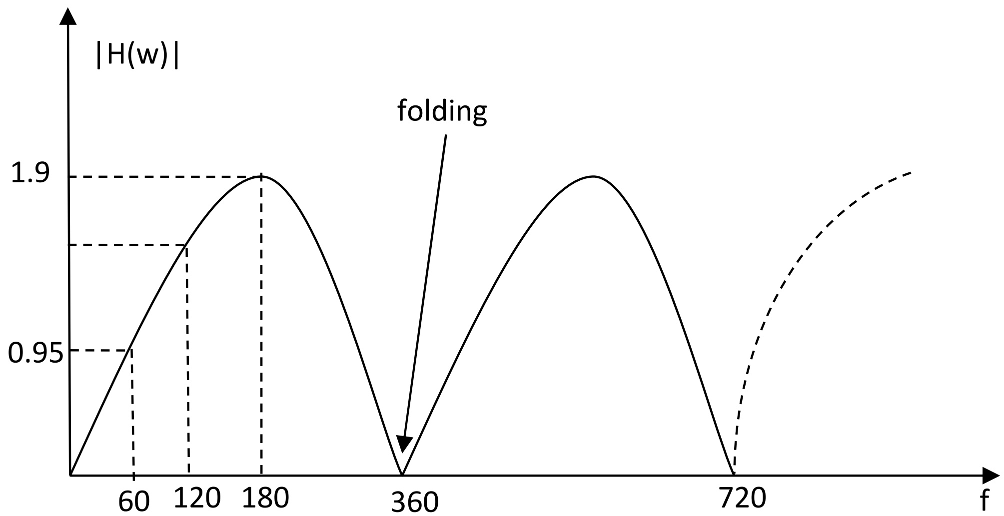
$$|H(\omega)| = \frac{6}{\pi} \times \sin(\omega\Delta T)$$

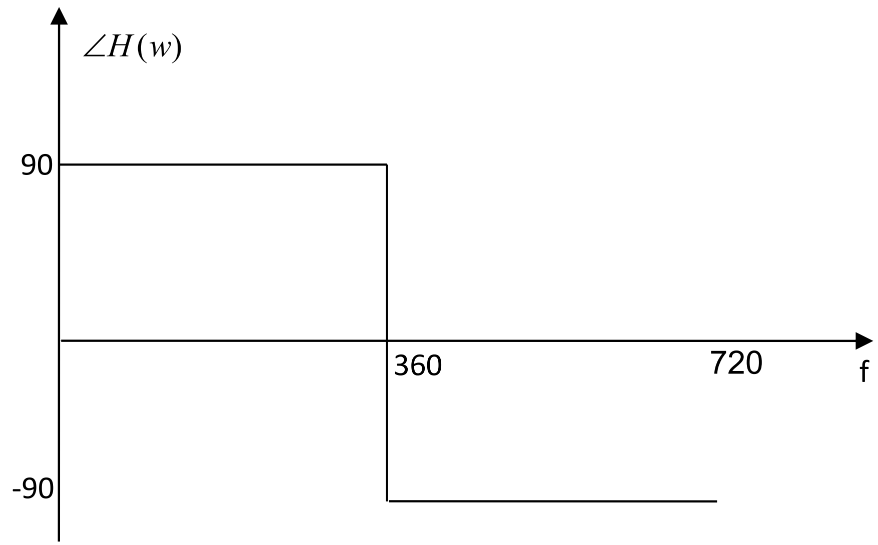
Frequency response of Mann& Morrison algorithm (short window)

f	ω=2fπ	ω ΔT	|H(w)|	$\angle H(w)$
0	0	0	0	90
10	20π	π/36	0.16	90
20	40π	π/18	0.3	90
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
60	120 π	π/6	3/ π = 0.95	90
90	180 π	π/4	1.35	90
180	360 π	π/2	1.9	90

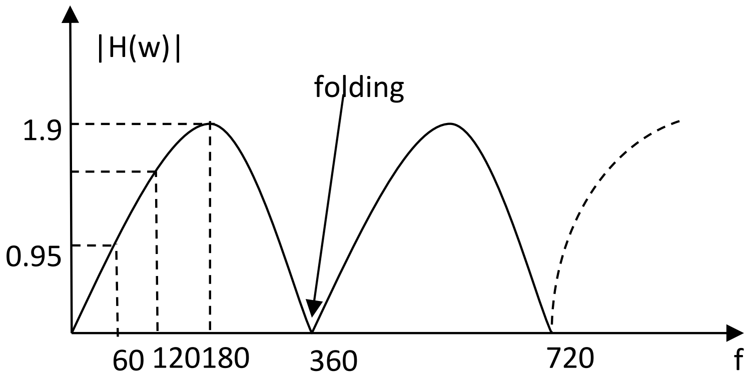
$$H(\omega) = \frac{6}{\pi} \times j\sin(\omega\Delta T)$$

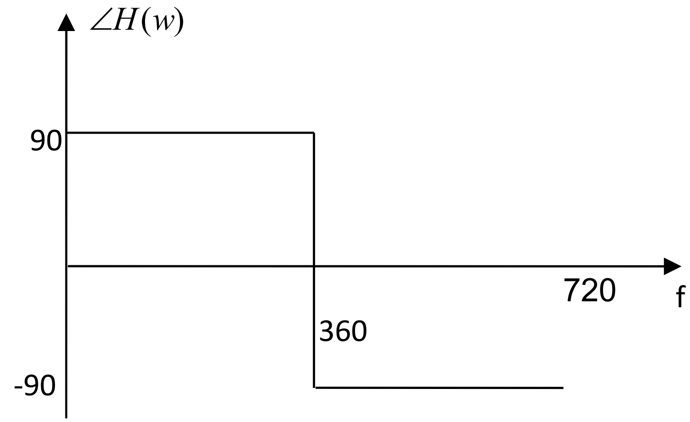
$$\Delta T = \frac{1}{(12 \times 60)}$$





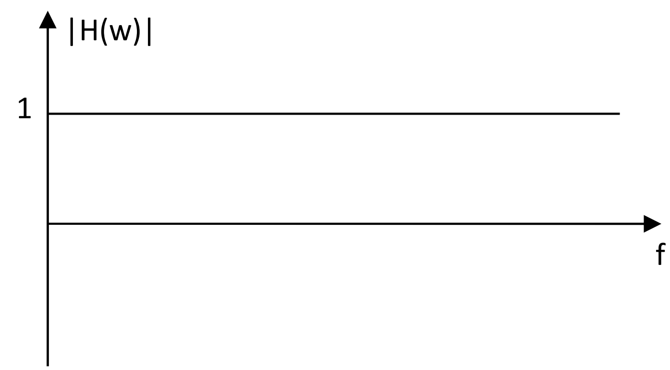
Frequency response of Mann& Morrison algorithm (short window)

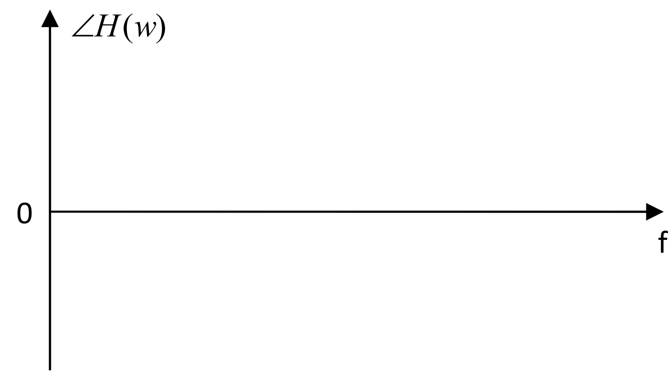




H2(W)

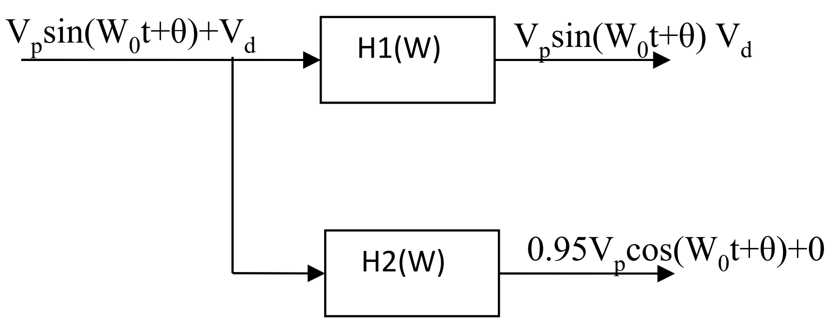
$\frac{-3}{\pi}$	0	$\frac{3}{\pi}$





H1(W)

0	1	0



W_0=60Hz
$0.95\ \angle 90$

$$V_p{}^2 = (A)^2 + (B)^2$$

$$V_p^2 = [V_d + V_p \sin(\omega_0 t+\theta)]^2 + [0.955\ V_p \cos^2(\omega_0 t+\theta)]^2$$

$$= V_p^2 \sin^2(\omega_0 t+\theta) + 2V_pV_d \sin(\omega_0 t+\theta) + V_d^2 + 0.955^2 V_p^2 \cos^2(\omega_0 t+\theta)$$

$$= 0.956 V_p^2 - 0.044 V_p^2 \cos(2\omega_0 t + 2\theta) + V_d^2 + 2V_pV_d \sin(\omega_0 t+\theta)$$

Mann& Morrison algorithm (short window)

$$V = V_p \sin(\omega_0 t + \theta)$$

$$V_p = \sqrt{(V_0)^2 + (\frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T})^2}$$

$$\theta = \tan^{-1}(\frac{V_0}{\frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}})$$

$\Delta T = \frac{1}{n \times f}$ f =60 or 50 and n is number of samples per cycle

Mann& Morrison algorithm (short window)

to estimate $V_p cos\theta$

$$\frac{dV}{\omega_0 dt}\Big|_{t=0} = \frac{V_{+1} - V_{-1}}{2\omega_0 \Delta T}$$

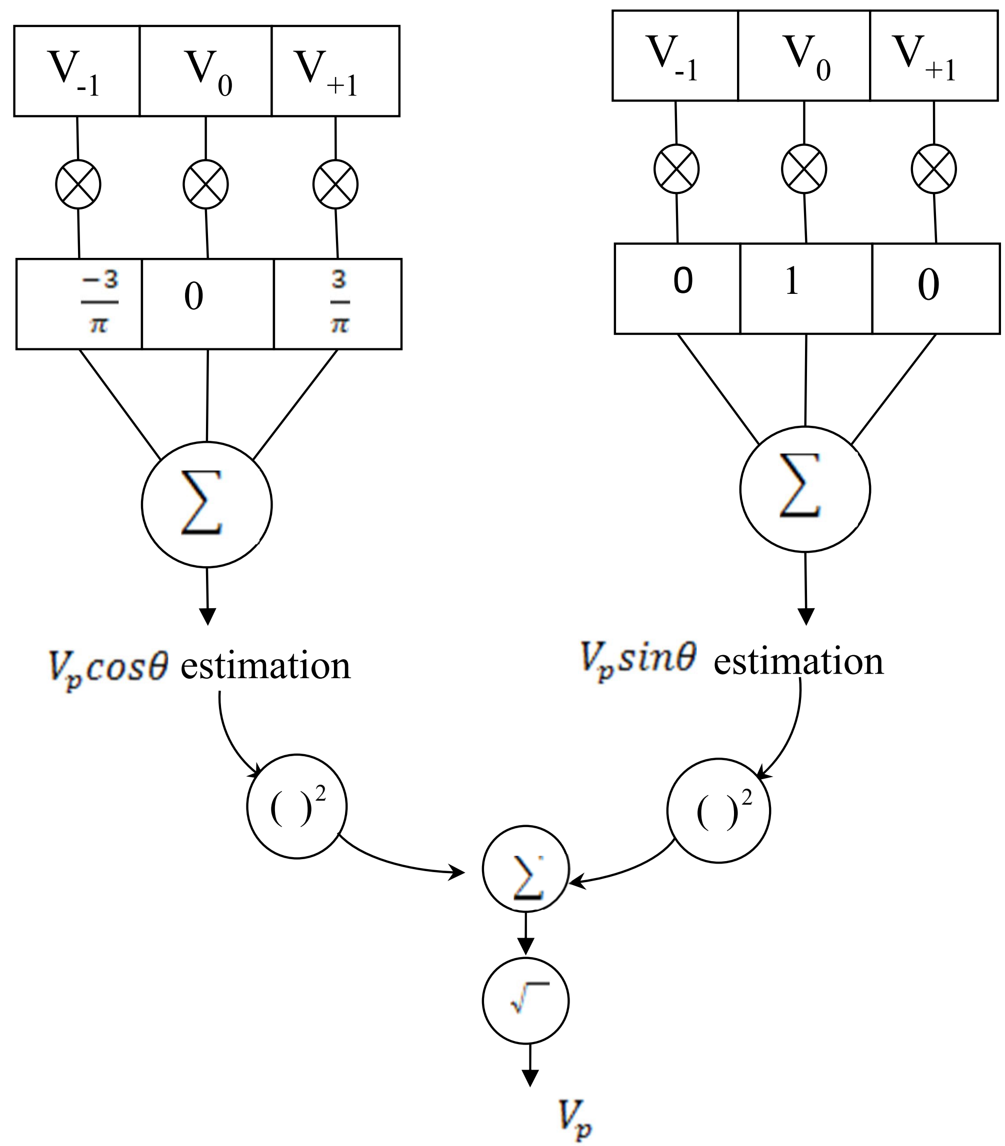
$$\begin{cases} f = 60 \rightarrow \omega_0 = 2\times\pi\times f = 2\times\pi\times 60 \\ n = 12 \rightarrow \Delta T = \dfrac{1}{12\times 60} = \dfrac{1}{720} \end{cases} \rightarrow$$

$$2\times\omega_0\times\Delta T = 2\times(2\times\pi\times 60)\times\frac{1}{720} = \frac{\pi}{3}$$

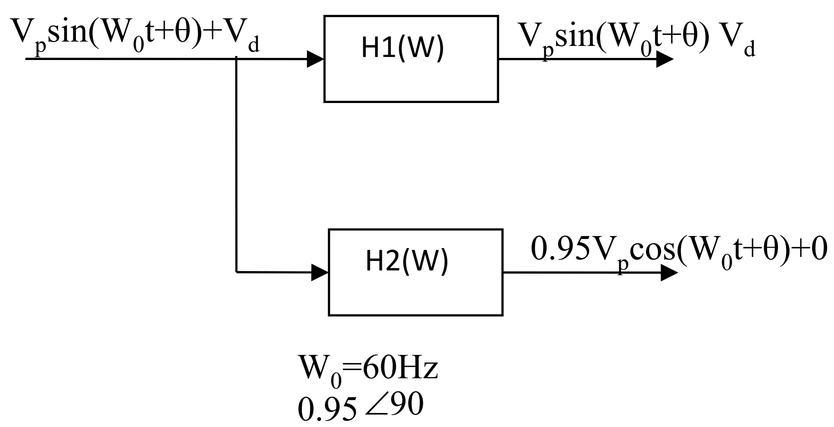
$$\frac{dV}{\omega_0 dt}\Big|_{t=0} = \frac{3}{\pi}\times(V_{+1} - V_{-1})$$

to estimate $V_p sin\theta$

$$V_0 = V_p \sin\theta$$



Mann& Morrison algorithm (short window)(with dc component)

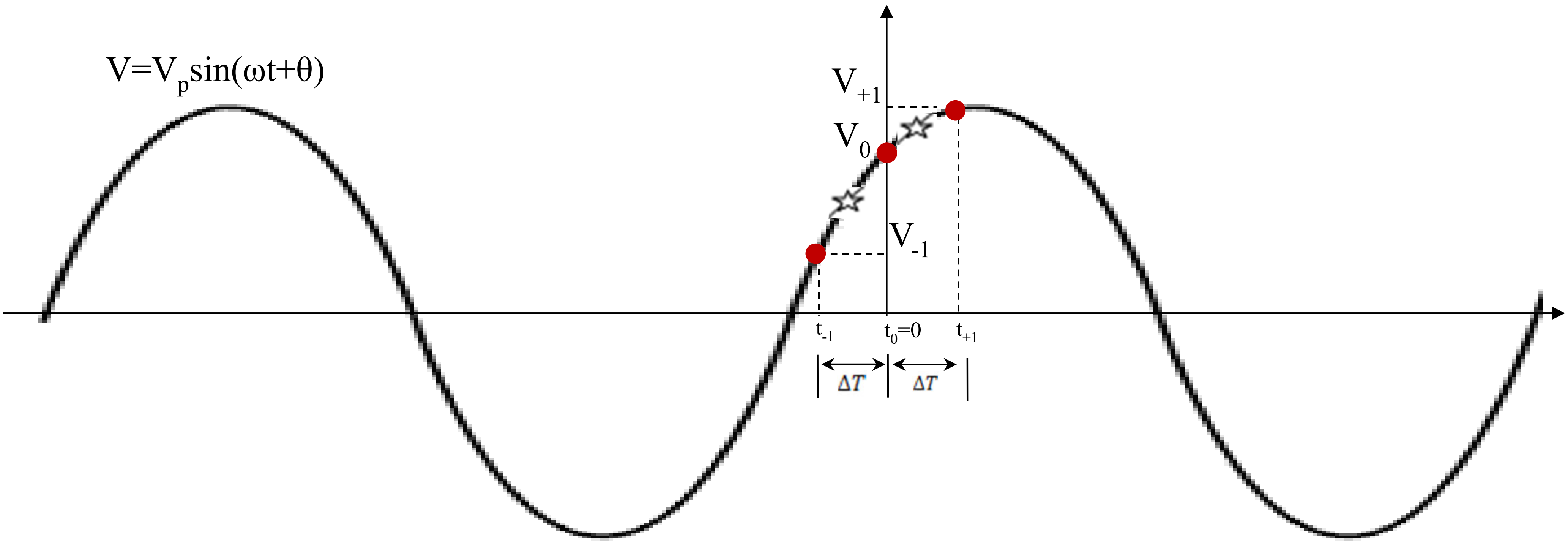


$$V_p^2 = [V_d + V_p \sin(\omega_0 t + \theta)]^2 + [0.955 \ V_p \cos^2(\omega_0 t + \theta)]^2$$

$$= V_p^2 \sin^2(\omega_0 t + \theta) + 2V_p V_d \sin(\omega_0 t + \theta) + V_d^2 + 0.955^2 V_p^2 \cos^2(\omega_0 t + \theta)$$

$$= 0.956 V_p^2 - 0.044 V_p^2 \cos(2\omega_0 t + 2\theta) + V_d^2 + 2V_p V_d \sin(\omega_0 t + \theta)$$

Prodar algorithm (short window) (eliminates the dc component)



$$V'|_{t=\frac{1}{2}\Delta T} = \frac{dV}{\omega_0 dt}|_{t=\frac{1}{2}\Delta T} = \frac{V_{+1}-V_0}{\omega_0 \Delta T} \qquad V'|_{t=-\frac{1}{2}\Delta T} = \frac{dV}{\omega_0 dt}|_{t=-\frac{1}{2}\Delta T} = \frac{V_0-V_{-1}}{\omega_0 \Delta T}$$

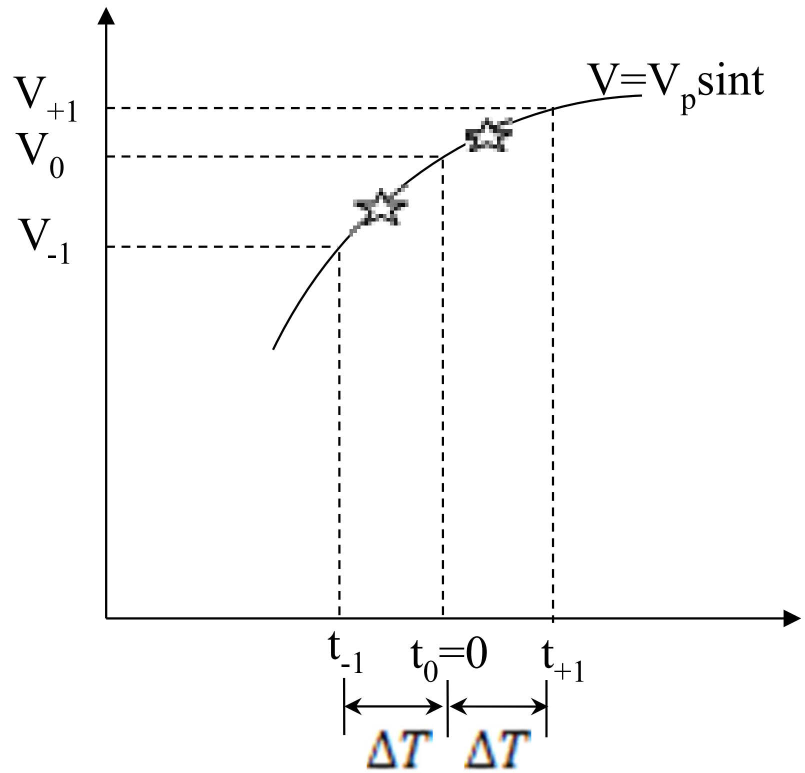
$$V''|_{t=0} = \frac{dV'}{\omega_0 dt}|_{t=0} = \frac{V'|_{t=\frac{1}{2}\Delta T} - V'|_{t=-\frac{1}{2}\Delta T}}{\omega_0 \Delta T} = \frac{(\frac{V_{+1}-V_0}{\omega_0 \Delta T}) - (\frac{V_0-V_{-1}}{\omega_0 \Delta T})}{\omega_0 \Delta T} = \frac{V_{+1}-2V_0+V_{-1}}{(\omega_0 \Delta T)^2}$$

$$\begin{cases} f = 60 \rightarrow \omega_0 = 2\times\pi\times f = 2\times\pi\times 60 \\ n = 12 \rightarrow \Delta T = \frac{1}{12\times 60} = \frac{1}{720} \end{cases} \rightarrow \omega_0 \times \Delta T = (2\times\pi\times 60)\times\frac{1}{720} = \frac{\pi}{6}$$

$\frac{-3}{\pi}$	0	$\frac{3}{\pi}$

$\frac{36}{\pi^2}$	$\frac{-72}{\pi^2}$	$\frac{36}{\pi^2}$

Prodar algorithm (short window) (eliminates the dc component)



$$V'|_{t=\frac{1}{2}\Delta T}=\frac{dV}{\omega_0 dt}|_{t=\frac{1}{2}\Delta T}=\frac{V_{+1}-V_0}{\omega_0\Delta T}$$

$$V'|_{t=-\frac{1}{2}\Delta T}=\frac{dV}{\omega_0 dt}|_{t=-\frac{1}{2}\Delta T}=\frac{V_0-V_{-1}}{\omega_0\Delta T}$$

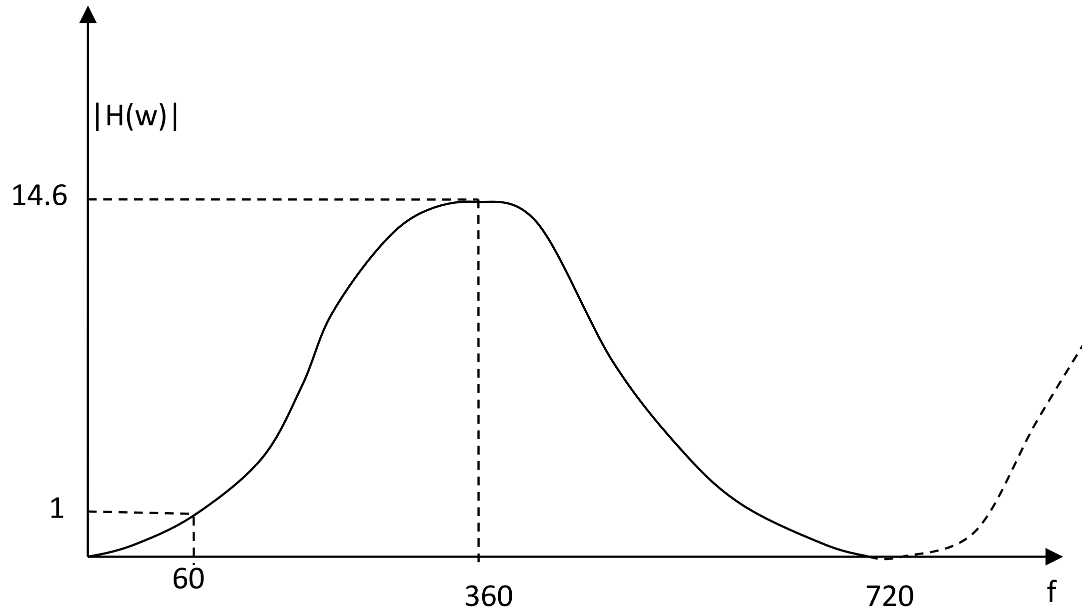
$$V''|_{t=0}=\frac{dV'}{\omega_0 dt}|_{t=0}=\frac{V'|_{t=\frac{1}{2}\Delta T}-V'|_{t=-\frac{1}{2}\Delta T}}{\omega_0\Delta T}=\frac{(\frac{V_{+1}-V_0}{\omega_0\Delta T})-(\frac{V_0-V_{-1}}{\omega_0\Delta T})}{\omega_0\Delta T}=\frac{V_{+1}-2V_0+V_{-1}}{(\omega_0\Delta T)^2}$$

$\frac{-3}{\pi}$	0	$\frac{3}{\pi}$

$$\begin{cases} f=60\rightarrow\omega_0=2\times\pi\times f=2\times\pi\times 60 \\ n=12\rightarrow\Delta T=\frac{1}{12\times 60}=\frac{1}{720}\end{cases}\rightarrow\omega_0\times\Delta T=(2\times\pi\times 60)\times\frac{1}{720}=\frac{\pi}{6}$$

$\frac{36}{\pi^2}$	$\frac{-72}{\pi^2}$	$\frac{36}{\pi^2}$

Prodar algorithm (short window)



$$f(Z) = \frac{36}{\pi^2} \times (Z^{-1} - 2Z^0 + Z^{+1})$$

$$H(\omega) = \frac{36}{\pi^2} \times (e^{-J\omega\Delta T} + 2 + e^{J\omega\Delta T})$$

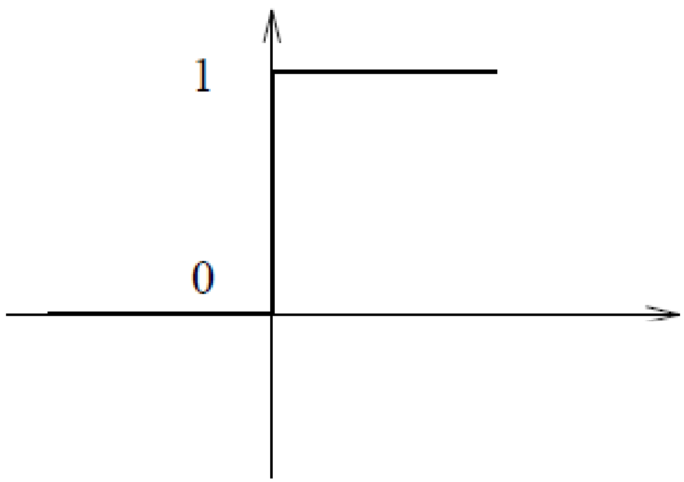
$$= \frac{36}{\pi^2} \times [\cos(\omega\Delta T) - j\sin(\omega\Delta T) - 2 + \cos(\omega\Delta T) - j\sin(\omega\Delta T)] = \frac{36 \times 2}{\pi^2}[\cos(\omega\Delta T) - 1]$$

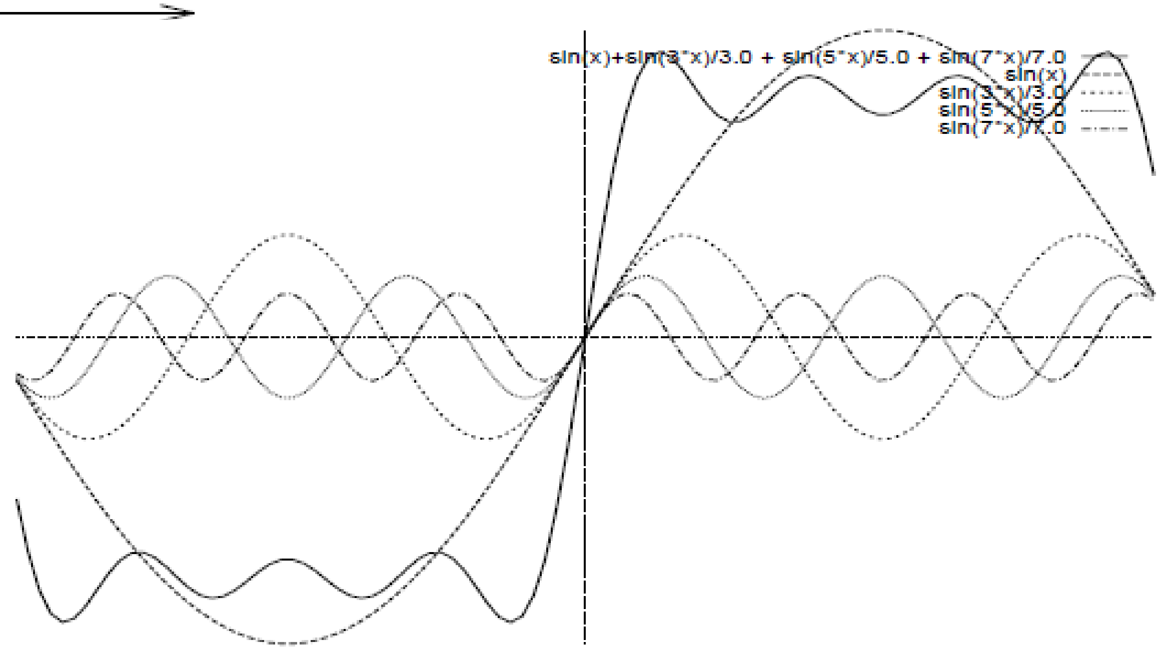
$$|H(\omega)| = \frac{36 \times 2}{\pi^2} \times (1 - \cos(\omega\Delta T))$$

$$\angle H(\omega) = 180^0$$

Fourier algorithm

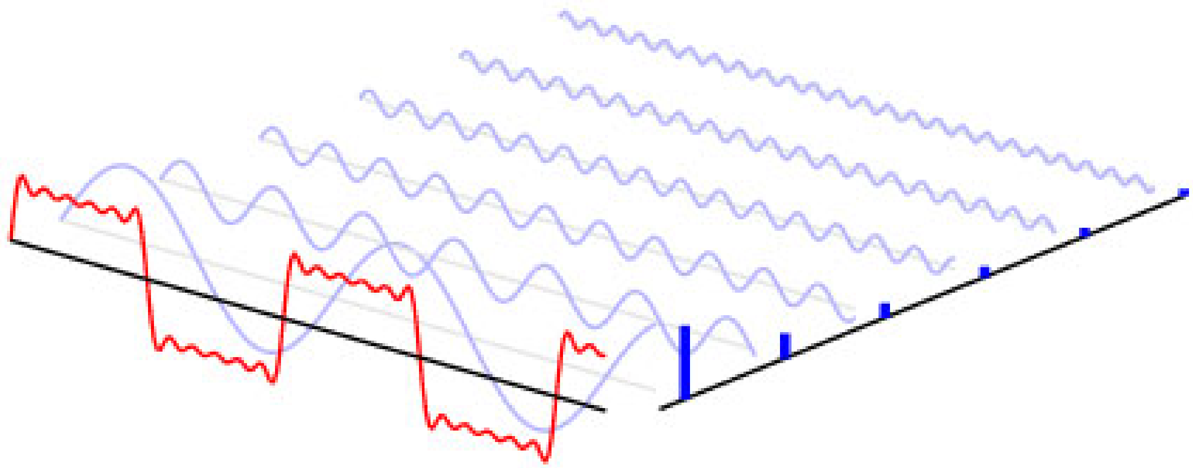
Fourier transform is a mathematical operation that decomposes a signal into its constituent frequencies





A step function is the sum of an infinite number of sine waves

Fourier algorithm



$$v(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + \sum_{n=1}^{\infty} b_n \sin(n\omega_0 t)$$

$$a_n = \frac{2}{T}\int_{t_0}^{t_0+T} v(t)\cos(n\omega_0 t)dt \qquad n = 0,1,\ldots \qquad a_1 = \frac{2}{N}\sum_{i=0}^{N-1} v_i \cos(\frac{2\pi i}{N}) = \frac{2}{N}\sum_{i=0}^{N-1} v_i W_{x,i}$$

$$b_n = \frac{2}{T}\int_{t_0}^{t_0+T} v(t)\sin(n\omega_0 t)dt \qquad n = 0,1,\ldots \qquad b_1 = \frac{2}{N}\sum_{i=0}^{N-1} v_i \sin(\frac{2\pi i}{N}) = \frac{2}{N}\sum_{i=0}^{N-1} v_i W_{x,i}$$

Fourier algorithm (full-cycle window)

$$v(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + \sum_{n=1}^{\infty} b_n \sin(n\omega_0 t)$$

$$a_n = \frac{2}{T}\int_{t_0}^{t_0+T} v(t)\cos(n\omega_0 t)dt \qquad n = 0,1,\ldots$$

$$b_n = \frac{2}{T}\int_{t_0}^{t_0+T} v(t)\sin(n\omega_0 t)dt \qquad n = 0,1,\ldots$$

Fourier algorithm (full-cycle window)

$$a_1 = \frac{2}{T}\int_{t_0}^{t_0+T} v(t)\cos(\omega_0 t)dt$$

$$a_1 = \frac{2}{N\Delta t}[v(t_0)\cos\omega_0 t_0 + v(t_1)\cos\omega_0 t_1 + \cdots + v(t_j)\cos\omega_0 t_i + \cdots + v(t_{N-1})\cos\omega_0 t_{N-1}]\Delta t$$

$$a_1 = \frac{2}{N}\sum_{i=0}^{N-1} v_i \cos(\frac{2\pi i}{N}) = \frac{2}{N}\sum_{i=0}^{N-1} v_i W_{x,i}$$

$$\begin{cases} W_{x,i} = \cos\omega_0 t_i = \cos\frac{2\pi}{T} i\Delta t \\ \Delta t = \frac{T}{N} \end{cases} \Longrightarrow \quad W_{x,i} = \cos\frac{2\pi}{N} i \qquad i = 0,1,\ldots,(N-1)$$

$$b_1 = \frac{2}{N}\sum_{i=0}^{N-1} v_i \sin(\frac{2\pi i}{N}) = \frac{2}{N}\sum_{i=0}^{N-1} v_i W_{x,i} \qquad W_{x,i} = \sin\frac{2\pi}{N} i \qquad i = 0,1,\ldots,(N-1)$$

Fourier algorithm (full-cycle window)

$$v_p \sin(\omega_0 t + \theta) = \underbrace{v_p \sin(\theta)}_{a_1} \cos(\omega_0 t) + \underbrace{v_p \cos(\theta)}_{b_1} \sin(\omega_0 t)$$

$$a_1 \cong \frac{2}{N} \sum_{i=0}^{N-1} v_i \cos(\frac{2\pi i}{N}) = \frac{2}{N} \sum_{i=0}^{N-1} v_i W_{x,i}$$

$$W_{x,i} = \cos \frac{2\pi}{N} i \qquad i = 0,1,\ldots,(N-1)$$

$$b_1 \cong \frac{2}{N} \sum_{i=0}^{N-1} v_i \sin(\frac{2\pi i}{N}) = \frac{2}{N} \sum_{i=0}^{N-1} v_i W_{x,i}$$

$$W_{x,i} = \sin \frac{2\pi}{N} i \qquad i = 0,1,\ldots,(N-1)$$

$$v(t) = v_p = \sqrt{b_1^2 + a_1^2} = \sqrt{v_p \cos(\theta)^2 + v_p \sin(\theta)^2}$$

$$\tan = \frac{b_1}{a_1} = \frac{v_p \sin(\theta)}{v_p \cos(\theta)}$$

Fourier algorithm (full-cycle window)

$$v_p \sin(\omega_0 t + \theta) = \underbrace{v_p \sin(\theta)}_{a_1} \cos(\omega_0 t) + \underbrace{v_p \cos(\theta)}_{b_1} \sin(\omega_0 t)$$

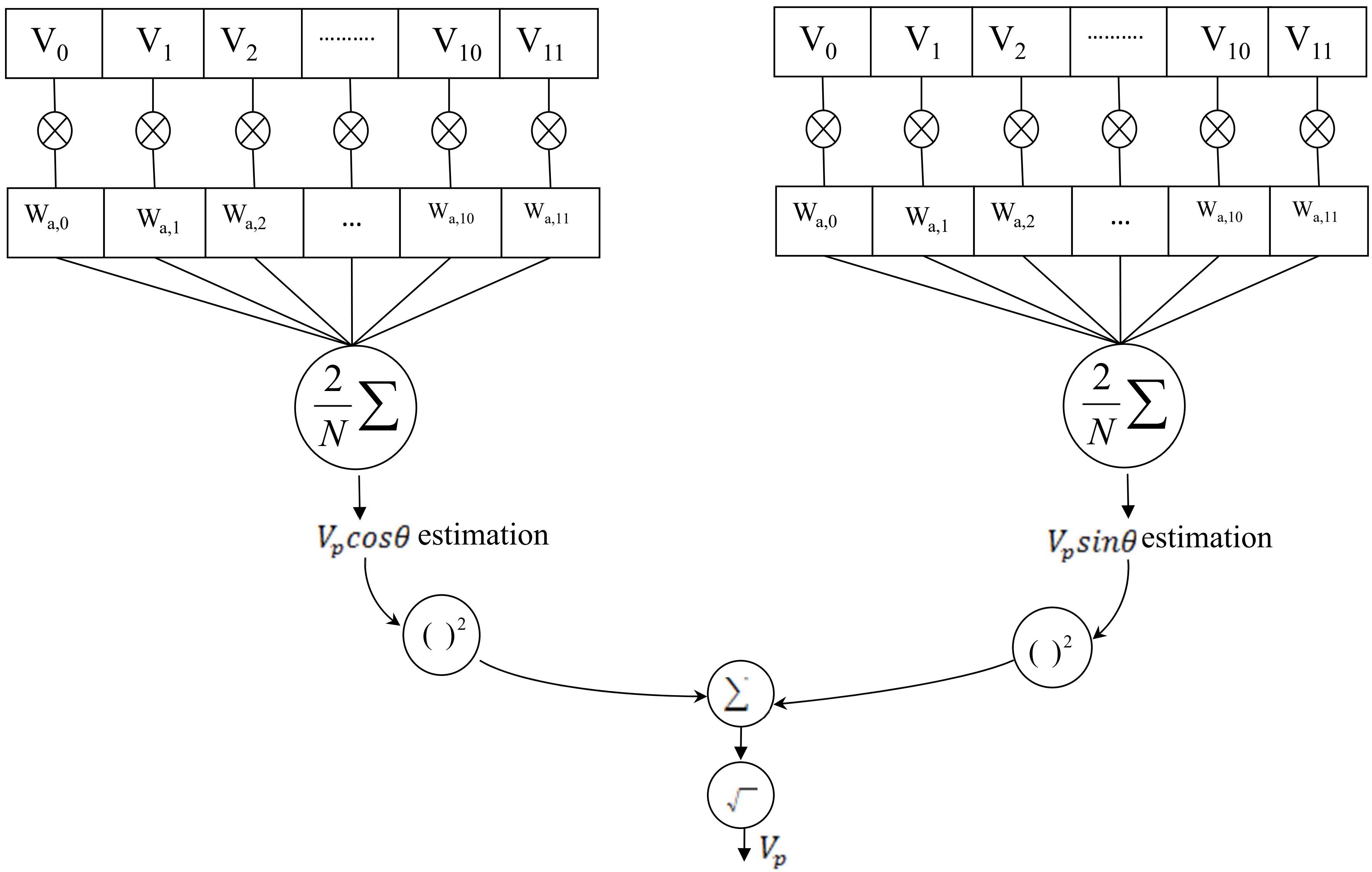
$$v(t) = v_p = \sqrt{b_1^2 + a_1^2} = \sqrt{v_p \cos(\theta)^2 + v_p \sin(\theta)^2}$$

$$\tan = \frac{b_1}{a_1} = \frac{v_p \sin(\theta)}{v_p \cos(\theta)}$$

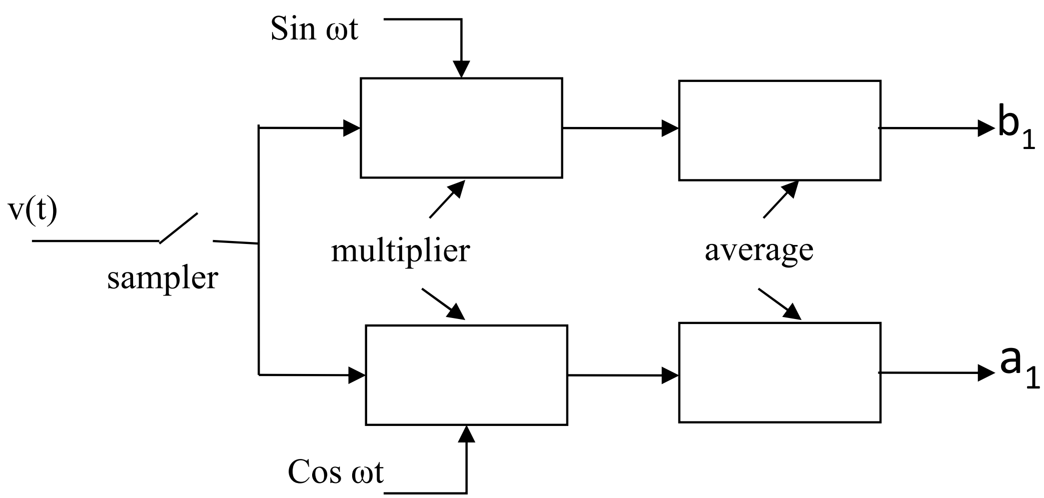
Fourier algorithm (full-cycle window)

$$W_{x,i} = \sin\frac{2\pi}{N}i \qquad i = 0,1,\ldots,(N-1)$$

$$W_{a,i} = \cos\frac{2\pi}{N}i \qquad i = 0,1,\ldots,(N-1)$$



Fourier algorithm (full-cycle window)



Fourier algorithm (full-cycle window)

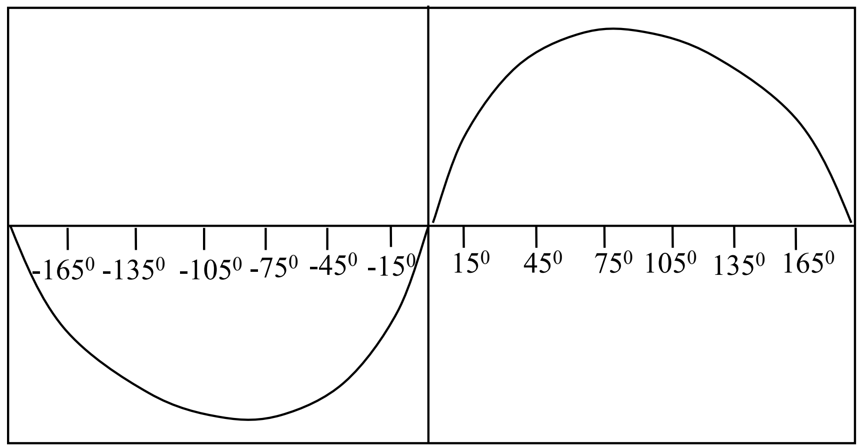
Weighting factors for N=12 samples per fundamental cycle

i	$W_{a1,i}$	i	$W_{b1,i}$
0	1	0,6	0
1,11	$\sqrt{3}/2$	1,5	½
2,10	½	2,4	$\sqrt{3}/2$
3,9	0	3	1
4,8	$-\sqrt{3}/2$	7,11	-1/2
5,7	-1/2	8,10	$-\sqrt{3}/2$
6	-1	9	-1

$$a_1 = \frac{1}{6}[v_0 + \frac{1}{2}(v_2 - v_5 + v_{10} - v_7) + \frac{\sqrt{3}}{2}(v_1 - v_4 + v_{11} - v_8)]$$

$$b_1 = \frac{1}{6}[v_3 - v_9 + \frac{1}{2}(v_1 - v_7 + v_5 - v_{11}) + \frac{\sqrt{3}}{2}(v_2 - v_8 + v_4 - v_{10})]$$

Fourier algorithm (full-cycle window)



0

$$\frac{-11\omega_0\Delta T}{2},\frac{-9\omega_0\Delta T}{2},\frac{-7\omega_0\Delta T}{2},\frac{-5\omega_0\Delta T}{2},\frac{-3\omega_0\Delta T}{2},\frac{-\omega_0\Delta T}{2} \;\vdots\; \frac{\omega_0\Delta T}{2},\frac{3\omega_0\Delta T}{2},\frac{5\omega_0\Delta T}{2},\frac{7\omega_0\Delta T}{2},\frac{9\omega_0\Delta T}{2},\frac{11\omega_0\Delta T}{2}$$

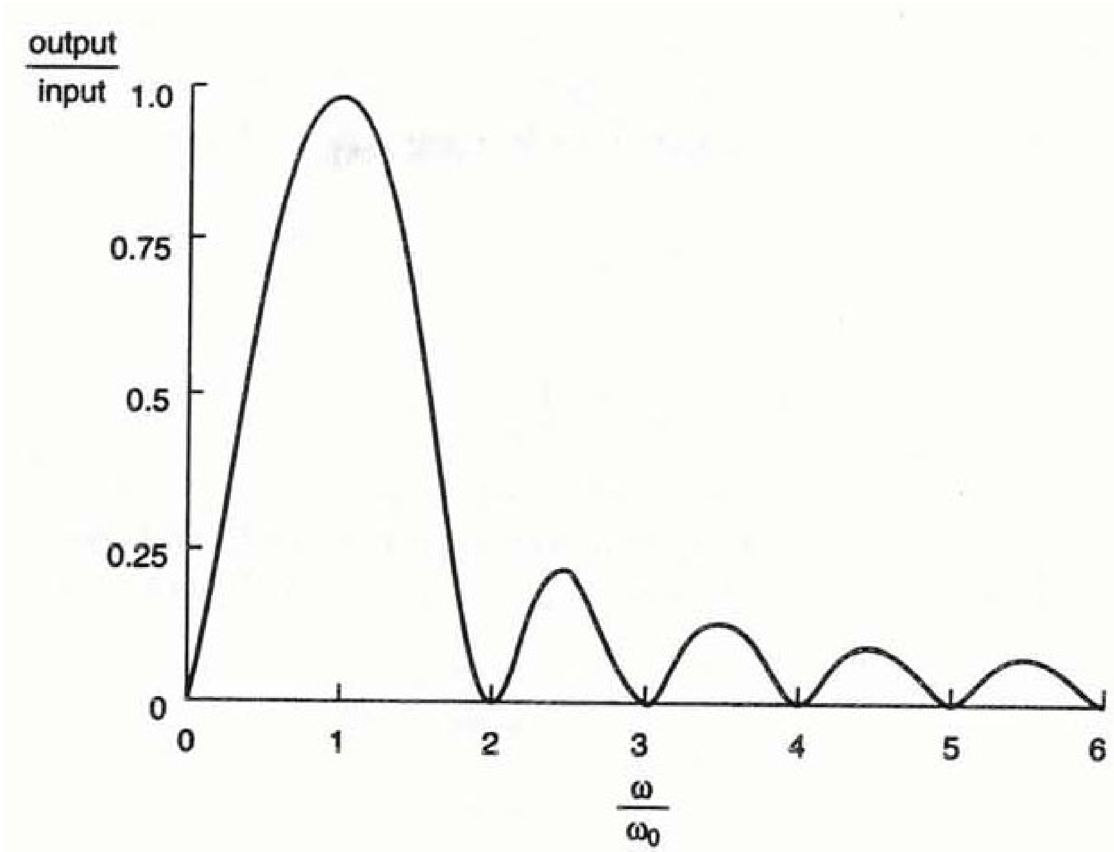
$$H(Z)=\frac{2}{12}[\sin(\frac{-11}{2}\omega_0\Delta T)Z^{-5.5}+\sin(\frac{-9}{2}\omega_0\Delta T)Z^{-4.5}+\sin(\frac{-7}{2}\omega_0\Delta T)Z^{-3.5}+\cdots] \quad Z=e^{\omega\Delta T}$$

$$\omega_0\Delta T=30^0$$

$$=\frac{1}{6}[\sin(\frac{11}{2}\omega_0\Delta T)(Z^{5.5}-Z^{-5.5})+\sin(\frac{9}{2}\omega_0\Delta T)(Z^{4.5}-Z^{-4.5})+\cdots]$$

$$=\frac{1}{6}[\sin(165^0)(2j\sin(5.5\omega\Delta T)+\sin(135^0)(2j\sin(4.5\omega\Delta T)+\cdots+\sin(15^0)(2j\sin(0.5\omega\Delta T)]$$

Fourier algorithm (full-cycle window)



Frequency response of the full-cycle window Fourier algorithm

Fourier algorithm (half-cycle window)

$$a_{1,\frac{T}{2}} = \frac{2}{\frac{T}{2}} \int_{t_0}^{t_0+\frac{T}{2}} v(t)\cos(\omega_0 t)dt$$

$$b_{1,\frac{T}{2}} = \frac{2}{\frac{T}{2}} \int_{t_0}^{t_0+\frac{T}{2}} v(t)\sin(\omega_0 t)dt$$

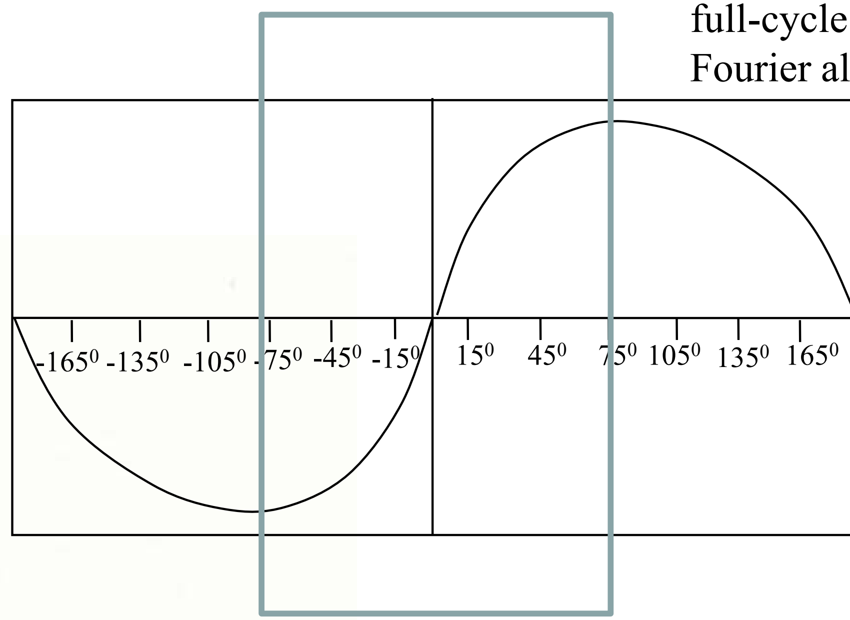
$$a_{1,\frac{T}{2}} \cong \frac{4}{N}\sum_{i=0}^{\frac{N}{2}-1} v_i \cos(\frac{2\pi i}{N})$$

$$b_{1,\frac{T}{2}} \cong \frac{4}{N}\sum_{i=0}^{\frac{N}{2}-1} v_i \sin(\frac{2\pi i}{N})$$

$$a_1 = \frac{1}{3}[v_0 + \frac{1}{2}(v_2 - v_5) + \frac{\sqrt{3}}{2}(v_1 - v_4)]$$

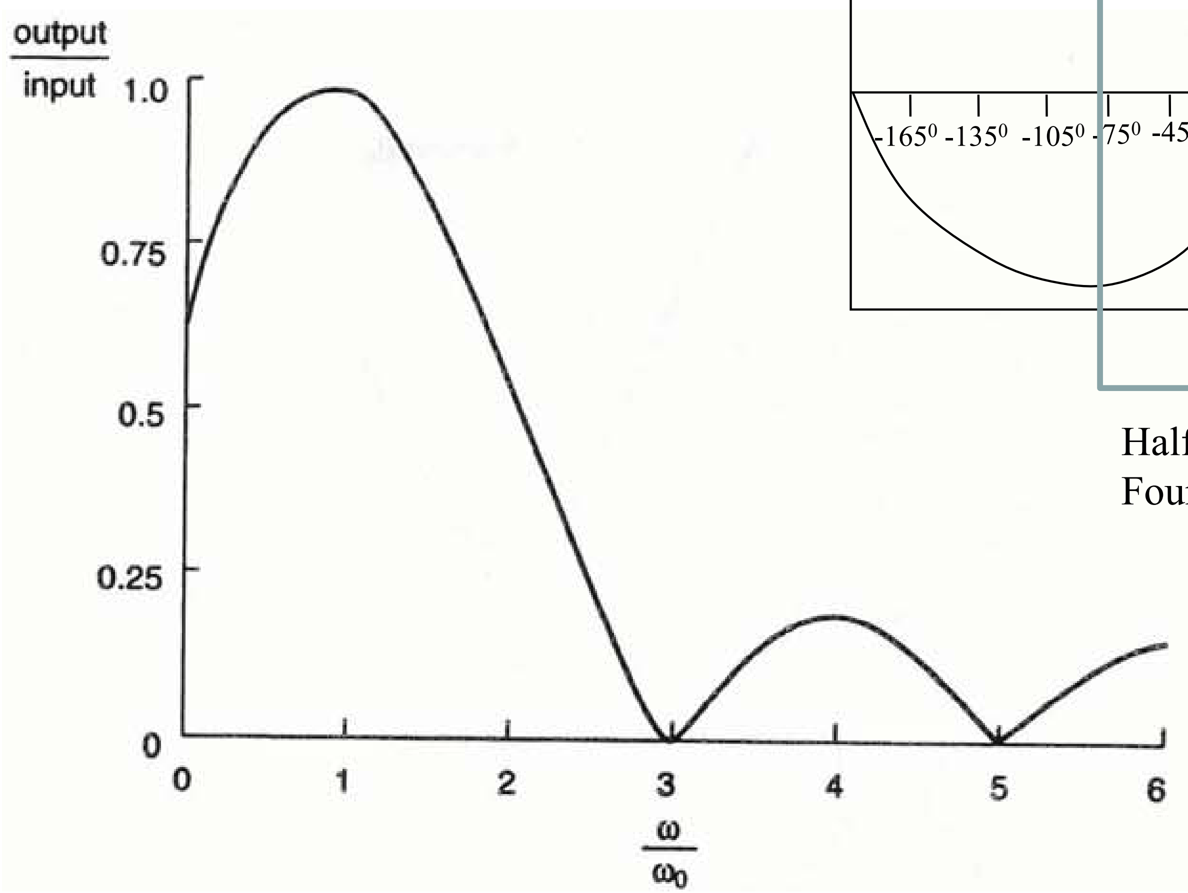
$$b_1 = \frac{1}{3}[v_3 + \frac{1}{2}(v_1 + v_5) + \frac{\sqrt{3}}{2}(v_2 + v_4)]$$

Fourier algorithm (half-cycle window)



full-cycle window
Fourier algorithm

Half-cycle window
Fourier algorithm



Frequency response of the half-cycle window Fourier algorithm

Least squares based algorithm

$$\begin{bmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 5/4 \\ -1/4 \\ 3/4 \end{bmatrix}$$

from first row $X_1=5/4$
from second row $X_2=6/4$
third row $6/4=3/4$

Least squares based algorithm

$$A\ x = b$$

A more reasonable approach to the “solution” of such equation is to recognize that there is an error and write

$$b = A\ x + e$$

$$e = b - Ax$$

$$\begin{aligned} e^T e &= (b - Ax)^T (b - Ax) \\ &= (b^T - x^T A^T)(b - A) \\ &= x^T A^T A\ x - x^T A^T b - b^T A\ x + b^T b \end{aligned}$$

The x that minimizes $e^T e$ can be obtained by taking the partial derivatives with respect to the components of x and equating to zero

pseudo inverse

$$\hat{X} = (A^T A)^{-1} A^T b$$

Least squares based algorithm

Let $\mathbf{x}$ and $\mathbf{y}$ be vectors of orders n and m respectively:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$$\frac{\partial \mathbf{y}}{\partial \mathbf{x}} \stackrel{\text{def}}{=} \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_2}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_1} \\ \frac{\partial y_1}{\partial x_2} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial x_n} & \frac{\partial y_2}{\partial x_n} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

$\mathbf{y}$	$\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$
$\mathbf{A}\mathbf{x}$	$\mathbf{A}^T$
$\mathbf{x}^T\mathbf{A}$	$\mathbf{A}$
$\mathbf{x}^T\mathbf{x}$	$2\mathbf{x}$
$\mathbf{x}^T\mathbf{A}\mathbf{x}$	$\mathbf{A}\mathbf{x} + \mathbf{A}^T\mathbf{x}$

Least squares based algorithm

$$\hat{X} = (A^T A)^{-1} A^T b$$

A mnemonic for equation is to multiply the equation on both sides by A^T and then multiply by the inverse of the square matrix ($A^T A$)

$$\underbrace{A}_{m \times n} \underbrace{\hat{X}}_{n \times 1} = \underbrace{b}_{m \times 1}$$

$$\underbrace{\underbrace{A^T}_{n \times m} \underbrace{A}_{m \times n}}_{n \times n} \underbrace{\hat{X}}_{n \times 1} = \underbrace{A^T}_{n \times m} \underbrace{b}_{m \times 1}$$

$$\hat{X} = (A^T A)^{-1} A^T b$$

Least squares based algorithm

$$\begin{bmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3/4 \\ -1/4 \\ 3/4 \end{bmatrix}$$

$$A^T A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

$$(A^T A)^{-1} = \frac{1}{3} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

$$A^T b = \begin{bmatrix} 1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 5/4 \\ -1/4 \\ 3/4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\hat{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Least squares based algorithm

$$V(t) = V_p \sin(\omega_0 t + \theta)$$

$$v(-\Delta T) = V_p \sin(-\omega_0 \Delta T + \theta) = \sin(-\omega_0 \Delta T)(V_p \cos\theta) + \cos(-\omega_0 \Delta T)(V_p \sin\theta)$$

$$v(0) = V_p \sin(\omega_0 \times 0 + \theta) = \sin(0)(V_p \cos\theta) + \cos(0)(V_p \sin\theta)$$

$$v(\Delta T) = V_p \sin(\omega_0 \Delta T + \theta) = \sin(\omega_0 \Delta T)(V_p \cos\theta) + \cos(\omega_0 \Delta T)(V_p \sin\theta)$$

$$\begin{bmatrix} \sin(-\omega_0 \Delta T) & \cos(-\omega_0 \Delta T) \\ \sin(0) & \cos(0) \\ \sin(\omega_0 \Delta T) & \cos(\omega_0 \Delta T) \end{bmatrix} \begin{bmatrix} V_p \cos\theta \\ V_p \sin\theta \end{bmatrix} = \begin{bmatrix} v_{-1} \\ v_0 \\ v_{+1} \end{bmatrix}$$

Least squares based algorithm

$$A^t = [A^T A]^{-1}[A]^T$$

$$\begin{cases} f = 60 \to \omega_0 = 2\times\pi\times f = 2\times\pi\times 60 \\ n = 12 \to \Delta T = \dfrac{1}{12\times 60} = \dfrac{1}{720} \end{cases} \to \omega_0 \times \Delta T = (2\times\pi\times 60)\times\frac{1}{720} = \frac{\pi}{6}$$

$$A = \begin{bmatrix} \sin(-\omega_0\Delta T) & \cos(-\omega_0\Delta T) \\ \sin(0) & \cos(0) \\ \sin(\omega_0\Delta T) & \cos(\omega_0\Delta T) \end{bmatrix} = \begin{bmatrix} \frac{-1}{2} & \frac{\sqrt{3}}{2} \\ 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix}$$

$$A^T A = \begin{bmatrix} \frac{-1}{2} & 0 & \frac{1}{2} \\ \frac{\sqrt{3}}{2} & 1 & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \frac{-1}{2} & \frac{\sqrt{3}}{2} \\ 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{5}{2} \end{bmatrix}$$

$$A^t = \begin{bmatrix} 2 & 0 \\ 0 & \frac{2}{5} \end{bmatrix} \begin{bmatrix} \frac{-1}{2} & 0 & \frac{1}{2} \\ \frac{\sqrt{3}}{2} & 1 & \frac{\sqrt{3}}{2} \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ \frac{\sqrt{3}}{5} & \frac{2}{5} & \frac{\sqrt{3}}{5} \end{bmatrix}$$

First filter

$$\begin{bmatrix} V_p \cos\theta \\ V_p \sin\theta \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ \frac{\sqrt{3}}{5} & \frac{2}{5} & \frac{\sqrt{3}}{5} \end{bmatrix} \begin{bmatrix} v_{-1} \\ v_0 \\ v_{+1} \end{bmatrix}$$

Second filter

Least squares based algorithm

$$\begin{bmatrix} \sin(-11\frac{\omega_0 \Delta T}{2}) & \cos(-11\frac{\omega_0 \Delta T}{2}) \\ \sin(-9\frac{\omega_0 \Delta T}{2}) & \cos(-9\frac{\omega_0 \Delta T}{2}) \\ \sin(-7\frac{\omega_0 \Delta T}{2}) & \cos(-7\frac{\omega_0 \Delta T}{2}) \\ \sin(-5\frac{\omega_0 \Delta T}{2}) & \cos(-5\frac{\omega_0 \Delta T}{2}) \\ \sin(-3\frac{\omega_0 \Delta T}{2}) & \cos(-3\frac{\omega_0 \Delta T}{2}) \\ \sin(-\frac{\omega_0 \Delta T}{2}) & \cos(-\frac{\omega_0 \Delta T}{2}) \\ \sin(\frac{\omega_0 \Delta T}{2}) & \cos(\frac{\omega_0 \Delta T}{2}) \\ \sin(3\frac{\omega_0 \Delta T}{2}) & \cos(3\frac{\omega_0 \Delta T}{2}) \\ \sin(5\frac{\omega_0 \Delta T}{2}) & \cos(5\frac{\omega_0 \Delta T}{2}) \\ \sin(7\frac{\omega_0 \Delta T}{2}) & \cos(7\frac{\omega_0 \Delta T}{2}) \\ \sin(9\frac{\omega_0 \Delta T}{2}) & \cos(9\frac{\omega_0 \Delta T}{2}) \\ \sin(11\frac{\omega_0 \Delta T}{2}) & \cos(11\frac{\omega_0 \Delta T}{2}) \end{bmatrix} \begin{bmatrix} V_p \cos\theta \\ V_p \sin\theta \end{bmatrix} = \begin{bmatrix} v_{-6} \\ v_{-5} \\ v_{-4} \\ v_{-3} \\ v_{-2} \\ v_{-1} \\ v_{+1} \\ v_{+2} \\ v_{+3} \\ v_{+4} \\ v_{+5} \\ v_{+6} \end{bmatrix}$$

A X b

Least squares based algorithm

$$A^T A = \frac{1}{6} \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A^t = \frac{1}{6} \times \begin{bmatrix} \sin(-165^0) & \sin(-135^0) & \cdots & \sin(135^0) & \sin(165^0) \\ \cos(-165^0) & \cos(-135^0) & \cdots & \cos(135^0) & \cos(165^0) \end{bmatrix}$$

Least squares based algorithm

$$V(t) = V_p \sin(\omega_0 t + \theta) + v_d$$

$$\begin{bmatrix} \sin(-2\omega_0 \Delta T) & \cos(-2\omega_0 \Delta T) & 1 \\ \sin(-\omega_0 \Delta T) & \cos(-\omega_0 \Delta T) & 1 \\ \sin(0) & \cos(0) & 1 \\ \sin(\omega_0 \Delta T) & \cos(\omega_0 \Delta T) & 1 \\ \sin(2\omega_0 \Delta T) & \cos(2\omega_0 \Delta T) & 1 \end{bmatrix} \begin{bmatrix} V_p \cos\theta \\ V_p \sin\theta \\ v_d \end{bmatrix} = \begin{bmatrix} v_{-1} \\ v_{-2} \\ v_0 \\ v_{+1} \\ v_{+2} \end{bmatrix}$$

Least squares based algorithm

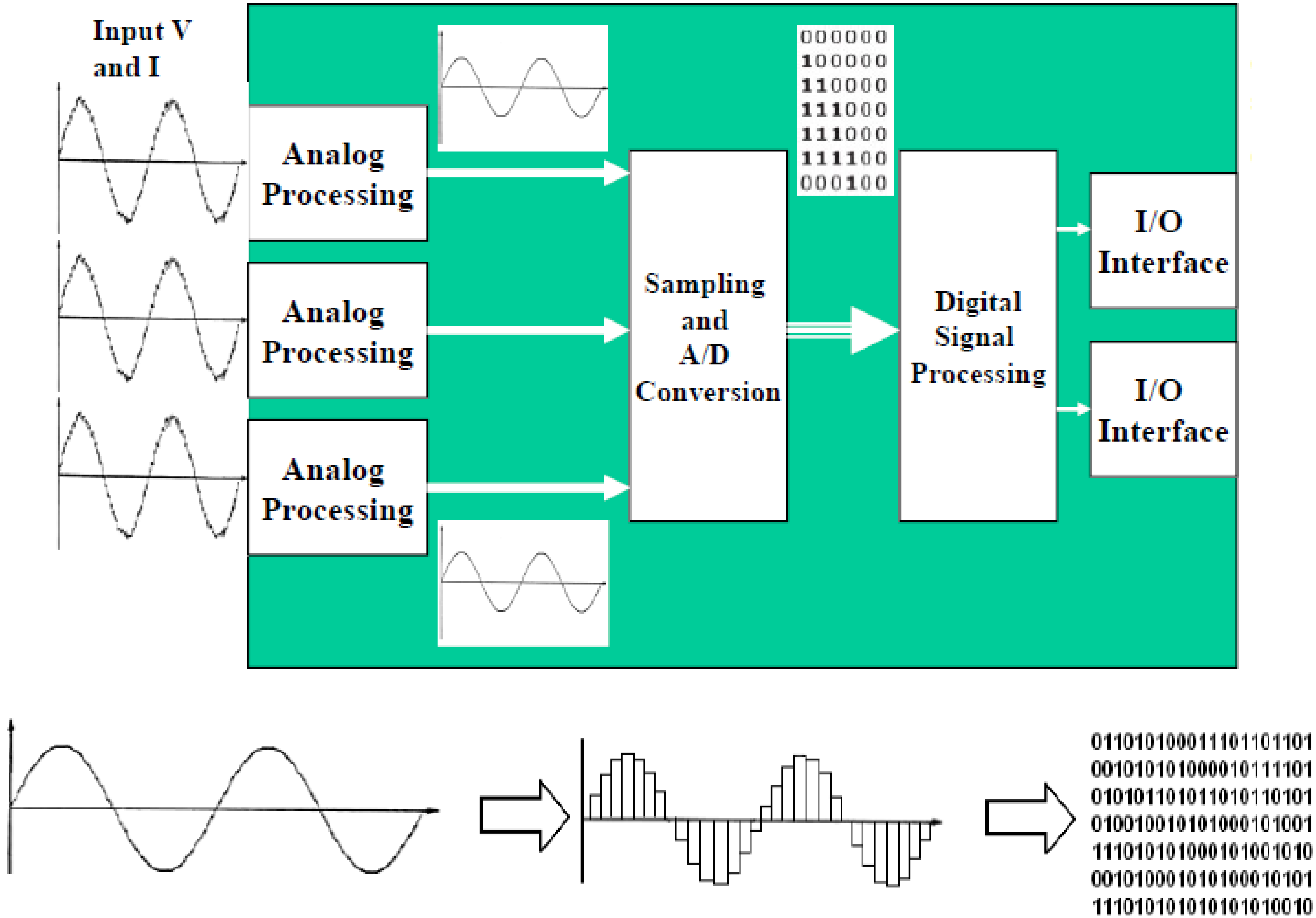
$$V(t) = V_p \sin(\omega_0 t + \theta_1) + V_{p2}\sin(2\omega_0 t + \theta_2) + v_d e^{\frac{-t}{\tau}}$$

$$e^{\frac{-t}{\tau}} = 1 - \tfrac{t}{\tau} + \tfrac{1}{2!}\tfrac{t^2}{\tau^2} - \tfrac{1}{3!}\tfrac{t^3}{\tau^3}$$

$$V(t) = \sin(\omega_0 t)(V_p \cos\theta_1) + \cos(\omega_0 t)(V_p \sin\theta_1) + \sin(2\omega_0 t)(V_{p2}\cos\theta_2) + \cos(2\omega_0 t)(V_{p2}\sin\theta_2)$$

$$+ v_d + t(\frac{-v_d}{\tau})$$

$$\begin{bmatrix} \sin(-4\omega_0\Delta T) & \sin(-4\omega_0\Delta T) & \sin(-4(2\omega_0)\Delta T) & \cos(-4(2\omega_0)\Delta T) & 1 & -4\Delta T \\ \sin(-3\omega_0\Delta T) & \cos(-3\omega_0\Delta T) & \sin(-3(2\omega_0)\Delta T) & \cos(-3(2\omega_0)\Delta T) & 1 & -3\Delta T \\ \sin(-2\omega_0\Delta T) & \cos(-2\omega_0\Delta T) & \sin(-2(2\omega_0)\Delta T) & \cos(-2(2\omega_0)\Delta T) & 1 & -2\Delta T \\ \sin(-1\omega_0\Delta T) & \cos(-1\omega_0\Delta T) & \sin(-1(2\omega_0)\Delta T) & \cos(-1(2\omega_0)\Delta T) & 1 & -1\Delta T \\ \sin(0\omega_0\Delta T) & \cos(0\omega_0\Delta T) & \sin(0(2\omega_0)\Delta T) & \cos(0(2\omega_0)\Delta T) & 1 & 0\Delta T \\ \sin(1\omega_0\Delta T) & \cos(1\omega_0\Delta T) & \sin(1(2\omega_0)\Delta T) & \cos(1(2\omega_0)\Delta T) & 1 & 1\Delta T \\ \sin(2\omega_0\Delta T) & \cos(2\omega_0\Delta T) & \sin(2(2\omega_0)\Delta T) & \cos(2(2\omega_0)\Delta T) & 1 & 2\Delta T \\ \sin(3\omega_0\Delta T) & \cos(3\omega_0\Delta T) & \sin(3(2\omega_0)\Delta T) & \cos(3(2\omega_0)\Delta T) & 1 & 3\Delta T \\ \sin(4\omega_0\Delta T) & \cos(4\omega_0\Delta T) & \sin(4(2\omega_0)\Delta T) & \cos(4(2\omega_0)\Delta T) & 1 & 4\Delta T \end{bmatrix} \begin{bmatrix} V_p \cos\theta_1 \\ V_p \sin\theta_1 \\ V_{p2}\cos\theta_2 \\ V_{p2}\cos\theta_2 \\ v_d \\ \frac{-v_d}{\tau} \end{bmatrix} = \begin{bmatrix} v_{-4} \\ v_{-3} \\ v_{-2} \\ v_{-1} \\ v_0 \\ v_{+1} \\ v_{+2} \\ v_{+3} \\ v_{+4} \end{bmatrix}$$

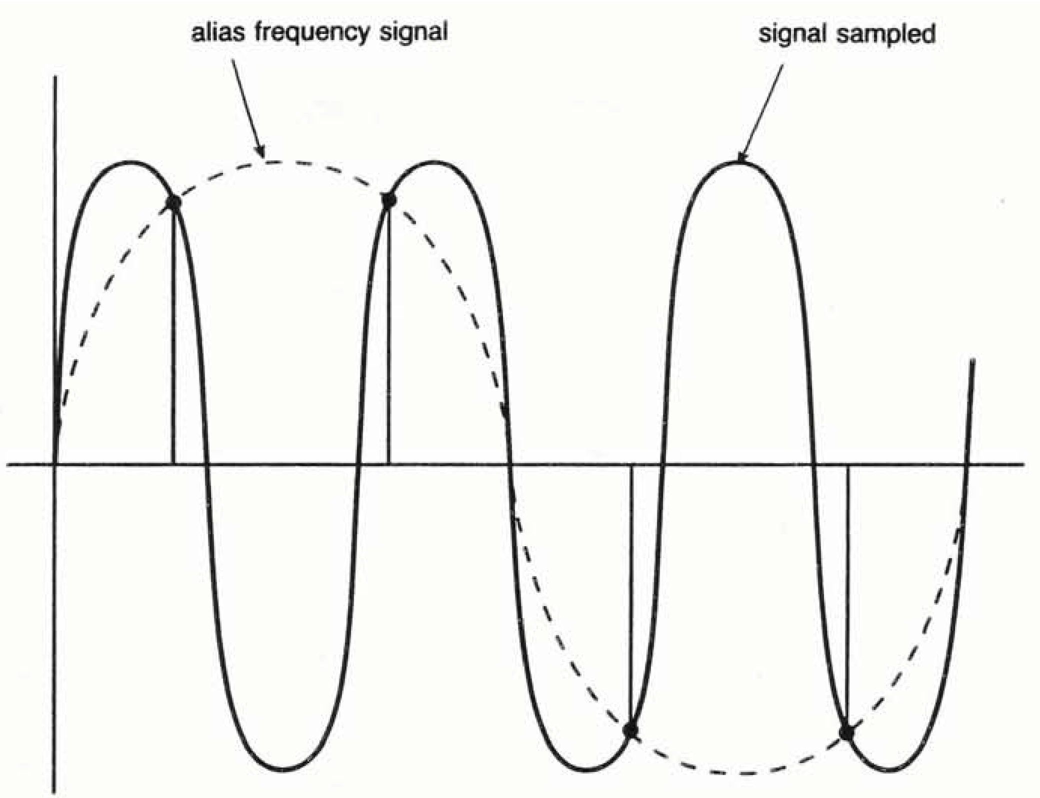


Input V
and I
Analog
Processing
Analog
Processing
Analog
Processing
Sampling
and
A/D
Conversion
000000
100000
110000
111000
111000
111100
000100
Digital
Signal
Processing
I/O
Interface
I/O
Interface
011010100011101101101
001010101000010111101
010101101011010110101
010010010101000101001
111010101000101001010
001010001010100010101
111010101010101010010

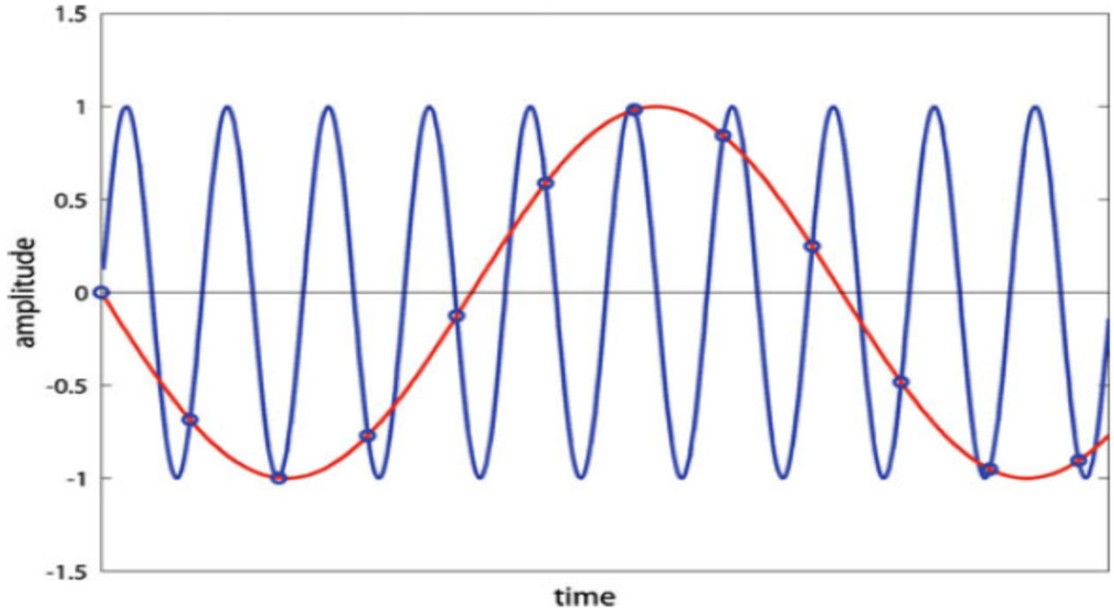
Antialiasing filter

The aliasing is a phenomenon that happens when a signal appears to be what is not really is

The *Nyquist Sampling Theorem* states that to avoid aliasing occurring in the sampling of a signal the sampling rate should be greater than or equal to twice the highest frequency present in the signal. This is referred to as the *Nyquist sampling rate*.



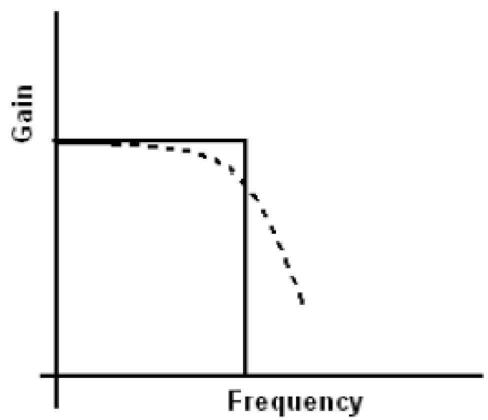
If sampling rate does not satisfy the Nyquist sampling theorem a low-frequency component that does not actually exist in the original signal would be present within the sampled signal. The figure gives an illustration of the aliasing phenomenon, in which the dotted line represents a low frequency that does not actually exist in the original signal



Misrepresentation of a signal due to low sampling rate

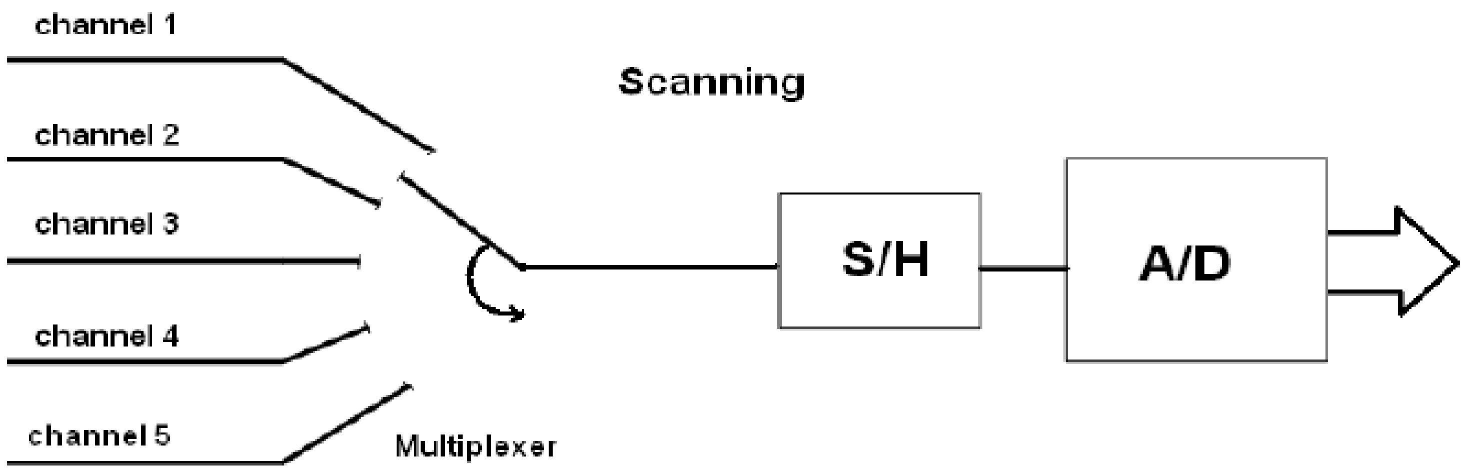
Low-Pass Filtering

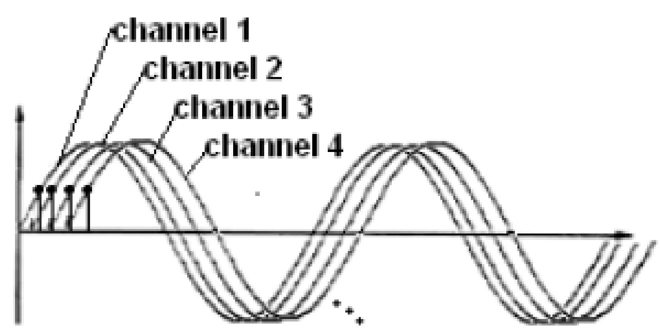
- Antialiasing filter



Scanning

- One sample and hold amplifier + one A/D converter

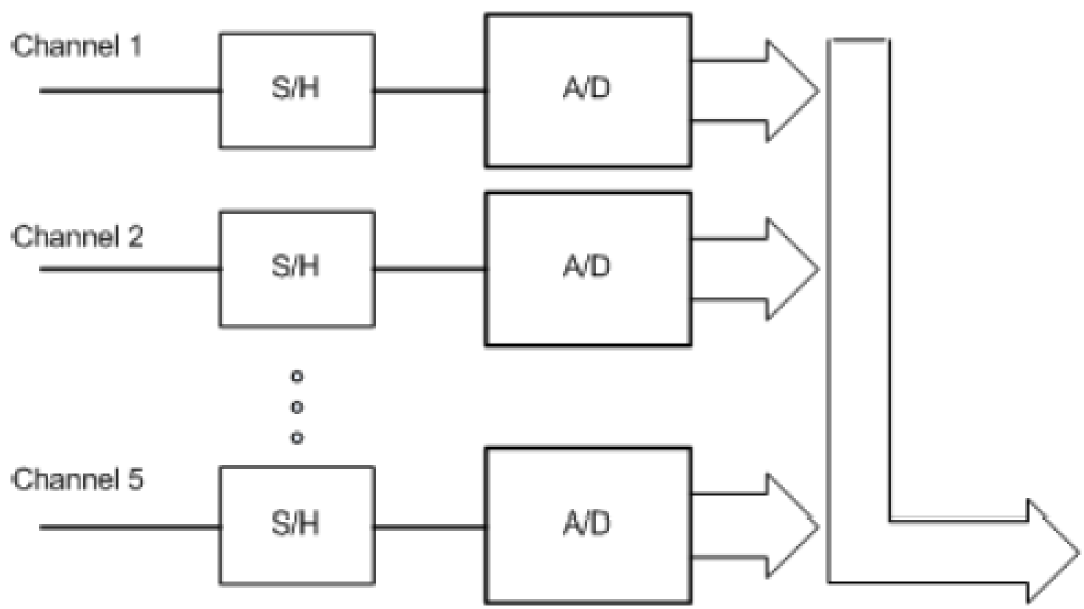




A sample and hold circuit is employed at the input before the ADC to hold the input value steady while the conversion is taking place.

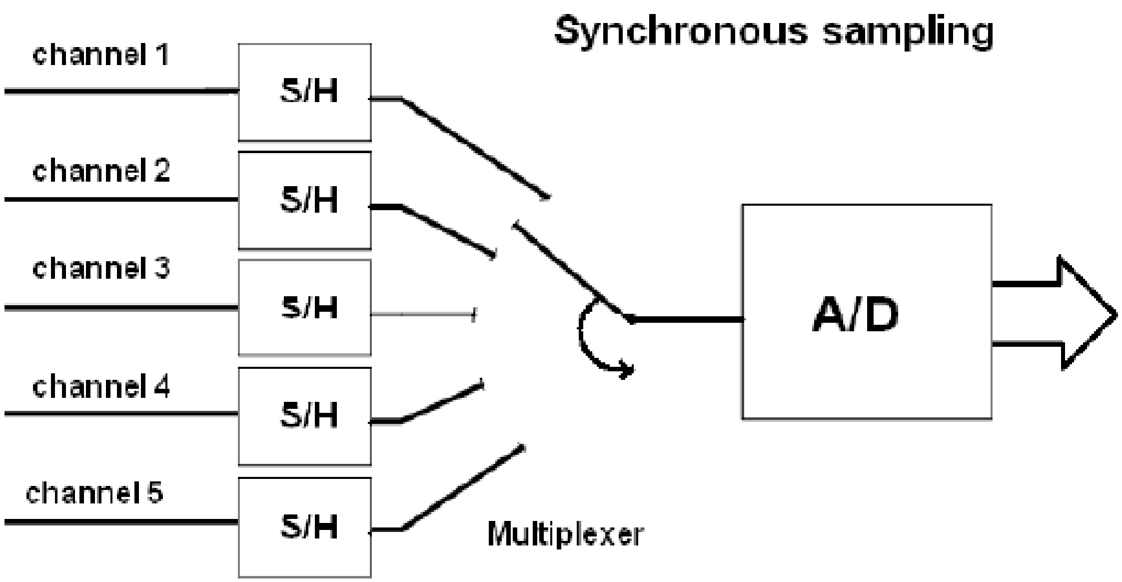
Synchronous sampling

- One S/H amplifier per channel + one A/D converter per channel



Synchronous sampling

- One S/H amplifier per channel + one A/D converter



Sampling frequency

Sampling frequency is one of the factors affecting the accuracy of signal representation. The higher the sampling frequency, the better the signal representation (or better "horizontal" resolution).

Resolution of A/D conversion

The Analog to Digital Converter (ADC) converts an analog signal to its digital representation. The main characteristic of an ADC is its word length expressed in bits. It affects the ability of the ADC to represent the analog signal. In general, if the word length of the ADC is N bits, and the maximum input signal for the ADC is V, the quantization error q is given by

$$q = \frac{V}{2 \times 2^{N-1}} = 2^{-N} V$$

The larger the number of bits in a converter word, the smaller is the quantization error (or better "vertical" resolution).