Digital Signal Processing

Short notes for B.Tech ECE · EEE · GATE EC

Uncodemy Handwritten Notes · free for students

1. Introduction

- **DSP** = processing signals as numbers (samples) using a computer or a DSP chip.
- Chain: analog input → anti-aliasing filter → ADC → DSP → DAC → smoothing filter → analog output.

➤ DSP vs analog processing

Point	Digital	Analog
Accuracy	Set by word length, repeatable	Depends on component tolerance
Flexibility	Change the program	Change the hardware
Ageing / temperature drift	None	Yes
Very high frequencies	Limited by ADC speed	Works well
Linear phase filters	Easy (FIR)	Hard

➤ Discrete-time systems recap

- LTI system: y[n] = sum of h[k] x[n − k]. Stable if sum of |h[n]| is finite. Causal if h[n] = 0 for n < 0.
- **FIR** system: h[n] has finite length (no feedback). **IIR**: infinite length (uses feedback).
- Difference equation: y[n] = sum of bk x[n − k] − sum of ak y[n − k]. System function H(z) = B(z) / A(z).

2. DTFT

FORMULA X(e^jω) = sum over n of x[n] e^(−jωn). It is continuous in ω and periodic with period 2π.

- Exists if x[n] is absolutely summable. DTFT = Z-transform on the unit circle ($z = e^{j\omega}$).
- Frequency response $H(e^{j\omega})$: $\omega = 0$ is DC, $\omega = \pi$ is the highest digital frequency ($f_s/2$).
- Digital frequency $\omega = 2\pi f / f_s$ (radians per sample).

3. Discrete Fourier Transform (DFT)

FORMULA $X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}$, $W_N = e^{-j2\pi/N}$, $k = 0, 1, \ldots, N-1$.

FORMULA Inverse DFT: $x(n) = (1/N) \sum_{k=0}^{N-1} X(k) W_N^{-kn}$.

- DFT = N equally spaced samples of the DTFT over one period. Bin spacing = f_s / N (frequency resolution).
- Twiddle factor properties: $W_N^{k+N} = W_N^k$ (periodic), $W_N^{k+N/2} = -W_N^k$ (symmetric).

➤ Solved example (4-point DFT)

- $x(n) = \{1, 1, 0, 0\}$. Then $X(k) = 1 + W_4^k = 1 + e^{-j\pi k/2}$.
  - $X(0) = 1 + 1 = 2$
  - $X(1) = 1 + (-j) = 1 - j$
  - $X(2) = 1 + (-1) = 0$
  - $X(3) = 1 + (j) = 1 + j$
  - Check: $X(3)$ = conjugate of $X(1)$, as expected for a real $x(n)$.

➤ DFT properties

Property	Time	DFT
Linearity	$a\,x1 + b\,x2$	$a\,X1 + b\,X2$
Circular shift	$x((n-m))_N$	$W_N^{km}\,X(k)$
Circular convolution	$x1(n) \circledast_N x2(n)$	$X1(k)\,X2(k)$
Multiplication	$x1(n)\,x2(n)$	$(1/N)\,X1(k) \circledast_N X2(k)$
Real $x(n)$	$x(n)$ real	$X(N-k) = X^*(k)$
Parseval	$\sum \lvert x(n)\rvert^2$	$(1/N) \sum \lvert X(k)\rvert^2$

- Sum of all samples: X(0) = sum of x(n). And x(0) = (1/N) sum of X(k).

➤ Circular vs linear convolution

- Circular convolution of two N-point sequences gives N points (it wraps around).
- To get linear convolution of lengths L and M with the DFT, pad both to $N >= L + M - 1$.

➤ Solved example (circular convolution)

- x1 = {1, 2, 3, 4}, x2 = {1, 0, 0, 1}, N = 4.
  - x2 = δ(n) + δ(n − 3), so y(n) = x1(n) + x1((n − 3) mod 4) = x1(n) + x1((n + 1) mod 4)
  - y(0) = 1 + 2 = 3, y(1) = 2 + 3 = 5, y(2) = 3 + 4 = 7, y(3) = 4 + 1 = 5
  - y = {3, 5, 7, 5}

➤ Long sequences (block filtering)

- **Overlap-add**: split input into blocks, convolve each, add the overlapping tails.
- **Overlap-save**: overlap input blocks by M − 1 samples, keep only the valid outputs, discard the first M − 1.

4. Fast Fourier Transform (FFT)

- FFT = fast algorithm for the DFT. Radix-2 needs N = 2^m.
- **DIT** (decimation in time): split x(n) into even and odd samples. Input in bit-reversed order, output in normal order.
- **DIF** (decimation in frequency): split X(k) into even and odd. Input in normal order, output in bit-reversed order.
- Number of stages = log2 N, butterflies per stage = N / 2.

Method	Complex multiplications	Complex additions
Direct DFT	N^2	N (N − 1)
Radix-2 FFT	(N / 2) log2 N	N log2 N

➤ Solved example (FFT saving)

- N = 1024 (log2 N = 10).
  - Direct DFT: 1024^2 = 1,048,576 multiplications
  - FFT: (1024 / 2) x 10 = 5120 multiplications

- Speed gain ≈ 1,048,576 / 5120 ≈ 205 times

```
DIT butterfly (one multiply by W per butterfly):
  a ----------------+-----------> A = a + W b
                     \ /
                      X
                     / \
  b ---[ x W ]------+-----------> B = a - W b
```

EXAMPLE Bit reversal for N = 8: index 1 (001) → 4 (100), index 3 (011) → 6 (110). Input order: 0, 4, 2, 6, 1, 5, 3, 7.

5. IIR Filter Design

- Design an analog filter first (Butterworth, Chebyshev), then convert to digital.

➤ Analog prototypes

Filter	Passband	Stopband	Transition
Butterworth	Maximally flat (no ripple)	Monotonic	Wider
Chebyshev Type I	Equiripple	Monotonic	Sharper
Chebyshev Type II	Monotonic	Equiripple	Sharper
Elliptic	Equiripple	Equiripple	Sharpest for given order

FORMULA Butterworth: abs(H(jΩ))^2 = 1 / (1 + (Ω/Ωc)^(2N)). Gain at Ωc = −3 dB. Roll-off = 20N dB/decade.

FORMULA Butterworth order N >= log(sqrt((10^(0.1 As) − 1) / (10^(0.1 Ap) − 1))) / log(Ωs / Ωp), Ap and As in dB.

➤ Analog to digital conversion methods

Method	Mapping	Good	Bad
Impulse invariance	Pole s = pk → z = e^(pk T)	Keeps impulse response shape	Aliasing; not for high pass / band stop
Bilinear transform	s = (2/T)(1 − z^−1) / (1 + z^−1)	No aliasing, maps whole jΩ axis to unit circle	Frequency warping

FORMULA Bilinear warping: Ω = (2/T) tan(ω/2). Pre-warp the analog edge frequencies with this before design.

- Stable analog filter (LHP poles) → stable digital filter (poles inside the unit circle) in both methods.

6. FIR Filter Design

- FIR filters are always stable and can have exact **linear phase**.
- Linear phase condition: $h(n) = h(N - 1 - n)$ (symmetric) or $h(n) = -h(N - 1 - n)$ (antisymmetric). Delay = $(N - 1)/2$ samples.

Type	Symmetry	N	Cannot be used as
I	Symmetric	Odd	(Any filter possible)
II	Symmetric	Even	High pass (zero at $\omega = \pi$)
III	Antisymmetric	Odd	Low pass and high pass (zeros at 0 and π)
IV	Antisymmetric	Even	Low pass (zero at $\omega = 0$)

➤ Window method

- Take the ideal (infinite) $hd(n)$, shift by $(N - 1)/2$ and multiply by a window $w(n)$.
- Ideal low pass: $hd(n) = \sin(\omega c\ (n - \alpha)) / (\pi\ (n - \alpha))$, $\alpha = (N - 1)/2$.

Window	Peak side lobe	Main lobe width (approx)	Min stopband attenuation
Rectangular	−13 dB	$4\pi / N$	about 21 dB
Hanning	−31 dB	$8\pi / N$	about 44 dB
Hamming	−41 dB	$8\pi / N$	about 53 dB
Blackman	−57 dB	$12\pi / N$	about 74 dB

REMEMBER Better side lobe suppression always costs a wider main lobe (wider transition band). The Kaiser window lets you trade them with one parameter β.

- Other methods: **frequency sampling**, **optimal equiripple** (Parks–McClellan).

➤ FIR vs IIR

Point	FIR	IIR
Stability	Always stable	Can be unstable
Phase	Linear phase possible	Non-linear phase
Order for same spec	Higher	Lower
Feedback	No	Yes
Limit cycles	No	Possible

7. Filter Structures

- **IIR**: direct form I (separate delays for x and y), direct form II (canonic, minimum delays = max(M, N)), cascade (second-order sections), parallel (partial fractions), lattice.
- **FIR**: direct form (transversal / tapped delay line), cascade, linear-phase form (about half the multipliers), lattice.

```
Direct form II (second order):
x[n] -->(+)---------+------[b0]-->(+)--> y[n]
         ^          |               ^
         |       [z^-1]             |
         +--[-a1]---+------[b1]-----+
         ^          |               ^
         |       [z^-1]             |
         +--[-a2]---+------[b2]-----+
```

8. Finite Word Length Effects

- **Input quantisation** noise (ADC): variance $\Delta^2 / 12$.
- **Coefficient quantisation**: poles move; high-order direct form IIR filters are most sensitive (use cascade).
- **Product round-off** noise in multipliers.
- **Overflow**: prevented by scaling. **Limit cycles**: small oscillations in IIR filters even with zero input.
- Fixed point is cheaper; floating point gives more dynamic range.

9. Multirate DSP

- **Decimation by M**: low pass filter (cutoff π/M) then keep every Mth sample. New rate = fs / M.
- **Interpolation by L**: insert L − 1 zeros between samples, then low pass filter (cutoff π/L). New rate = L fs.
- Rational rate change L/M: interpolate first, then decimate (one combined filter with cutoff $\min(\pi/L, \pi/M)$).
- Uses: sample-rate converters, filter banks, audio (for example 48 kHz to 44.1 kHz).

FORMULA Down-sampler output $y[n] = x[Mn]$. Its spectrum: $Y(e^{j\omega}) = (1/M) \sum_{k=0}^{M-1} X(e^{j(\omega - 2\pi k)/M})$.

10. Most Asked Questions (practice)

1. Find the 4-point and 8-point DFT of a sequence.
2. State and prove the circular convolution property of the DFT.
3. Find linear convolution using circular convolution with zero padding.
4. Draw the 8-point radix-2 DIT FFT flow graph; count multiplications.
5. Compare impulse invariance and bilinear transformation. Explain warping.
6. Design a Butterworth low pass digital filter using the bilinear transform.
7. Design a linear phase FIR low pass filter using the Hamming window.
8. Compare FIR and IIR filters. Explain the four types of linear phase FIR.
9. Draw direct form I, direct form II, cascade and parallel structures.
10. Explain decimation and interpolation with spectra.

TIP GATE EC asks DFT values, circular convolution, FFT counts, bilinear mapping and linear phase FIR conditions. Always check symmetry and N first.