HOSPODOR

UNIVERSITY OF SANTA CLARA
Electrical Engineering and Computer Science

EECS 458
Sig. Proc. and Coding

Spring 1989

COURSE SYLLABUS

INSTRUCTOR

Dr. Hemant Thapar (408) 284-0308

Dr. Paul Siegel (408) 927-2044

COURSE GRADE

The course grade will be based on

Homework 20%; Mid-term Examination 30%; Final Examination 50%
SIGNAL PROCESSING COMPREHENSIVE

REFERENCES

C. D. Mee and E. D. Daniel, **Magnetic Recording Vol. II**, McGraw Hill, 1988.

J. C. Mallinson, **The Foundations of Magnetic Recording**, Academic Press, 1987.

J. Proakis, **Digital Communications**, McGraw Hill, 1983.

E. Lee and D. Messerschmitt, **Digital Communication**, Kluwer Academic, 1988.

Peterson and Weldon, **Error Correcting Codes**, 2-nd Ed., Prentice-Hall, 1984.

COURSE OUTLINE

Lecture	Content	Reference
1.	Introduction and Review of Signal Theory	
2.	Review of Signal Theory (contd.) Recording Channel Modelling	Notes
3.	Analog and Digital Detection Methods	Notes
4.	Partial Response Methods	Notes
5.	Partial Response Methods (contd.) Decision-Feedback Equalization	Notes
6.	MID-TERM EXAMINATION	
7.	Overview of Code Design Techniques	Notes
8.	Run-length Limited Codes	Notes
9.	Codes for Partial Response Channels	Notes
10.	Trellis Coding for Recording	Notes
11.	FINAL EXAMINATION	

HOSPODOR

RECORDING SYSTEM CLASSIFICATION

Analog Recording Systems – CONTINUOUS

- Information (message) signal to be recorded has infinite number of amplitude levels that change continuously with time.

- Typical Requirements: high signal-to-noise ratio (SNR), low distortion, and low cost.

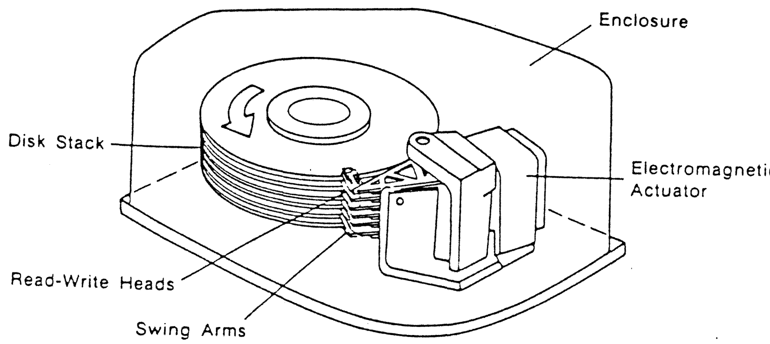
Digital Recording Systems – DISCRETE LEVELS

- Information (message) signal to be recorded has finite number (usually 2) of amplitude levels that change at discrete time points.

- Typical Requirements: <u>high reliability</u> (low probability of error), fast access to recorded information, and low cost ($/Mbyte).

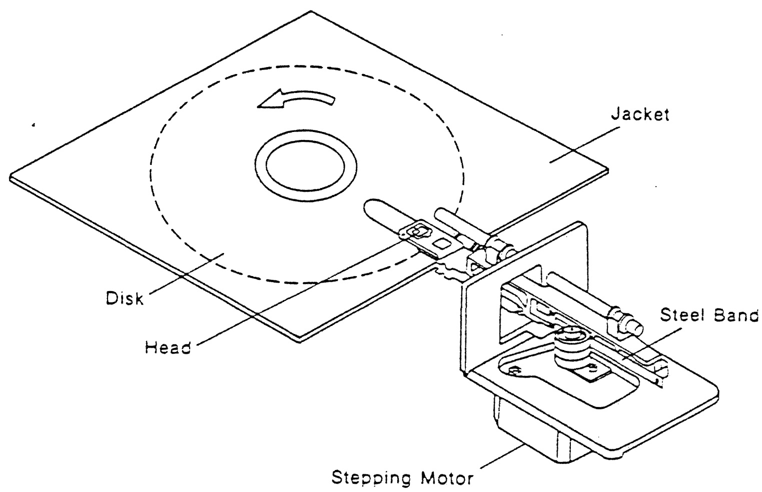
RECORDING SYSTEM APPLICATIONS

- Rigid Disk Drives



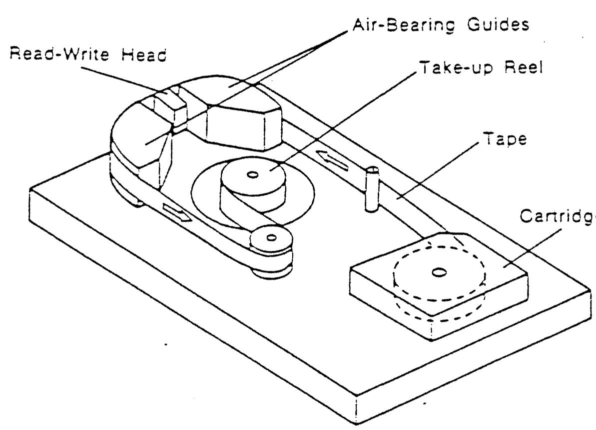
Rigid disk file components. (From Mee & Daniels)

- Flexible Disk Drives



Flexible diskette and head accessing system. (From Mee & Daniels)

- Tape Drives

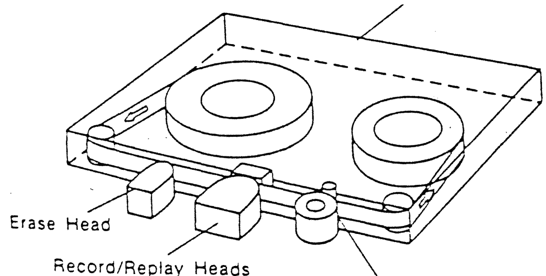


Schematic of cartridge drive for data recording. (From Mee & Daniels)

RECORDING SYSTEM APPLICATIONS (contd.)

Audio Recording Systems

- Analog Audio Recording



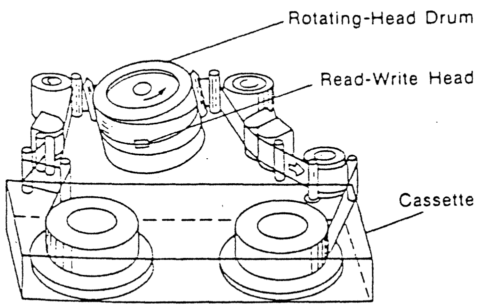
Audio cassette recorder. (From Mee & Daniel)

- Digital Audio Recording

RECORDING SYSTEM APPLICATIONS (contd.)

Image Recording Systems

- FM Video



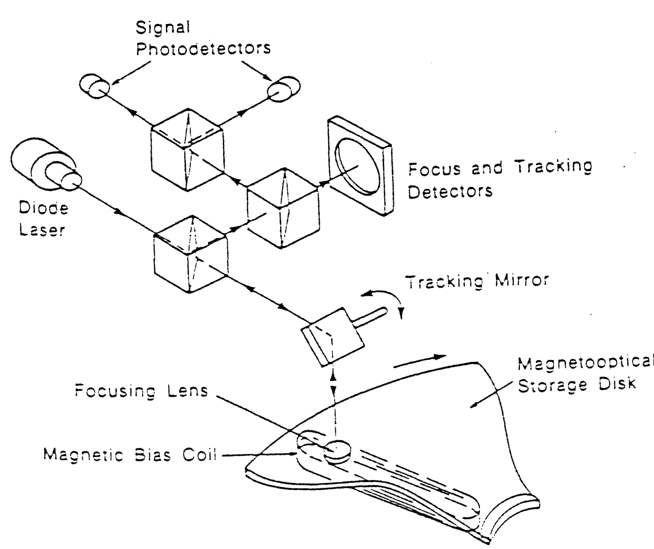
Helical-scan video recorder. (From Mee & Daniel)

- Digital Video

RECORDING SYSTEM APPLICATIONS (contd.)

Instrumentation Systems

Magneto-optical Recording Systems



Components for an optical read-write head for recording on magnetic media. (From Mee & Daniel)

DIGITAL MAGNETIC RECORDING SYSTEMS

General Requirements

- Low Probability of Error

- Fast Access to Recorded Data   REAL TIME SIGNAL PROCESSING

- Low Cost

Parameters of Interest

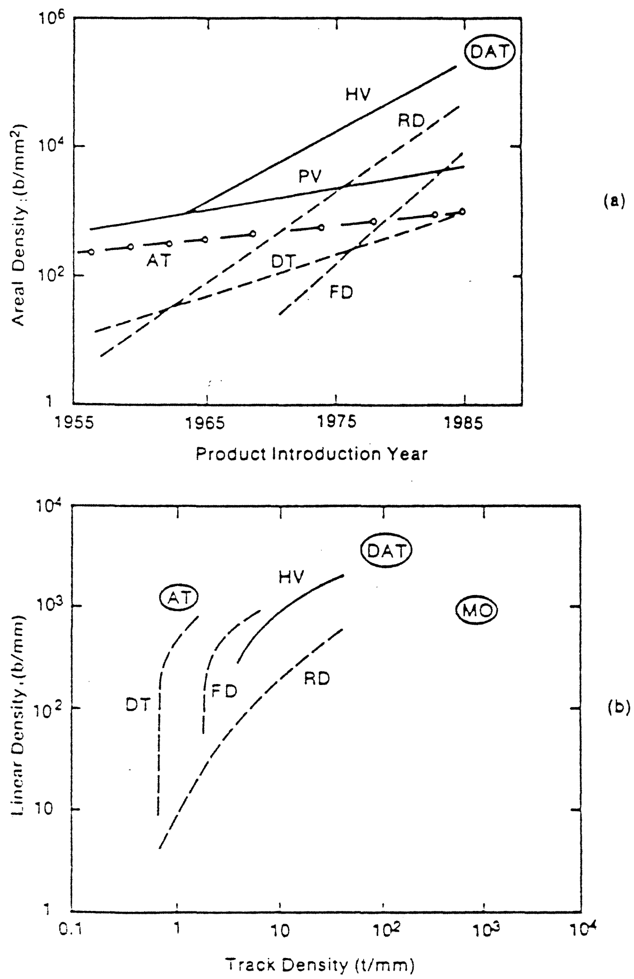
- Linear Density, number of bits per unit length along a track (bits/inch)

- Track Density, number of tracks per unit length (tracks/inch)

- Areal Density, number of bits per unit surface area (bits/inch^2); product of track and linear density.

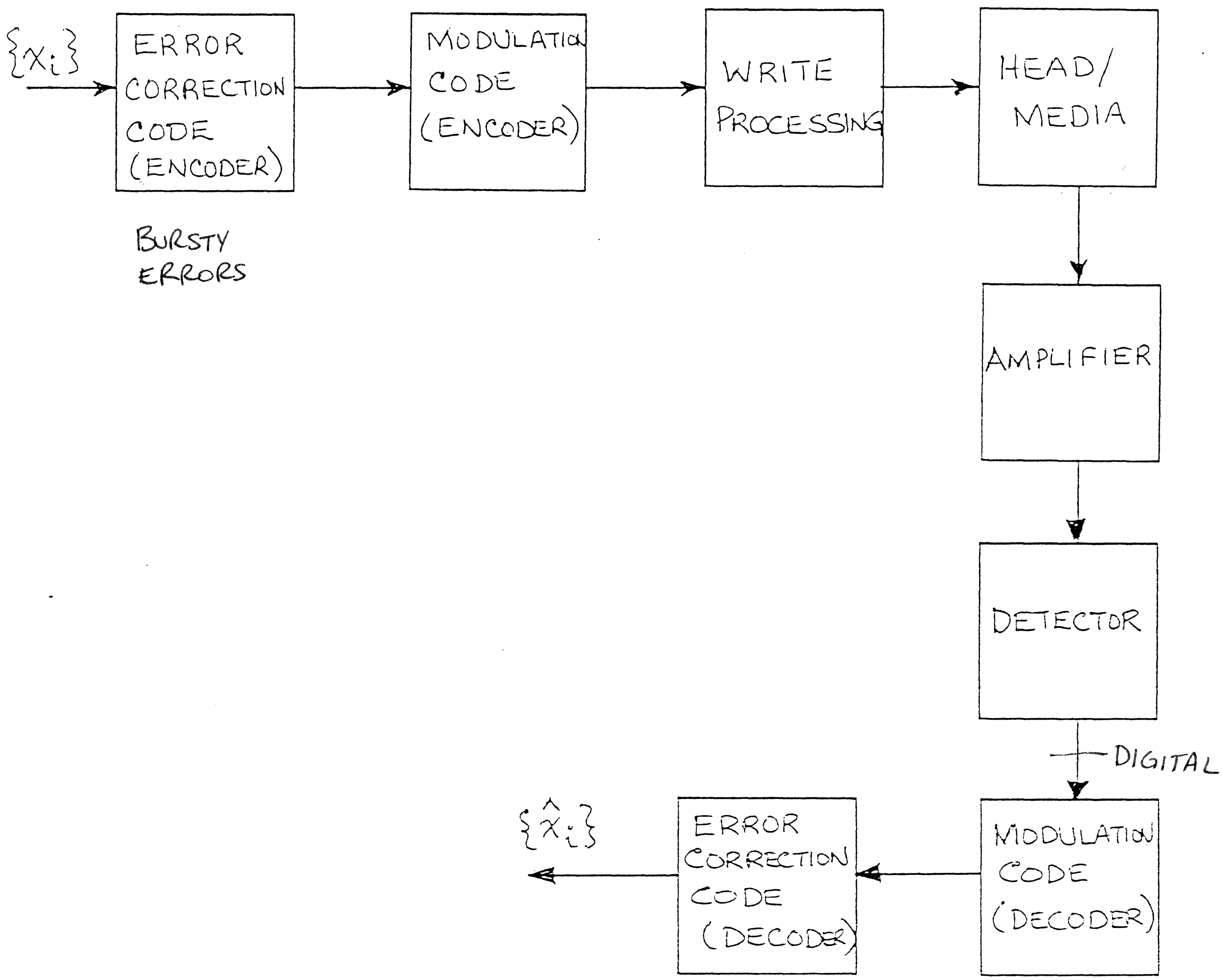
- Volumetric Density, number of bytes per unit volume (MBytes/ cu. ft.)

STATE-OF-THE-ART IN TPI AND BPI



(*a*) Areal density progress for magnetic recording systems; (*b*) relationships of linear and track densities. The density curves for audio and video recording are obtained from the approximation that the shortest recorded wavelength is equivalent to two recorded bits. HV, home video tape; PV, professional video tape; RD, rigid disk; DT, data tape; FD, flexible disk; AT, audio tape (DAT, digital audio tape); MO, magnetooptical storage projections. (From Mee & Daniels)

RECORDING CHANNEL



Gaussian distribution

If X represents the sum of a large number of independent random components, and if each component makes only a small contribution to the sum, then

$$p_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-(x-m)^2/2\sigma^2}$$

$$\bar{x} = m \qquad \sigma_x^2 = \sigma^2$$

(See Table T.6 for gaussian probabilities.)

Rayleigh distribution

If $R^2 = X^2 + Y^2$, where X and Y are independent gaussian r.v.'s with zero mean and variance σ^2, then

$$p_R(r) = \frac{r}{\sigma^2}\, e^{-r^2/2\sigma^2} \qquad r \geq 0$$

$$\bar{R} = \sqrt{\pi/2}\,\sigma \qquad \overline{R^2} = 2\sigma^2$$

TABLE T.6

GAUSSIAN PROBABILITIES

The probability that a gaussian random variable with mean m and variance σ^2 will have an observed value greater than $m + k\sigma$ is given by the function

$$Q(k) \triangleq \frac{1}{\sqrt{2\pi}} \int_k^{\infty} e^{-\lambda^2/2}\, d\lambda$$

called the area under the gaussian tail. Thus

$$P(X > m + k\sigma) = P(X \le m - k\sigma) = Q(k)$$

$$P(|X - m| > k\sigma) = 2Q(k)$$

$$P(m < X \le m + k\sigma) = P(m - k\sigma < X \le m) = \tfrac{1}{2} - Q(k)$$

$$P(|X - m| \le k\sigma) = 1 - 2Q(k)$$

$$P(m - k_1\sigma < X \le m + k_2\sigma) = 1 - Q(k_1) - Q(k_2)$$

Other functions related to $Q(k)$ are as follows:

$$\operatorname{erf} k \triangleq \frac{2}{\sqrt{\pi}} \int_0^{k} e^{-\lambda^2}\, d\lambda = 1 - 2Q(\sqrt{2}\,k)$$

$$\operatorname{erfc} k \triangleq \frac{2}{\sqrt{\pi}} \int_k^{\infty} e^{-\lambda^2}\, d\lambda = 1 - \operatorname{erf} k = 2Q(\sqrt{2}\,k)$$

$$\Phi(k) \triangleq \frac{1}{\sqrt{2\pi}} \int_0^{k} e^{-\lambda^2/2}\, d\lambda = \tfrac{1}{2} - Q(k)$$

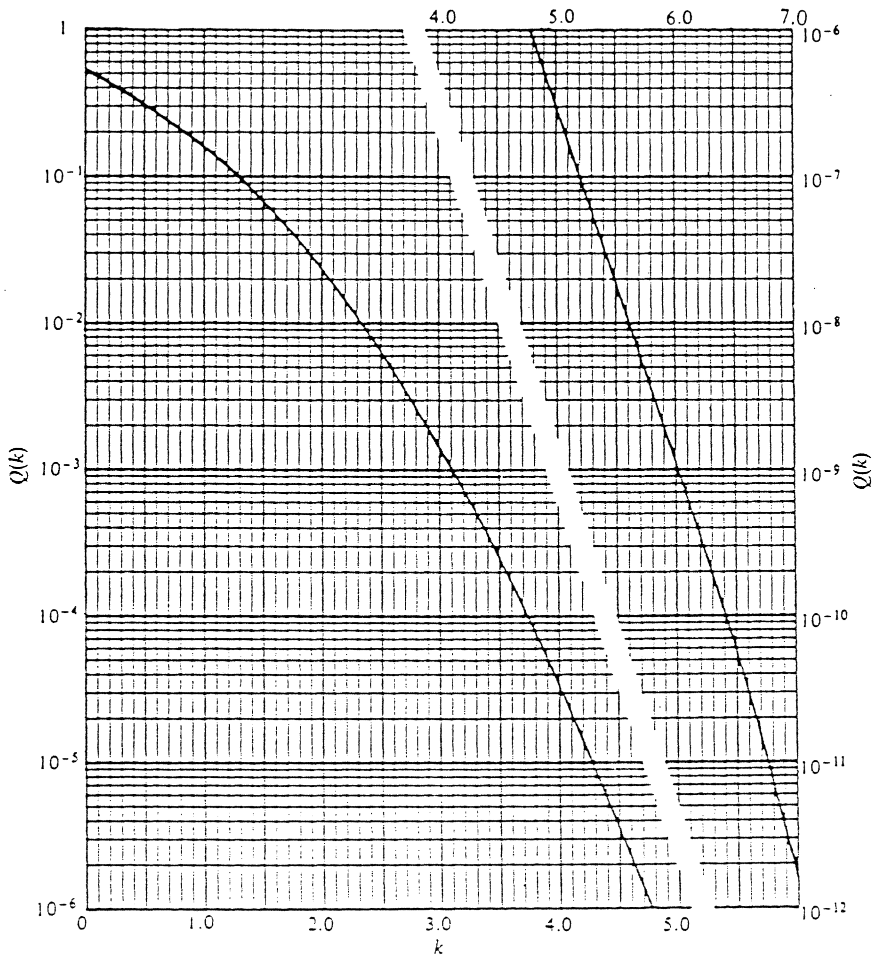
All of the foregoing relations are for $k \ge 0$. If $k < 0$, then

$$Q(-|k|) = 1 - Q(|k|)$$

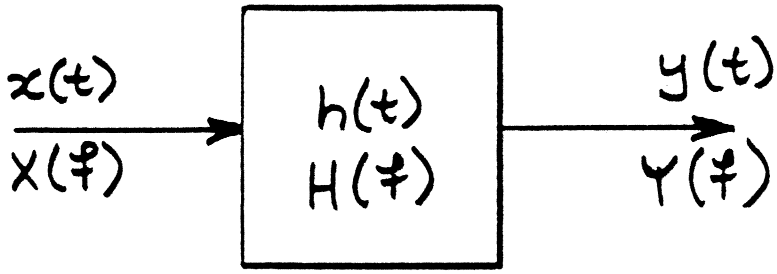
Numerical values of $Q(k)$ are plotted below for $0 \leq k \leq 7.0$. For larger values of k, $Q(k)$ may be approximated by

$$Q(k) \approx \frac{1}{\sqrt{2\pi}\,k}\, e^{-k^2/2}$$

which is quite accurate for $k > 3$.



RESPONSE OF LINEAR SYSTEM

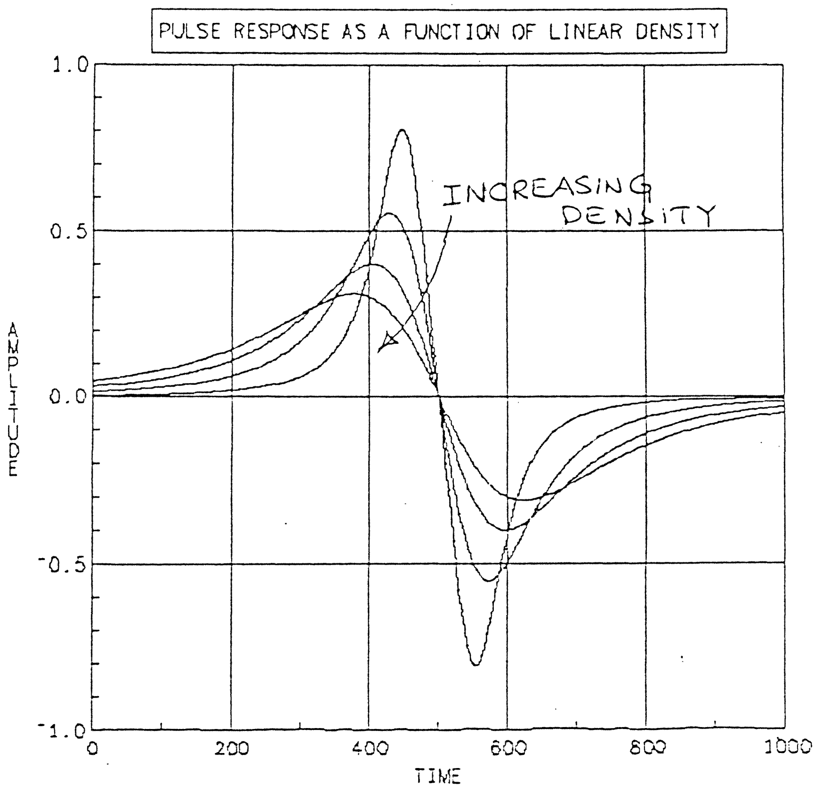


$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau$$

$$Y(f) = H(f)X(f)$$

$$S_y(f) = |H(f)|^2 S_x(f)$$

PROBLEM OF DETECTION IN MAGNETIC RECORDING



INTERSYMBOL INTERFERENCE (ISI) IS A KEY EFFECT OF HIGH-DENSITY RECORDING

increased isi $\Longrightarrow$ lowered signal-to-noise ratio (SNR)

lower SNR $\Longrightarrow$ higher error rate

SIGNAL PROCESSING AND CODING METHODS

- Peak Detection Method

- Partial Response Methods

- Equalization Methods

- Maximum-likelihood detection (Optimum)

- Codes for Peak Detection

- Codes for Partial Response Methods

- Signal space coding (trellis coding)

TABLE T.1
FOURIER TRANSFORMS

Definitions

Transform $$V(f) = \mathscr{F}[v(t)] = \int_{-\infty}^{\infty} v(t)e^{-j2\pi ft}\,dt$$

Inverse transform $$v(t) = \mathscr{F}^{-1}[V(f)] = \int_{-\infty}^{\infty} V(f)e^{j2\pi ft}\,df$$

Integral theorem

$$\int_{-\infty}^{\infty} v(t)w^*(t)\,dt = \int_{-\infty}^{\infty} V(f)W^*(f)\,df$$

Theorems

Operation	Function	Transform
Superposition	$a_1 v_1(t) + a_2 v_2(t)$	$a_1 V_1(f) + a_2 V_2(f)$
Time delay	$v(t - t_d)$	$V(f)e^{-j\omega t_d}$ PHASE SHIFT
Scale change	$v(\alpha t)$	$\dfrac{1}{\lvert\alpha\rvert} V\left(\dfrac{f}{\alpha}\right)$
Conjugation	$v^*(t)$	$V^*(-f)$
Duality	$V(t)$	$v(-f)$
Frequency translation	$v(t)e^{j\omega_c t}$	$V(f - f_c)$
Modulation	$v(t)\cos(\omega_c t + \phi)$	$\frac{1}{2}[V(f - f_c)e^{j\phi} + V(f + f_c)e^{-j\phi}]$
Differentiation	$\dfrac{d^n v(t)}{dt^n}$	$(j2\pi f)^n V(f)$
Integration	$\displaystyle\int_{-\infty}^{t} v(\lambda)\,d\lambda$	$\dfrac{1}{j2\pi f} V(f) + \frac{1}{2}V(0)\,\delta(f)$
Convolution	$v * w(t)$	$V(f)W(f)$
Multiplication	$v(t)w(t)$	$V * W(f)$
Multiplication by t^n	$t^n v(t)$	$(-j2\pi)^{-n} \dfrac{d^n V(f)}{df^n}$

Transforms

Function	$v(t)$	$V(f)$
Rectangular	$\Pi\left(\frac{t}{\tau}\right)$	$\tau \operatorname{sinc} f\tau$
Triangular	$\Lambda\left(\frac{t}{\tau}\right)$	$\tau \operatorname{sinc}^2 f\tau$
Gaussian	$e^{-\pi(bt)^2}$	$(1/b)e^{-\pi(f/b)^2}$
Causal exponential	$e^{-bt}u(t)$	$\frac{1}{b+j2\pi f}$
Symmetric exponential	$e^{-b\|t\|}$	$\frac{2b}{b^2+(2\pi f)^2}$
Sinc	$\operatorname{sinc} 2Wt$	$\frac{1}{2W}\Pi\left(\frac{f}{2W}\right)$
Sinc squared	$\operatorname{sinc}^2 2Wt$	$\frac{1}{2W}\Lambda\left(\frac{f}{2W}\right)$
Constant	1	$\delta(f)$
Phasor	$e^{j(\omega_c t+\phi)}$	$e^{j\phi}\,\delta(f-f_c)$
Sinusoid	$\cos(\omega_c t+\phi)$	$\frac{1}{2}[e^{j\phi}\,\delta(f-f_c)+e^{-j\phi}\,\delta(f+f_c)]$
Impulse	$\delta(t-t_d)$	$e^{-j\omega t_d}$
Sampling	$\sum_{k=-\infty}^{\infty}\delta(t-kT_s)$	$f_s\sum_{n=-\infty}^{\infty}\delta(f-nf_s)$
Signum	$\operatorname{sgn} t$	$1/j\pi f$
Step	$u(t)$	$\frac{1}{j2\pi f}+\frac{1}{2}\delta(f)$

for x from 0

$\mathrm{sinc}^2\, x$
0.000
0.002
0.007
0.013
0.016
0.016
0.014
0.009
0.004
0.001
0.000
0.001
0.003
0.006
008
08
0.007
0.005
0.002
0.001

TABLE T.5
PROBABILITY FUNCTIONS

Binomial distribution

Let the discrete r.v. I be the number of times an event A occurs in n independent trials. If $P(A) = \alpha$, then

$$P_I(i) = \binom{n}{i}\alpha^i(1-\alpha)^{n-i} \qquad i = 0, 1, \ldots, n$$

$$\bar{i} = n\alpha \qquad \sigma_i^2 = n\alpha(1-\alpha)$$

If $n \gg 1$, $\alpha \ll 1$, and $m = n\alpha$ remains finite, then

$$P_I(i) \approx e^{-m}m^i/i!$$

Poisson distribution

Let the discrete r.v. I be the number of times an event A occurs in time T. If $P(A) = \mu\,\Delta T \ll 1$ in a small interval ΔT, and if multiple occurrences are statistically independent, then

$$P_I(i) = e^{-\mu T}(\mu T)^i/i! \qquad \bar{i} = \mu T$$

Uniform distribution

If the continuous r.v. X is equally likely to be observed anywhere in a finite range, and nowhere else, then

$$p_X(x) = \frac{1}{b-a} \qquad a \leq x \leq b$$

$$\bar{x} = \tfrac{1}{2}(a+b) \qquad \sigma_x^2 = \tfrac{1}{12}(b-a)^2$$

Sinusoidal distribution

If X has a uniform distribution with $b - a = 2\pi$ and $Z = A\cos(X + \theta)$, where A and θ are constants, then

$$p_Z(z) = \frac{1}{\pi\sqrt{A^2 - z^2}} \qquad |z| \leq A$$

$$\bar{z} = 0 \qquad \sigma_z^2 = \tfrac{1}{2}A^2$$

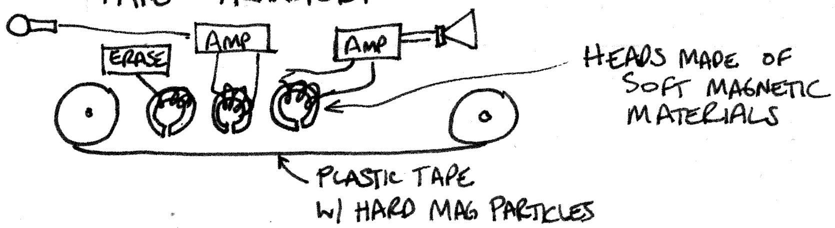
EECS 458 3/29/89

25
15
9
49

MAGNETISM

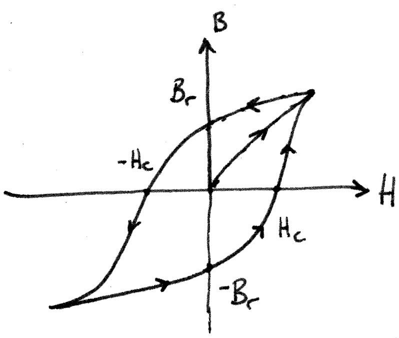
ELECTROMAGNETIC INDUCTION

TAPE TRANSPORT



ERASE ~~B~~WRITE READ
HEADS

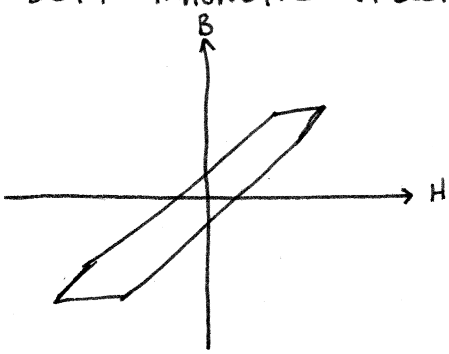
HARD MAGNETIC (HIGH COERCIVITY REQUIRED TO CHANGE STATE)



B_r = REMNANT MAGNETIZATION

H_c = COERCIVITY

SOFT MAGNETIC (RECORDING HEAD, NO REMNANT MAGNETIZATION



RECORDING SYSTEM

HEAD
MEDIUM
TRANSPORT MECHANISM

SIGNAL PROCESSING - DIGITAL ONLY

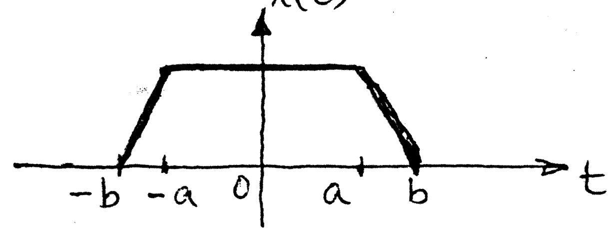
RECORDING CHANNEL

SUPER POSITIONING & BIT SHIFTING
INTERSYMBOL INTERFERENCE

EECS 458

Homework

1. Determine the Fourier transform of the trapezoidal pulse given below. (Hint: Use the differentiation theorem).



2. Given $x(t) = \text{sinc}\left(\frac{t}{T}\right)$, obtain and sketch the amplitude and the phase ~~sp~~ spectrum of $y(t)$, where

    a) $y(t) = x(t) + x(t-T)$

    b) $y(t) = x(t) - x(t-T)$

    c) $y(t) = x(t) - x(t-2T)$

    Assume $|f| \leq 1/2T$. NO INTERFERENCE

3. Find the response of a system with impulse response $h(t) = 10\exp(-10t)\,u(t)$ to the input $x(t) = \Pi\left(\frac{t}{2}\right)$. [$u(t) \triangleq$ unit step function).

4. Compare $Q(d/2\sigma)$ for $d = 2$ and $\sigma = 0.5$ and $\sigma = 0.25$.

$Q\left(\frac{d}{2\sigma}\right) = Q(2)$

$Q(4)$

$\frac{1}{\sqrt{2\pi}\,K} e^{-K^2/2}$

I. FOURIER TRANSFORM AND SIGNAL SPECTRA

Defn.: The Fourier transform of a signal $x(t)$ is

$$X(f) = \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt$$

FREQUENCY DOMAIN

$X(f)$ is in general complex, i.e.,

$$X(f) = \underset{\text{REAL}}{A(f)} + j \underset{\text{IMAGINARY}}{B(f)} \qquad \text{QUADRATURE FORM}$$

$$= |X(f)| e^{j\theta_x(f)} \qquad \text{POLAR FORM}$$

where $|X(f)|$ = Amplitude spectrum of $x(t)$

$$= \sqrt{A^2(f) + B^2(f)}$$

FREQUENCY CONTENT OF SIGNAL

and $\theta_x(f)$ = Phase of $x(t)$ PHASE SPECTRUM

$$= \tan^{-1}\left[\frac{B(f)}{A(f)}\right]$$

HOW FREQUENCIES ARE DISTRIBUTED IN TIME RELATIVE TO EACH OTHER

Defn.: The inverse Fourier transform of $X(f)$ is

$$x(t) = \mathcal{F}^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$$

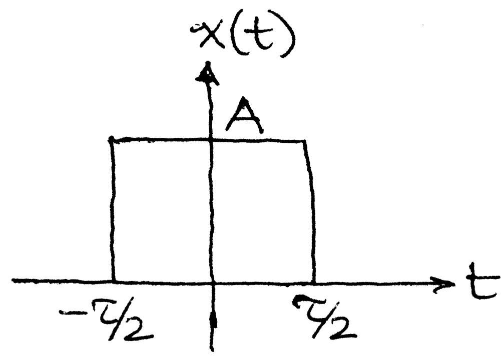
- <u>Sufficient</u> (Dirichlet) conditions for the existence of Fourier transform:

  1) $x(t)$ has a finite number of maxima, minima, and number of discontinuities over a finite time duration
     RELATIVELY SMOOTH

  2) $x(t)$ is absolutely integrable, i.e.,

$$\int_{-\infty}^{\infty} |x(t)|\, dt < \infty$$

- <u>EXAMPLE</u>:



$$x(t) = \text{rect}\left(\frac{t}{\tau}\right)$$

$$= \Pi\left(\frac{t}{\tau}\right)$$

$$X(f) = \int_{-\infty}^{\infty} \text{rect}\left(\frac{t}{\tau}\right) e^{-j2\pi f t}\, dt = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t}\, dt$$

$$= \int_{-\tau/2}^{\tau/2} A e^{-j2\pi f t}\, dt = -\frac{A}{j\pi 2f} e^{-j2\pi f t}\Big|_{-\tau/2}^{\tau/2} = \frac{-A}{j2\pi f}\left(e^{-\frac{j2\pi f\tau}{2}} - e^{j2}\right.$$

$$= \frac{+A}{\pi f} \cdot \sin \pi f \tau$$

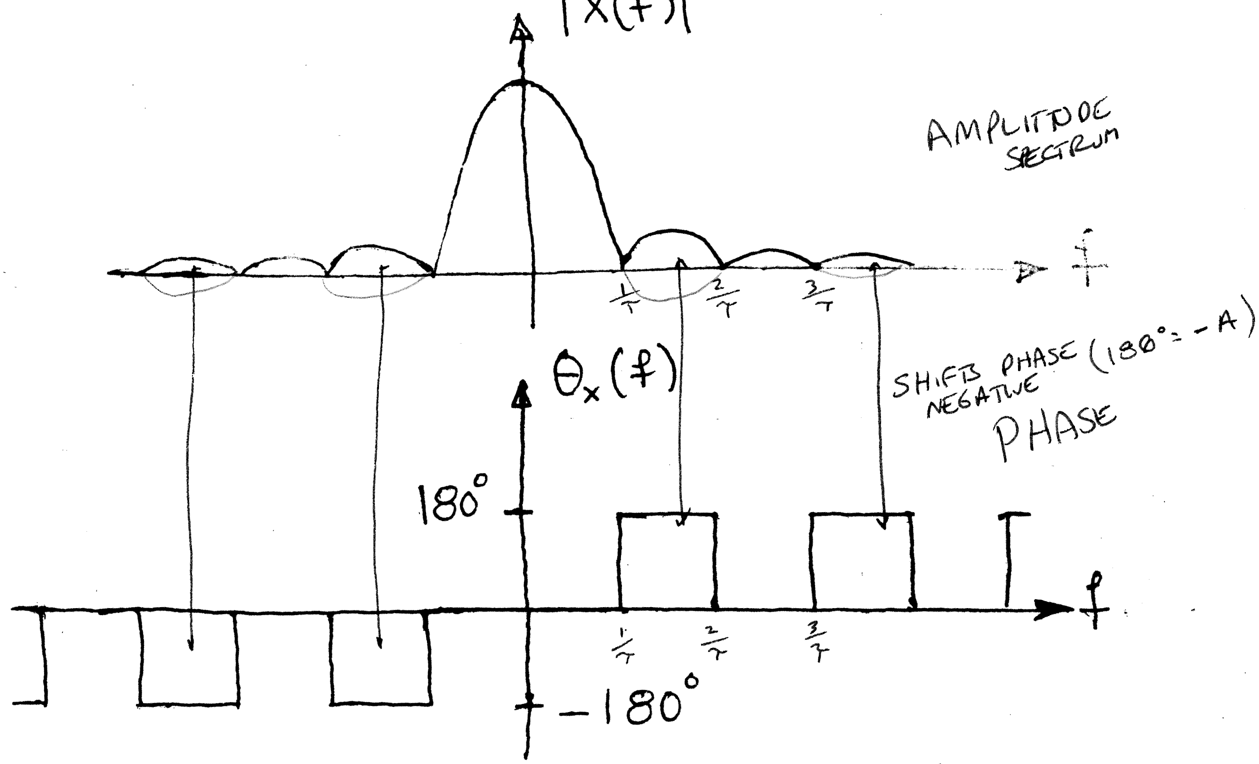
$$= A\tau\, \text{sinc}\, f\tau$$

$$= +A\tau \frac{\sin \pi f \tau}{\pi f \tau}$$

$$= A\tau\, \text{sinc}\, (f\tau)$$

where $\text{sinc}\, u \triangleq \dfrac{\sin \pi u}{\pi u}$

$$\boxed{\Pi\left(\frac{t}{\tau}\right) \Longleftrightarrow A\tau\, \text{sinc}\, f\tau}$$



II. FOURIER TRANSFORM THEOREMS

a) Spectral symmetry of real signals: If $x(t)$ is real, then

$$X(-f) = X^*(f)$$

$$\Rightarrow \quad |X(-f)| = |X(f)|$$

$$\theta_x(-f) = -\theta_x(f)$$

b) Parseval's relationship

$$\underset{\text{ENERGY}}{E} \triangleq \int_{-\infty}^{\infty} |x(t)|^2 \, dt \Leftrightarrow \int_{-\infty}^{\infty} |X(f)|^2 \, df$$

c) Linearity

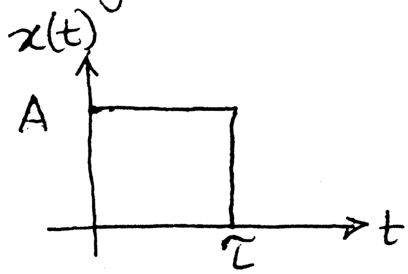
$$\mathcal{F}[a x(t) + b y(t)] \overset{=}{\Leftrightarrow} a X(f) + b Y(f)$$

d) Time delay: If $x(t) \Leftrightarrow X(f)$

$$x(t-t_0) \Leftrightarrow X(f)\, e^{-j2\pi f t_0}$$

SHIFT RIGHT
$(t+t_0)$
SHIFT LEFT

e.g.



$$x(t) = A\, \text{rect}\left(\frac{t-t_0}{\tau}\right) \qquad t_0 = \frac{\tau}{2}$$

$$\Leftrightarrow (A\tau\, \text{sinc} f\tau)\, e^{-j2\pi\tau}$$

e) Scale Change: If $x(t) \Leftrightarrow X(f)$

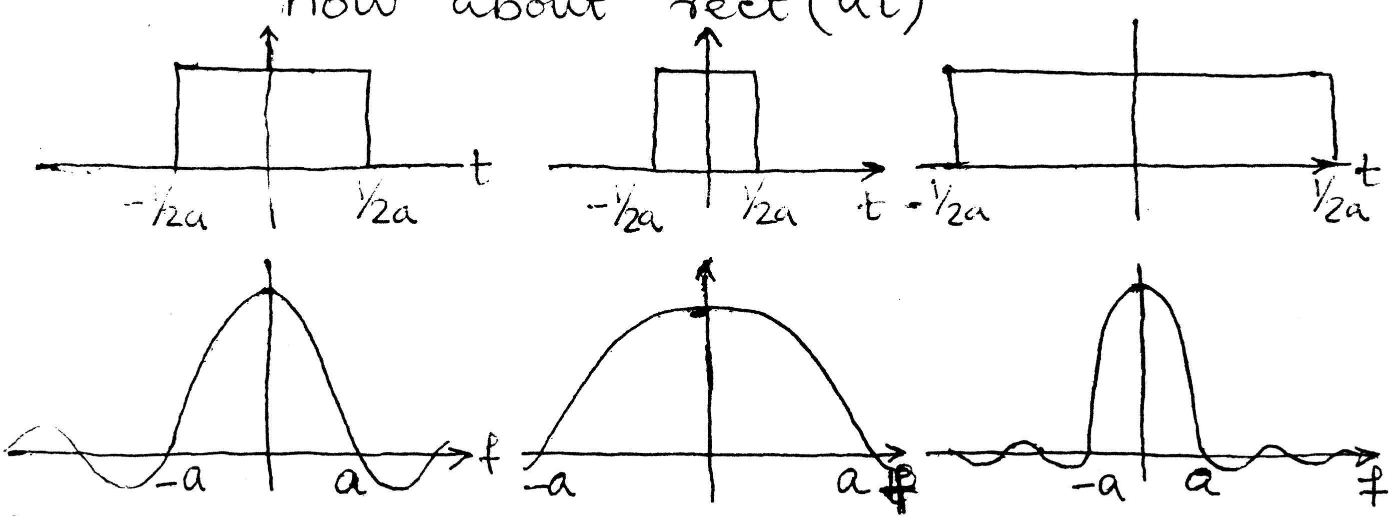
$$x(at) \Leftrightarrow \frac{1}{|a|} X\left(\frac{f}{a}\right)$$

NARROW IN TIME
⇒ MORE HIGH FREQUENCIES

Example: $x(t) = \text{rect}(t)$

$\Leftrightarrow X(f) = \text{sinc} f$ (since $\tau = 1$)

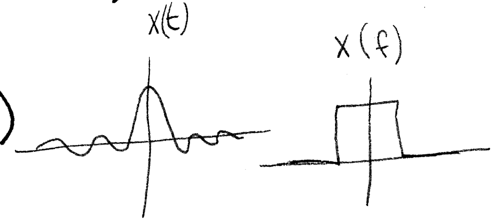
how about $\text{rect}(at)$



NARROW PULSE HAS
MANY HIGH FREQUENCIES

f) Duality: If $x(t) \Leftrightarrow X(f)$

then $X(t) \Leftrightarrow x(-f)$



g) Differentiation & Integration:

If $x(t) \Leftrightarrow X(f)$

then $\dfrac{dx(t)}{dt} \Leftrightarrow j2\pi f\, X(f)$ DERIVITIVE

$\dfrac{d^n x(t)}{dt^n} \Leftrightarrow (j2\pi f)^n X(f)$

if $x(t)$ is d.c free, then

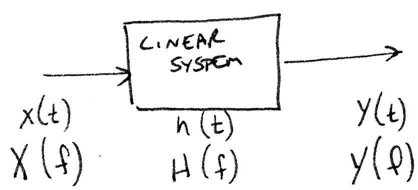
$$\int_{-\infty}^{t} x(\tau)\, d\tau \Leftrightarrow \frac{X(f)}{j2\pi f}$$ INTEGRAL

h) Multiplication and Convolution: If

$x(t) \Leftrightarrow X(f)$ and $y(t) \Leftrightarrow Y(f)$

then $$\int_{-\infty}^{\infty} x(t-\tau)\, y(\tau) \Leftrightarrow X(f)\, Y(f)$$

$(\triangleq x(t) * y(t))$



$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau$

$Y(f) = H(f)\cdot X(f)$

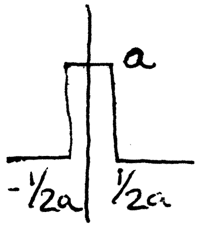
$x(t)y(t) \Leftrightarrow \int_{-\infty}^{\infty} X(f-\nu)\, Y(\nu)$

BY DUALITY

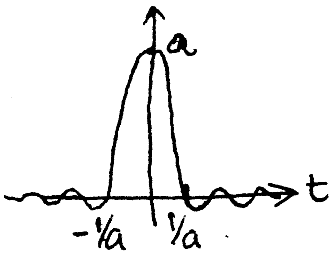
III. DIRAC DELTA FUNCTION & SAMPLING

Defn.: delta function (generalized function)

$$\int_{-\infty}^{\infty} \delta(t)\, dt = 1$$



$$\delta(t) = \lim_{a\to\infty} a \,\mathrm{rect}(at)$$



$$\delta(t) = \lim_{a\to\infty} a \,\mathrm{sinc}\, at$$

shape is not miportant !!

- Important features of $\delta(t)$:
  - unit area
  - rapid decrease to zero for $t \neq 0$ (NARROW)

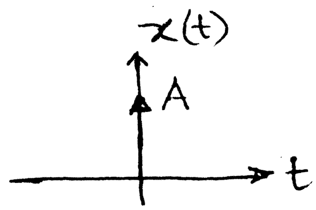
- Properties of the Dirac delta function:

a) Convolution: $\int_{-\infty}^{\infty} f(\tau)\,\delta(t-\tau)\,d\tau = f(t)$

b) $\delta(-t) = \delta(t)$

c) $\mathcal{F}[\delta(t)] = 1$

d) $x(t) = A\,\delta(t)$

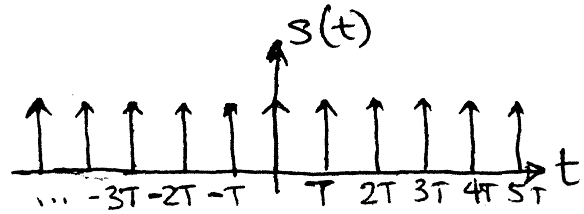


REPRESENTATION OF WEIGHTED IMPULSE (AREA = A)

e) sifting property: $\int_{-\infty}^{\infty} f(t)\,\delta(t-t_0)\,dt = f(t_0)$

- <u>SAMPLING</u>

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT)$$



PERIODIC IN t

$$x(t)\cdot s(t) = \left[x(t)\right]\left[\sum_{n=-\infty}^{\infty} \delta(t-nT)\right]$$

$$= \sum_{n=-\infty}^{\infty} x(nT)\,\delta(t-nT)$$

(a sequence of numbers)

if $s(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT)$ (a periodic signal)

$$\Leftrightarrow \quad S(f) = \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta\left(f-\frac{k}{T}\right)$$

FOURIER TRANSFORM OF IMPULSE TRAIN IS ANOTHER IMPULSE TRAIN

$1/T$ = sampling rate ; T = sampling interval

$f_s \triangleq 1/T$ = sampling frequency.

IV. SIMILARITY BETWEEN SIGNALS: DISTANCE AND CORRELATION

DISTANCE

defn.: $$d^2[f(t), g(t)] = \int_{-\infty}^{\infty} [f(t)-g(t)]^2 dt$$

$$= \int_{-\infty}^{\infty} f^2(t)\,dt + \int_{-\infty}^{\infty} g^2(t)\,dt - 2\int f(t)g(t)\,dt$$

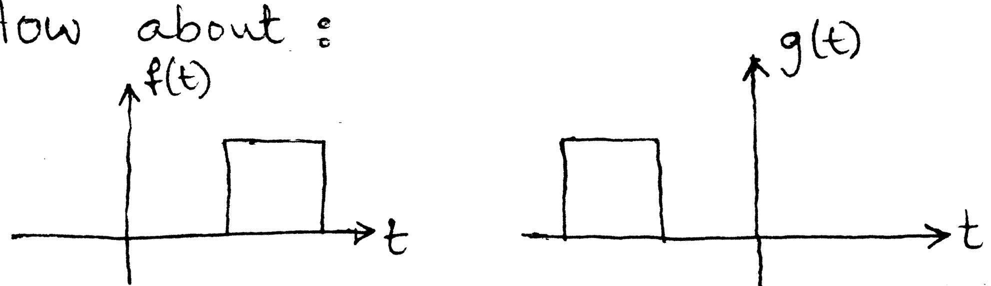
OVERLAPPING TERM

$d(F,G) \geq 0$ (equality iff $F=G$)

$d(X,Y) \leq d(Z,X) + d(Z,Y)$ TRIANGULAR INEQUALITY

$\underline{f} = [f_1\ f_2\ f_3 \dots f_N]$ $\quad \underline{g} = [g_1\ g_2 \dots g_N]$ $\quad d^2(\underline{f},\underline{g}) = \sum_{i=1}^{N} (f_i - g_i)^2$

How about:



MAXIMUM (NON OVERLAPPING)

$$d[f(t), g(t)] = \cancel{0} \; \int_{-\infty}^{\infty} f^2(t)\,dt + \int_{-\infty}^{\infty} g^2(t)\,dt \; ?$$

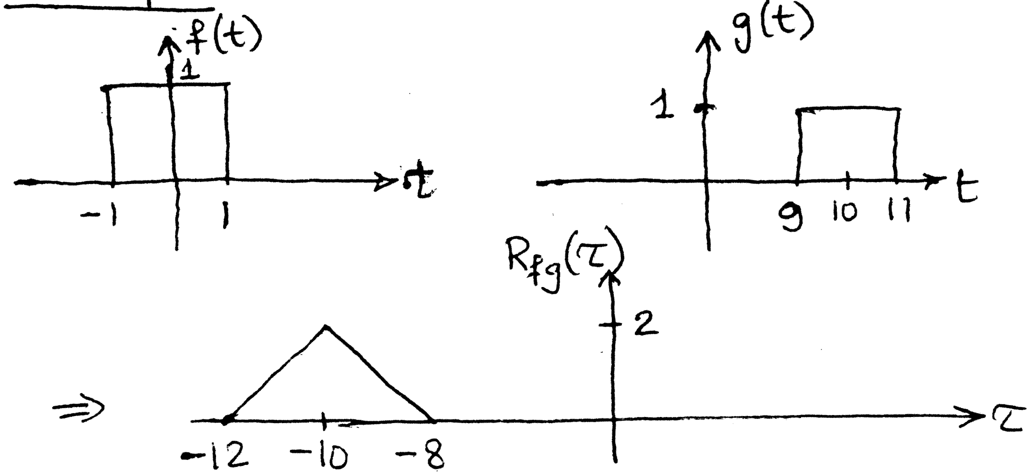
distance function does not account for shifts!!

defn.: Time cross-correlation function

$$R_{fg}(\tau) \triangleq \int_{-\infty}^{\infty} f(t)\, g(t-\tau)\, dt$$
$$= \int_{-\infty}^{\infty} f(t+\tau)\, g(t)\, dt$$

if $R_{fg}(\tau) = 0 \Rightarrow f$ and g are uncorrelated.

<u>Example:</u>



("most similar" for a shift of -10 units)

<u>defn.</u>: Time autocorrelation function

$$R_{xx}(\tau) \triangleq \int_{-\infty}^{\infty} x(t)\, x(t-\tau)\, dt$$

CORRELATION OF SIGNAL WITH ITSELF

Properties of $R_{xx}(\tau)$:

1) $R_{xx}(0) = \int_{-\infty}^{\infty} x(t)\, x(t)\, dt$ = Energy or Power of

2) $R_{xx}(-\tau) = R_{xx}(\tau)$ $\quad F[R_{xx}(\tau)] \Leftrightarrow G(f)$ ENERGY/POWER SPECTRAL DENSITY

3) $R_{xx}(\tau) \leq R_{xx}(0) \quad \forall \tau$

- $\mathcal{F}[R_{xx}(\tau)] \Leftrightarrow S_x(f)$ Energy or Power spectral density of $x(t)$

V. PROBABILITY & RANDOM VARIABLES

defn. : Probability :

a) Frequency of occurance

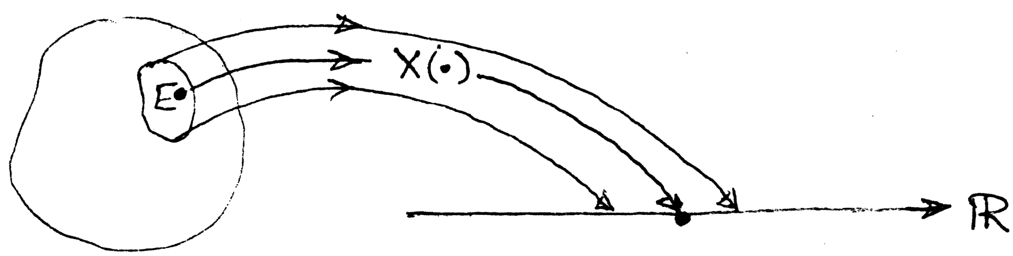
$$P(E) = \lim_{n\to\infty} \frac{n_E}{n}$$

b) Favorable outcome

$$P(E) = \frac{n(E)}{n}$$ NUMBER OF WAY IN WHICH E CAN OCCUR / NUMBER OF POSSIBLE OUTCOMES

– Axiomatic Approach

defn. : Random variable is a mapping of the outcome onto the real line.



- Probability distribution function, $F(x)$:

$$\text{Prob}(X \le x_0) = F(x_0)$$

a) $F(\infty) = 1$, $F(-\infty) = 0$

b) $x_1 \le x_2 \Rightarrow F(x_1) \le F(x_2)$ nondecreasing function of x.

c) $F(x)$ is right-continuous,

$$F(x) = \lim_{\epsilon \to 0} F(x+\epsilon) , \quad \epsilon > 0$$

- Probability density function: (pdf)

$$f(x) = \frac{dF(x)}{dx}$$

a) $\int_{-\infty}^{\infty} f(x)\,dx = 1$

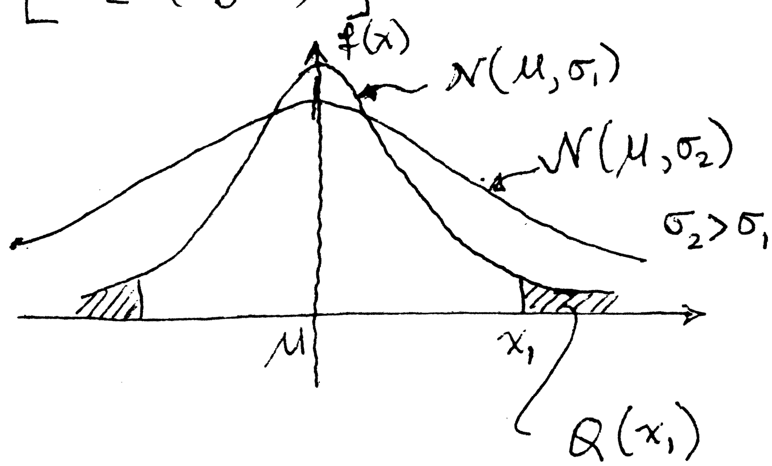
b) $F(x) = \int_{-\infty}^{x} f(\sigma)\,d\sigma = P(X \leq x)$

c) $F(x_2) - F(x_1) = \int_{x_1}^{x_2} f(\sigma)\,d\sigma = P(x_1 \leq X \leq x_2)$

<u>EXAMPLE</u> : Gaussian pdf

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

$\mathcal{N}(\mu, \sigma)$



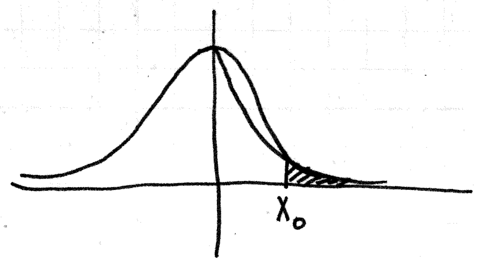
- WHITE GAUSSIAN NOISE

GAUSSIAN $\Rightarrow$ distribution of noise amplitude

WHITE $\Rightarrow$ correlation between amplitudes

$$R(\tau) = \delta(\tau)$$

GAUSSIAND DENSITY FUNCTION



$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(X-\mu)^2}{2\sigma^2}\right)}$$

μ IS EXPECTED (MEAN / AVG) VALUE OF X

σ IS STANDARD DEVIATION (DISPERSION)

σ^2 IS VARIANCE OR POWER (OF NOISE)

KNOWN UNKNOWN RANDOM VARIABLE

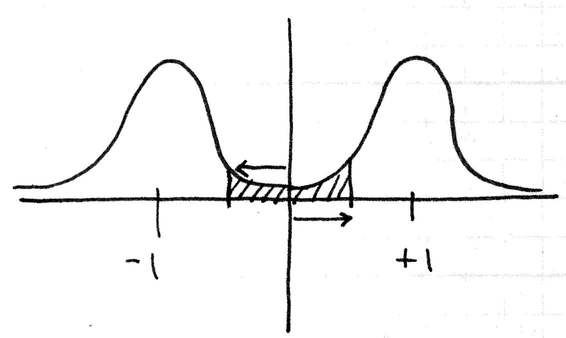
$r = s + n$

$E[n] = \emptyset$ EXPECTED VALUE OF NOISE (IS CONSTANT)

$E[r] = E[s] + E[n]$

$= s$

$\sigma_r^2 = \sigma_n^2$



ERRORS WHERE $-1 \rightarrow +1$
OR $+1 \rightarrow -1$

$P(X > X_0) \triangleq Q(X_0)$

$P(X < X_0) \triangleq 1 - Q(X_0)$

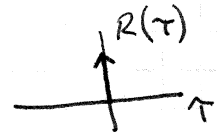
NORMALIZED $X_n = \frac{X - \mu}{\sigma}$ $\mu = 10$, $\sigma = 3$ $P(X > 2\emptyset) = Q\left(\frac{2\emptyset - 1\emptyset}{3}\right) = Q(3.33)$

WHITE GAUSSIAN NOISE

GAUSSIAN → DISTRIBUTION

WHITE → CORRELATION OF NOISE AMPLITUDES IN TIME

AUTOCORRELATION FUNCTION $= \delta(\tau)$



Least-squares storage-channel identification

by J. M. Cioffi

Pulse (dibit) and step (transition) responses for magnetic-storage channels are important for detection-circuitry design and for comparison of various media, heads, and other channel components. This paper presents a least-squares procedure that can be used to identify the dibit and transition responses from measurements of the read-head response to any known data sequence written on the medium. The method yields significantly higher-quality estimates for the dibit and step shapes than does determining these same characteristics by measuring the average response to isolated transition or by performing a Discrete Fourier Transform (DFT) on the response to a pseudorandom data pattern. The new method can be implemented off line but also can be made sufficiently efficient to be implemented with a microprocessor for use in self-optimizing (adaptive) channel detection circuitry.

1. Introduction

Storage-channel identification is the measurement and/or computation of the characteristics of the read-back channel in a data storage device, such as a magnetic disk, magnetic tape, or optical disk. The identified characteristics are most often the channel's response to a step input (the "transition" response) or to a pulse (the "dibit" response). These characteristics are important for many purposes, such as the design of the detection circuitry (especially for equalizers and for maximum-likelihood detectors), for determining the maximum data density of the device, and for comparing various media, heads, and other channel components.

This paper presents a least-squares procedure for identification of the linear time-invariant filter that most closely approximates the desired step or pulse responses. The storage device is excited with a known data sequence, and, later, the read-head response to the known sequence is measured (or digitized) at regular intervals. The resulting measurements are then processed via the least-squares procedure to determine the step and/or pulse responses.

The resultant estimates of these responses are of significantly higher resolution (higher quality) than those produced by previous procedures, such as measuring the average response to isolated transitions (or isolated dibits) or computing the Discrete Fourier Transform (DFT) of the response to some known (usually pseudorandom) data pattern. Furthermore, the new method, although based on a linear model of the channel as presented here, can indicate the average accuracy of the linear model over any data pattern, thus indicating the presence of potential nonlinearities in the responses, unlike the aforementioned methods. The degree of agreement between the linear model and measurements can be useful in determining the data rates at which various data detection methods do and do not apply.

Section 2 defines in more detail the quantities used in channel identification and the least-squares procedure, and it compares the quality of estimates of the new and previous procedures. Section 3 studies some details of the solution and displays the results of the new procedure for several measurements taken from actual storage devices, including magnetic disks with thin-film heads, tape systems with magnetoresistive heads, and optical disks. Section 4 is a brief conclusion. Appendix A extends the channel identification procedure to apply at any digitizer sampling rate (an integer ratio of the sampling to data rates is assumed in the main body of the paper). Appendix B discusses streamlining of the least-squares procedure for possible use with adaptive detection methods, while Appendix C discusses the detection of nonlinearities.

2. Storage-channel identification methods

This section mathematically defines and analyzes the quantities and procedures used in storage-channel identification. **Figures 1(a)** and **1(b)** summarize the definitions used throughout this section.

- *Variable definitions*

The read-back channel and associated identification parameters are illustrated in Figures 1(a) and 1(b). The continuous read-head output signal, $d(t)$, can be modeled in one of two ways [1]:

$$d(t) = \sum_k x_k h(t - kT) + u(t), \tag{1a}$$

$$d(t) = \sum_k s_k h_s(t - kT) + u(t), \tag{1b}$$

where $h(t)$ and $h_s(t)$ are the unknown linear time-invariant pulse and step responses, respectively, and $u(t)$ denotes an uncorrelated, additive, zero-mean noise,† x_k takes on the values ± 1 (or $+1$ and 0 for some optical storage systems), corresponding to 1's and 0's, respectively, in the stored data sequence at time kT, $1/T$ is the data rate, and k is an integer. In Equation (1b), s_k can take on the values ± 2 or 0 (± 1 or 0 for optical) according to the relation

$$s_k = x_k - x_{k-1}. \tag{2}$$

Likewise, one determines for a linear channel

$$h(t) = h_s(t) - h_s(t - T). \tag{3}$$

It is a property of the method presented that the estimates also obey Equation (3); however, it is sometimes informative to separately identify $h(t)$ and $h_s(t)$, rather than identify only one and compute the other from it. It is assumed that $d(t)$ is digitized at some rate T_d, such that

$$T_d \triangleq \frac{T}{p}, \tag{4}$$

where p is an integer (≥ 1) oversampling factor. This restriction is relaxed to a rational fraction in Appendix A. The sampled read-head output is then, with $t = mT_d$ in (1),

$$\begin{aligned} d(mT_d) &= \sum_k x_k h(mT_d - kT) + u(mT_d) \\ &= \sum_k x_k h[(m - kp)T_d] + u(mT_d) \end{aligned} \tag{5a}$$

† Even though the assumption that the noise is additive may not be completely true in practice, our objective is to find the values for the parameters in such a model that most closely approximate the measured responses, and deviations from such a model appear in the final results of the method in this paper.

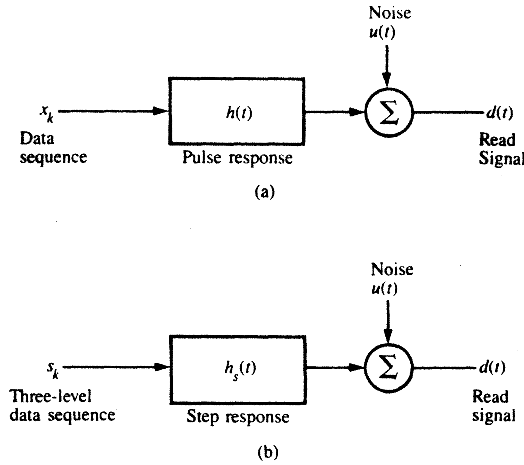


Figure 1

Summary of storage quantity definitions (a) for pulse responses and (b) for step responses.

or‡

$$\begin{aligned} d(mT_d) &= \sum_k s_k h_s(mT_d - kT) + u(mT_d) \\ &= \sum_k s_k h_s[(m - kp)T_d] + u(mT_d). \end{aligned} \tag{5b}$$

The channel is estimated by

$$\hat{d}(mT_d) \triangleq \sum_k x_k w(mT_d - kT), \tag{6}$$

where $w(t)$ is a linear filter response whose sampled values at times mT_d are to be computed via the channel identification procedure [ideally $w(t) = h(t)$]. Likewise, for the step response, the estimate is

$$\hat{d}_s(mT_d) \triangleq \sum_k s_k w_s(mT_d - kpT_d). \tag{7}$$

We also define an error signal

$$\varepsilon(mT_d) \triangleq d(mT_d) - \hat{d}(mT_d). \tag{8}$$

As an example, note that, if x_k or s_k is a sequence corresponding to an isolated pulse or transition input, (5a) and (5b) reduce to

$$d(mT_d) = h(mT_d) + u(mT_d) \tag{9a}$$

or

$$d_s(mT_d) = h_s(mT_d) + u(mT_d), \tag{9b}$$

respectively, the desired pulse shapes in noise. Then, $w(mT_d)$

‡ The reader may note that (5a) and (5b) are equivalent to p subchannels, each at spacing T; this observation is exploited to reduce computation in the new procedure in Section 3.

and $w_s(mT_d)$ can be estimated by the averages

$$w(mT_d) = \frac{1}{n} \sum_{k=1}^{n} d(mT_d;k), \tag{10a}$$

$$w_s(mT_d) = \frac{1}{n} \sum_{k=1}^{n} d_s(mT_d;k), \tag{10b}$$

where the index k denotes the kth experiment. That is, one measures the response n times and averages, which is the basis for the aforementioned isolated step and dibit identification methods. Some deficiencies of the estimates identified via such isolated step or pulse methods are discussed later. Equations (9) and (10) were given only to verify the utility of the definitions in (1)–(8). We now proceed with a discussion of the least-squares channel-identification procedure.

- *The application of least squares*

In the least-squares identification procedure, a known data pattern is written on the storage device. The $w(mT_d)$ are chosen to minimize

$$\xi_l = \sum_{m=1}^{l} \varepsilon(mT_d)^2, \tag{11}$$

where $\varepsilon(mT_d)$ is given in (8). If we denote $W_{M,l}$ by the $M \times 1$ column vector

$$W_{M,l} \triangleq \begin{bmatrix} w_l(0) \\ \cdot \\ w_l[(M-1)T_d] \end{bmatrix}, \tag{12}$$

then the solution to (11) is conveniently written [2]

$$W_{M,l} = \Big(\sum_{m=1}^{l} X_{M,m}X'_{M,m}\Big)^{-1} \Big(\sum_{m=1}^{l} X_{M,m}d(mT_d)\Big), \tag{13}$$

where $'$ denotes transpose, and

$$X_{M,m} \triangleq \begin{bmatrix} x_m \\ \cdot \\ x_{m-M+1} \end{bmatrix} \tag{14}$$

for $p = 1$. There are $p - 1$ zeros between entries in (14) if $p > 1$. We have further assumed that M is large enough to span the nonzero extent of the pulse (step) response in intervals of sampling periods or $MT_d = NT$ data periods containing p samples each, $M = Np$. Equation (13) can be rewritten

$$W_{M,l} = R_{M,l}^{-1}P_{M,l}, \tag{15}$$

where

$$R_{M,l} \triangleq \frac{1}{l} \sum_{m=1}^{l} X_{M,m}X'_{M,m},$$

$$P_{M,l} \triangleq \frac{1}{l} \sum_{m=1}^{l} X_{M,m}d(mT_d). \tag{16}$$

A similar expression holds for the step response, with x's replaced by s's and w's replaced by w_s's in the solution. Note that $M \times M$ matrix inversion is explicit in (13); however, because of the special structure in this problem, no matrix need ever be inverted directly. For more details, see Section 3 and especially [2].

- *A performance measure*

The mean of $W_{M,l}$ can be easily determined as

$$E[W_{M,l}] = H_M = \begin{bmatrix} h(0) \\ \cdot \\ \cdot \\ h[(M-1)T_d] \end{bmatrix}, \tag{17}$$

the desired solution, when the above least-squares method is used. The Norm Tap Deviation is a mean-square measure of statistically how far the estimated $W_{M,l}$ is from $H_{M,l}$ and is also easily computed, if $u(t)$ is white (spectrally flat over the frequency range of interest), as

$$\theta_{M,l} = E[\| W_{M,l} - H_{M,l}\|^2] = \frac{1}{l} \operatorname{trace}(R_{M,l}^{-1})\sigma_u^2, \tag{18}$$

where

$$\sigma_u^2 \triangleq E[u(KT_d)^2]. \tag{19}$$

We show in the next few sections that both the isolated transition (or dibit) and DFT methods are special cases of the general least-squares method with very special restrictions on the input sequence and on M and l. Thus, we are able to use (18) as a performance indicator for those methods as well.

- *Isolated transition example and analysis of resolution*

As an example, once again consider an isolated dibit; then $X_{M,m}$ has only one nonnegative entry per column and (13) reduces to (using generalized inverses, see [3])

$$W_{M,l} = \begin{bmatrix} d(MT_d) \\ \cdot \\ d(T_d) \end{bmatrix}. \tag{20a}$$

A string of n "isolated" (far enough apart) dibits occurring within a large data record (length l) has a least-squares solution,

$$W_{M,l} = \frac{1}{n} \sum_{k=0}^{n-1} \begin{bmatrix} d(kMT_d + T_d) \\ \cdot \\ d(kMT_d + MT_d) \end{bmatrix}, \tag{20b}$$

that is exactly the same as the isolated pulse solution in (10a). The least-squares identification procedure is more general in that the input need not be an isolated transition or dibit.

Equation (18) allows us to compare the quality of the least-squares estimates of $H_{M,l}$ ($W_{M,l}$) for different input sequences. Note that, for a string of n isolated (MT_d apart, so $l = Mn$) inputs, one determines for white zero-mean u

$$E\| W_{M,l} - H_{M,l}\|^2 = \frac{M}{n}\sigma_u^2. \tag{21}$$

Pseudorandom sequences are generally desirable [4, 5] for channel inputs because of their broadband spectral response. An identity for $R_{M,l}$ can easily be determined, if l = length of the pseudorandom sequence,§ as (see [6–8])

$$R_{M,l} = \frac{1}{l}[(l+1)I_M - 1_M 1'_M], \tag{22a}$$

where 1_M is an $M \times 1$ vector of M ones. One can also easily show that

$$R^{-1}_{M,l} = \frac{l}{l+1}\left(I_M + \frac{1}{l-M+1}1_M 1'_M\right). \tag{22b}$$

Thus, (18) becomes, for a pseudorandom sequence of length M repeated n times,

$$\theta_{M,l} = \frac{1}{nM+1}\left\{\frac{(n-1)M^2+2M}{(n-1)M+1}\right\}\sigma_u^2 \rightarrow \frac{1}{n}\sigma_u^2. \tag{23}$$

For $n = 1$, there is an improvement of $(M+1)/2$ with respect to (21). As n increases to a large value, there is an improvement by a factor of M in estimate quality, or equivalently, M more digitized outputs from isolated dibits must be processed in the isolated dibit identification schemes to get the same resolution estimates as those produced by least squares with a pseudorandom length-M input. For oversampling ($p > 1$), the comparison favors the pseudorandom input by the same amount. Heuristically, when using pseudorandom or "scrambled" data in channel identification, the input is more spectrally "rich" and all frequencies are more equally weighted than when a single pulse is used. The resulting flat nature of the spectrum results in the inverse autocorrelation matrix being close to an identity which makes $\theta_{M,l}$ in (18) smaller (better). When x_k has a flat spectrum, s_k does not have a flat spectrum, but a similar slightly more complex argument can be given to justify the least-squares improvements.

In practice, it may not be difficult to average the extra data for the isolated input method. However, there is another very practical advantage of using more random data, as was first noted by C. M. Melas [9]. This is that in the isolated transition or isolated dibit methods, the AGC (Automatic Gain Control) must be removed from the channel to prevent the sudden change in energy associated with the isolated input from suddenly varying the gain parameter of the AGC. Then, the identified pulse characteristics will not include the effect of the AGC. This effect can commonly be more than a simple gain factor and is determined by the bandwidth and tracking rate of the AGC.

- *Comparison with frequency-domain methods*

Another more recent method used in storage-channel identification is [4, 5, 10] to compute the DFT of the response to some prescribed pattern written on the media. In order to invert the DFT to get a time-domain estimate of the pulse response, one must first divide the measured DFT by the DFT, *including phase*, of the input before the inverse DFT, which [10] also observes. Using this last restriction, one can also generalize the methods of [4, 5, 10] to estimate the channel response for any inputs, including the ±2, 0 normally associated with identification of the step (transition) response.

Nevertheless, with the division by input spectra, the frequency-domain method is the same as the time-domain least-squares method of this paper if $M = l$, and as we shall see, the case $M = l$ gives very poor estimate quality. In the case that $u(mT_d)$ is white and Gaussian, the least-squares method (see [11]) achieves the famed Cramer-Rao bound for a fixed l and M; that is, no other estimator has higher resolution for the given data. If the assumption on $u(t)$ is just white (not also necessarily Gaussian), then the least-squares estimator is a Best Linear Unbiased Estimator (BLUE) [3].

Theoretically, the difference between the DFT technique and the time-domain least-squares method can be quantified via the following analysis. It is usually wise to pick $M < l$ so as to introduce more noise averaging, or equivalently, to make the Cramer-Rao bound lower for fewer parameters. Generally speaking, in any estimation scheme, we desire $l > M$ to get good quality estimates. Nevertheless, picking M too small can introduce extraneous harmonic distortion in the estimated step response. The time-domain least-squares method can be rewritten as that $W_{M,l}$ that minimizes [2]

$$\xi_{M,l} = \underline{\varepsilon}'_{M,l}\underline{\varepsilon}_{M,l} = \|\underline{\varepsilon}_{M,l}\|^2, \tag{24}$$

where

$$\underline{\varepsilon}_{M,l} = \underline{d}_{l,l} - \underline{X}_{M,l,l}W_{M,l} \tag{25}$$

and

$$\underline{d}_{l,k} = \begin{bmatrix} d(kT_d) \\ \cdot \\ d[(k-l+1)T_d] \end{bmatrix}; \qquad \underline{x}_{l,k} = \begin{bmatrix} x_k \\ \cdot \\ x_{k-l+1} \end{bmatrix}, \tag{26a}$$

where $p - 1$ zeros can be inserted between nonzero entries in $\underline{x}_{l,k}$ and

$$\underline{X}_{M,l,k} = [\underline{x}_{l,k}, \underline{x}_{l,k-1}, \cdots, \underline{x}_{l,k-M+1}]. \tag{26b}$$

The DFT-based method is a special case of a linear $M \times l$ transformation on $\varepsilon_{M,l}$, that is, let

$$\underline{E}_{M,l} = \phi\underline{\varepsilon}_{M,l}, \tag{27}$$

where ϕ is an $M \times l$ (possibly complex) matrix representing the linear transformation. Then

$$\underline{E}^*_{M,l}\underline{E}_{M,l} = \underline{\varepsilon}^*_{M,l}\phi^*\phi\underline{\varepsilon}_{M,l}, \tag{28}$$

where * denotes conjugate transpose. If ϕ is a unitary transformation ($\phi^*\phi = I$), then

§ Even when the output is oversampled, we show later that the only autocorrelation matrix of interest is at the data rate; thus all of the analysis here is also valid for $p > 1$.

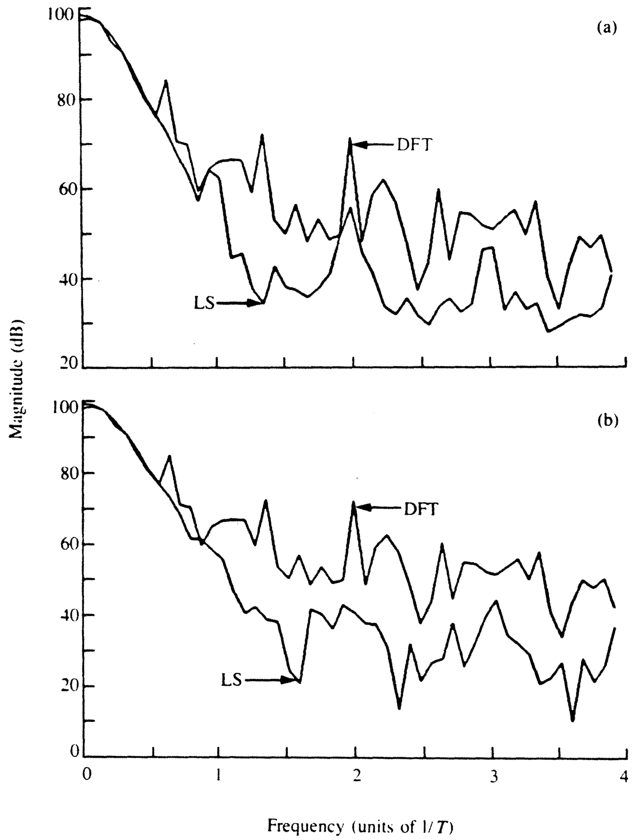


Figure 2

Comparison of DFT and least squares for (a) 8-bit and (b) 16-bit periods.

$$\|\underline{E}_{M,l}\|^2 = \|\underline{\varepsilon}_{M,l}\|^2, \tag{29}$$

and the minimized $\underline{\varepsilon}_{M,l}$ is obtained by

$$\underline{\varepsilon}_{M,l} = \phi^* \underline{E}_{M,l} \,. \tag{30}$$

In the DFT methods of [4, 5], the matrix ϕ is chosen, *under the very special assumptions that $M = l$ and the input is periodic (pseudorandom) of length $l = M$, as*

$$\phi = \phi_M = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 1 & \cdot & 1 \\ 1 & e^{-j\omega_1 T} & \cdot & e^{-j\omega_1 (M-1)T} \\ \cdot & \cdot & \cdot & \cdot \\ 1 & e^{-j\omega_{M-1} T} & \cdot & e^{-j\omega_{M-1}(M-1)T} \end{bmatrix}, \tag{31}$$

where

$$\omega_i = i\,\frac{2\pi}{lT} \qquad i = 0, \cdots, M-1. \tag{32}$$

ϕ_M can easily be shown to be unitary [12], so the relation in (29) holds, apparently yielding the time-domain least-squares solution. ϕ^*_M is the inverse DFT in this case. However, in the time-domain method of this paper, M is much less than l to average the effects of noise and other nonideal effects.

Using our performance measure in (18) and (23) ($n = 1$, $l = M$) again, one determines the estimate quality as

$$\theta_{l,l} = \frac{2l}{l+1}\sigma_u^2 \,, \tag{33}$$

while the general formula for a pseudorandom sequence of length l with M parameters is

$$\theta_{M,l} = \frac{2M + Ml - M^2}{(l+1)(l-M+1)}\sigma_u^2 \,. \tag{34}$$

Substitution of $l = 10M$, a good practical rule of thumb, into (34) yields the advantage

$$\frac{\theta_{l,l}}{\theta_{M,l}} = \frac{2l(0.9l+1)}{0.09l^2 + 0.2l} = 20\,\frac{(0.9l+1)}{(0.9l+2)} \,. \tag{35}$$

Even for $l \cong 1000$, another reasonable number, the improvement in (35) is close to its limiting value of 20. This large improvement is typically evident when comparing the spectra of a pulse produced by the time-domain least squares and by the DFT method, as we have illustrated in **Figures 2(a)** and **2(b)**. Note from the level of "frequency ripple" in the DFT plot that the time-domain least squares is at least an order of magnitude improvement. Also note the lower "noise level" at higher frequencies with the least-squares identification procedure. It is also important to note that $l = M = 2^i - 1$ (i a positive integer) for a pseudorandom input, which, at least, requires special attention for efficient DFT implementation [12–14]. The reason for the two different lengths (M's) in Figures 3(a) and 3(b) is discussed later.

• *An averaged DFT identification scheme*

Here, we propose an averaged DFT method for the special case that $l = nM$, where n is an integer greater than 1, and the input sequence is periodic with period M. [The case of oversampling ($p > 1$) is identical for each of the subchannels (see Section 3).] There is a very special set of circumstances when the inverted matrix in (13) is Toeplitz and DFTs can be used. Generally, (13) is not Toeplitz and DFTs are not appropriate. This method is equivalent to least squares, as can be seen from the following. Define ϕ_l by

$$\underline{\phi}_l \triangleq \begin{bmatrix} \phi_M & 0 & \cdot & 0 \\ 0 & \phi_M & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \phi_M \end{bmatrix}. \tag{36}$$

Multiplication by ϕ is equivalent to n M-point DFTs performed on the n groups of M inputs. Note that $\underline{\phi}_l$ is unitary,

$$\underline{\phi}_l \underline{\phi}_l^* = I. \tag{37}$$

The least-squares estimates in the frequency domain are given for each frequency bin by

$$\nu(k) = \frac{\sum_{i=1}^{n} \delta(k, i)\chi^*(k)}{\sum_{i=1}^{n} \chi(k)\chi^*(k)} = \frac{1}{n}\frac{\sum_{i=1}^{n} \delta(k, i)}{\chi(k)}$$

$$k = 0, \cdots, M - 1, \quad (38)$$

where $\nu(k)$ and $\chi(k)$ are the M-point DFTs of $W'_{M,l}$ and $Y_{M,M}$, respectively. $\delta(k, i)$ is the M-point DFT of the time series d_k in the ith of the n groups. Equation (38) is really the average of n uses of the original DFT method, when a period-M input is recycled to fill l time periods. Then, some averaging will be introduced, in the optimal least-squares sense, into the DFT identification scheme. The method of (36) and (38), because of (37), is equivalent to an l-point least-squares time-domain procedure. Of course, an inverse DFT on the quantities in (38) must be performed to obtain the desired time-domain parameters, $W_{M,l}$. This method requires the unnecessary imposition of an integer ratio restriction on l and m, which is not required in the more general and straightforward time-domain least-squares solution (13).

• *A note on maximum-likelihood detection schemes*

The identified responses can be used in Maximum-Likelihood Sequence Detection (MLSD) [15, 16]. In this case, the Mean Square Error (MSE) is a more useful estimate of performance than (18). It is shown in [17] that (given a certain input sequence)

$$MSE = E[\varepsilon^2(mT_d)] = \sigma_u^2 \gamma_{M,l}, \quad (39)$$

where $\gamma_{M,l}$ is given by

$$\gamma_{M,l} = 1 - X'_{M,l} R^{-1}_{M,l} X_{M,l} \quad (40)$$

and $'$ denotes transpose. One also can show (see [17]) that

$$0 \le \gamma_{M,l} \le 1; \quad (41)$$

thus, the worst (because the desired value is σ_u^2) MSE after M measurements is

$$MSE_{worst} = 0, \quad (42)$$

which is exactly the value given by a length-M pseudorandom sequence. In fact, it is shown in [8] that choices for x_k other than length-M pseudorandom sequences can yield MSE between 0 and σ_u^2 after M data points, while still maintaining good (low) $E[\| W_{M,l} - H_{M,l} \|^2]$. Thus the length-$M$ pseudorandom sequence may not be the best training sequence if MLSD is used. Some data with statistics equivalent to what is expected in actual use would be the best choice for MLSD and other similar sampling detection schemes.

• *Signal-to-noise ratio estimation*

The SNR for the read-head response can be estimated (when $M \ll l$) by

$$SNR \cong \frac{(l - N) \| W_{M,l} \|^2}{p \cdot \xi_{M,l}}, \quad (43)$$

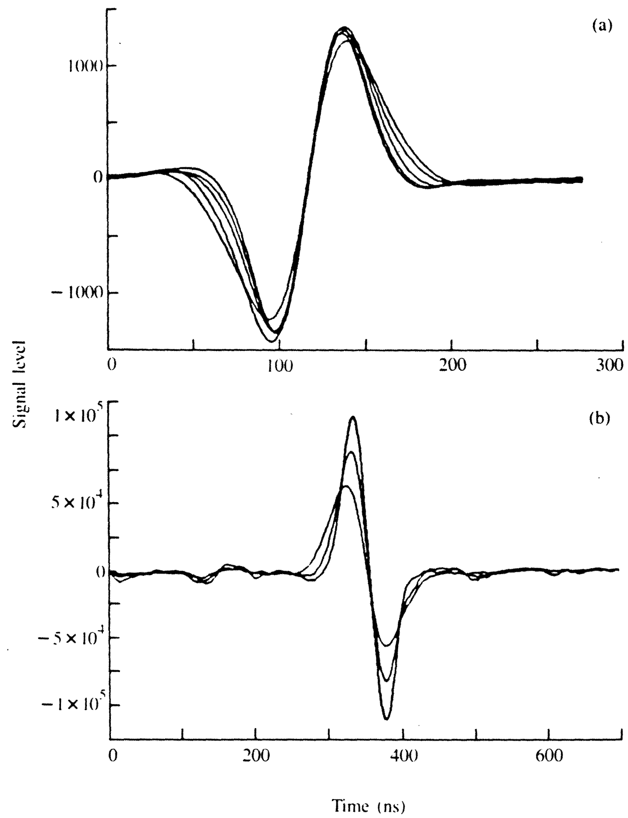
where $\| W_{M,l} \|^2/p$ is the signal power for the binary input to the pulse response, and $\xi_{M,l}/l - N$ is the noise power. However, one must ensure that data measured at the read-head output have NOT BEEN AVERAGED before digitizing to ensure a meaningful estimate in (43). Also, as Howell [4] has noted, that distortion in the measuring devices, particularly the nonlinearities in the CRT sweep rate if a storage scope is used, can add appreciable noise not inherent in the actual storage channel. Of course, such contamination would leave (37) as a measure of the mean-square distortion in the measuring procedure, rather than the desired channel noise + media noise + modeling mean-square errors. Even if measurements are carefully taken, (43) is usually more indicative of the levels of nonlinear mismatch to the model and can therefore be very useful in evaluating the potential success or failure of advanced detection schemes.

• *Determination of M*

We have previously assumed that the order M (number of identified parameters) was overestimated or known *a priori*. However, the best quality estimate for l data points is given by the so-called "Minimum Description Length" principle of [18], which jointly estimates M and the corresponding $W_{M,l}$ for l-points. The improvement in the general storage-channel identification problem is negligible if $l \ge 10M$. It is interesting to understand just what happens if M is chosen too small. Suppose $h(kT_d) \ne 0$ for $k < 0$, $k > M$. Then $u(kT_d)$ can be modeled as the sum of white noise and the distortion caused by the neglected terms in h. This second distortion term is just a linear filter acting on the pseudorandom pattern. When oversampled, the output of such a filter is the product of its transfer function and the transform of the oversampled pseudorandom pattern. The response of the oversampled pseudorandom pattern can easily be shown to be maximum at multiples of $1/T$, thus explaining why choosing M larger in Figure 2(b) than in Figure 2(a) caused the "harmonics" to disappear. Of course, picking M too large as in the DFT methods has a far more distorting effect on the output because of the lack of noise averaging. Generally speaking, conservative values for M and l are 15 bit periods and $l = 10M$, respectively.

• *Summary*

In this section, we have introduced the least-squares channel-identification procedure, compared its performance with other commonly used procedures, and found the least-squares method superior in the quality of estimates that it produces. We now turn to implementation/programming of this new procedure.



Pulse responses at 27 Mb/s for (a) thin-film medium and thin-film head and (b) particulate medium and thin-film head.

3. Efficient implementation of the off-line least-squares identification procedure

The time-domain least-squares solution is described using a matrix inverse in (13). This matrix can be large, requiring large storage and long processing time in an off-line computer program implementing the inversion. However, matrix inversion can be avoided to simplify the determination of $W_{M,l}$. This section describes several special features of the least-squares procedure that can be used to reduce considerably the computation and storage in an off-line implementation. Such simplifications could also become important if the characteristics of each particular storage device, and possibly at several different radii on each, were to be computed during the manufacturing process either for identifying defective devices or for optimization of the channel-detection circuitry for each particular unit. An efficient on-line procedure, similar to that of [8], is suggested in Appendix B.

• *Subchannels*

In most cases of practical interest, the oversampling factor p is greater than one. Then, one writes $mT_d = nT + iT_d$, where

$$n = \lfloor \left(\frac{m}{p}\right), \tag{44}$$

where $\lfloor$ denotes the "greatest integer less than," and i takes the values $0, \cdots, p-1$. Equation (5a) is rewritten [(5b) can be similarly rewritten]

$$d(nT + iT_d) = \sum_k x_k h[(n-k)T + iT_d] + u(nT + iT_d). \tag{45}$$

The index i has no effect upon the convolution operation, and the p phases of $d(mT_d)$ per sample period, $T = pT_d$, are described by

$${}^i d_n = \sum_k x_k {}^i h_{n-k} - {}^i u_n \qquad i = 0, \cdots, p-1, \tag{46}$$

where the ${}^i h_n$ are i independent "subchannels." With minor algebra, one can reduce the least-squares identification procedure to p subprocedures that can all be solved separately. The p solutions can be interspersed to obtain $W_{M,l} = W_{Np,l}$, where $N = M/p$ (we assume that p divides M or that M is picked slightly larger so that it does). Then, only one $N \times N$ matrix need be inverted (it is the same for all subchannels), rather than one $M \times M$ matrix, a considerable computational and storage saving. This matrix is the autocorrelation matrix of the input data alluded to in an earlier footnote (§). However, much greater savings are also available.

• *Use of fast algorithms*

The most efficient solution to the *general* least-squares identification problem appears in [2]. The DFT cannot be used in the general least-squares filtering problem because a Toeplitz structure must be imposed on (13) for its use. This solution requires about

$$\left(\frac{p+1}{p}\right) lN + 4.5N^2 + pN^2 \tag{47}$$

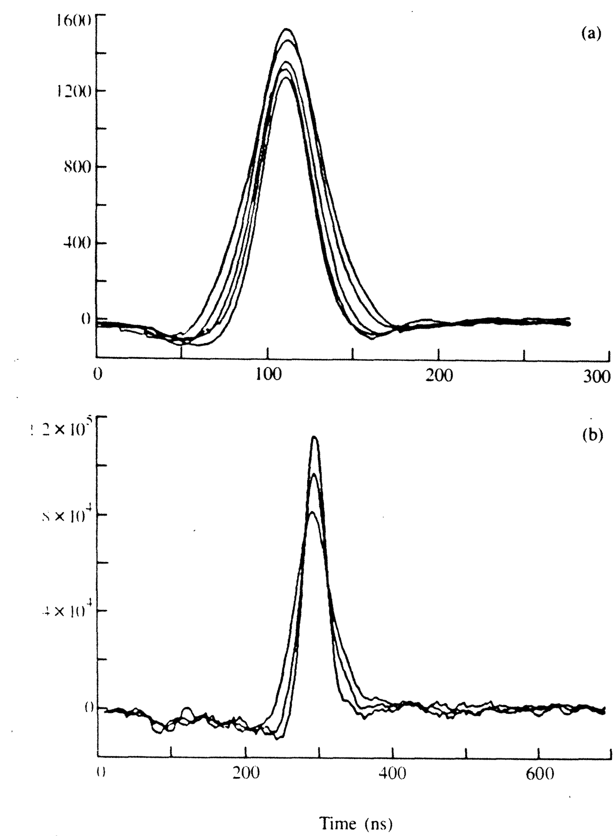
multiplications, divisions, and additions in comparison to $O(N^3)$ for straightforward matrix inversion. [$O(x)$ is a number that asymptotically rises no faster than in direct proportion to x.] The term $(lN/p) + 4.5N^2$ is the fixed cost of the equivalent of inverting the matrix $R_{N,l/p}$ (fixed because it is the same for each subchannel); the remaining term $pN^2 + lN$ is the additional cost, at $N^2 + lN/p$ per subchannel, for computing the equivalent of the product $R^{-1}_{N,l/p}\,{}^i P_{N,l/p} = {}^i W_{N,l/p}$ for each of the p subchannels. The storage requirements are about $6N + 2l$ locations for the algorithm in [2]. The cost reductions accrue to the shifted nature of $X_{N,k}$ with respect to $X_{N,k-1}$, or equivalently, that $R_{N,l/p}$ can be rewritten as a product of Toeplitz matrices,

$$R_{N,l/p} = \underline{X}'_{N,l/p,l}\underline{X}_{N,l/p,l}, \tag{48}$$

where $X_{N,l,k}$ is defined in (26b). For more details, see [2].

• *Choice of the input sequence*

Further computational and storage reductions are possible if the length-l sequence $x_{l,k}$ is chosen beforehand for all storage



Step responses for (a) thin-film medium and thin-film head and (b) particulate medium and thin-film head.

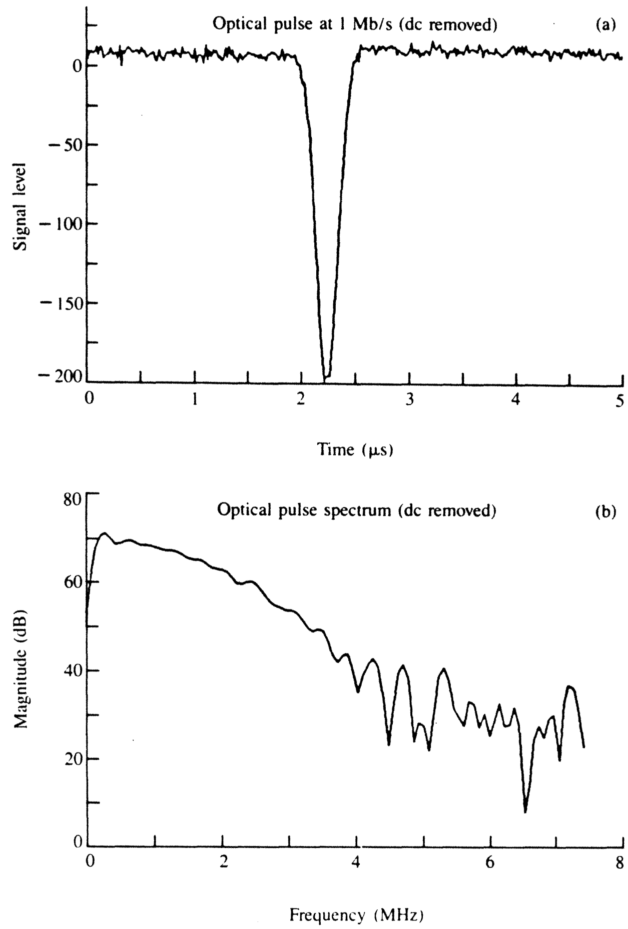


Figure 5

(a) Pulse response of optical medium at 1 Mb/s (dc removed) and (b) Fourier transform of pulse response.

channels to be identified. A currently popular choice is a 63-bit pseudorandom sequence. When the input data sequence is known beforehand, many of the quantities in the Fast (BFTF) algorithm of [2] can be precomputed and stored once, reducing computation to

$$pN^2 + lN \tag{49}$$

multiplications and additions (no divides) and storage (random access) to about

$$2N + l \tag{50}$$

locations. Neither these counts nor the counts in (47) and (49) can be matched by the DFT or other methods of comparable estimate quality for reasonable N (20 or less). Asymptotically, because of the $N \log_2 N$ computation in FFT implementations of the DFT, these FFT methods may have an advantage in terms of computational requirements, but N is never chosen that large in practice.

- *Experimental results*

To demonstrate the robustness of the new least-squares identification method, several channel pulse shapes are plotted in **Figures 3(a)** and **3(b)**, while the corresponding steps are plotted in **Figures 4(a)** and **4(b)**. These responses were obtained using the new procedure for a 63-bit pseudorandom sequence on digitized measurements of a thin-film disk/thin-film head channel [Figures 3(a) and 4(a)], and on a particulate disk/thin-film head channel [Figures 3(b) and 4(b)]. The measurements were taken at several different diameters on each device. The diameters for Figures 3(a) and 4(a) were 105, 120, 135, 150, and 165 mm, while those for 3(b) and 4(b) were 103, 136, and 172 mm. **Figures 5(a)** and **5(b)** show the pulse response and its spectrum, respectively, for an optical storage device. In **Figures 6(a)** and **6(b)**, we have plotted pulse and step responses for a magnetoresistive head in a magnetic-tape system; this time a 62-bit pattern corresponding to NRZI coding of two cycles of a 31-bit pseudorandom data pattern was used [10]. In **Figure 7**, the delay for the magnetoresistive head is plotted to illustrate the ability of the new least-squares identification

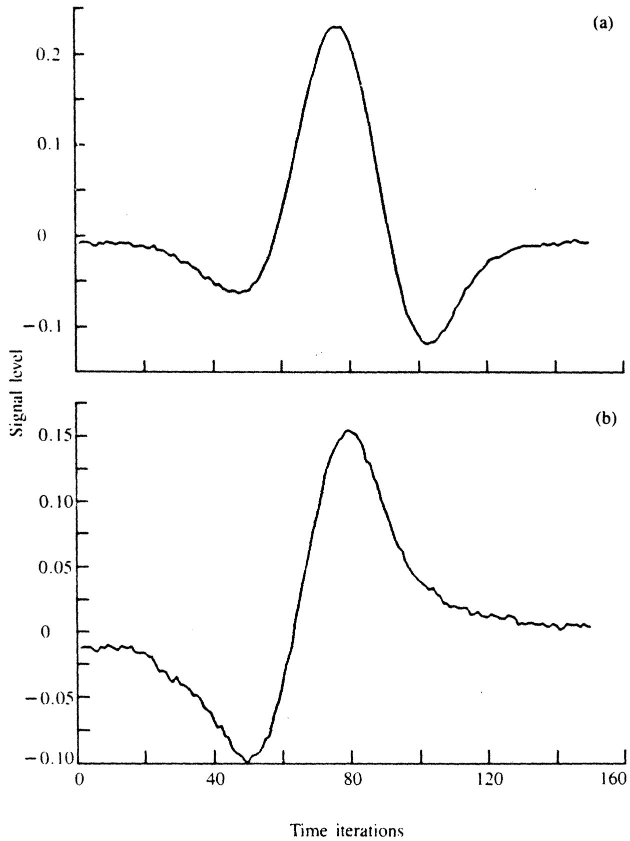


Figure 6

(a) Pulse response and (b) step response of magnetoresistive head.

procedure to capture that quantity as well. The dc level was removed from the desired signal for the optical device to facilitate inspection of the plots; the true optical channel is a baseband channel. The plots in Figures 3, 4, and 5 demonstrate the robust utility of the least-squares procedure.

4. Conclusions

In this paper, we have introduced a new least-squares storage-channel identification procedure. We have analyzed the procedure thoroughly and demonstrated via experiment its utility and its improvements over existing methods. Several methods for reducing the implementational cost of the procedure were also discussed. The procedure can become a uniform standard for identifying and comparing the channel characteristics of various storage media.

Acknowledgments

I would like to thank L. Barbosa, F. Dolivo, M. Haynes, K. Hense, T. Howell, R. Lynch, C. M. Melas, P. Seger, W. Schott, P. Siegel, H. Thapar, D. Turner, G. Ungerboeck, and J. E. Vaughn for their technical comments and/or data.

Appendix A: Arbitrary sampling rates

In this appendix, the sampling interval T_d is permitted to take the values

$$T_d = \frac{q}{p} T, \tag{A1}$$

where q and p are relatively prime positive integers such that $q < p$. Any arbitrary ratio of sampling to data rates can be as closely approximated as desired by the relation in (A1), as long as it is known, which implies some synchronization between digitizer and write clock. We also define a smaller time interval τ by

$$\tau = \frac{T_d}{q} = \frac{T}{p} \tag{A2}$$

or

$$qp\tau = pT_d = qT. \tag{A3}$$

The samples at rate T_d can be organized into successive disjoint sets of p members and of duration $pT_d = pq\tau$. Then any sampling instant mT_d can be rewritten as

$$mT_d = npq\tau + iq\tau = (np + i)T_d \qquad i = 0, \cdots, p - 1. \tag{A4}$$

The equivalent of (37) becomes

$$d[npq\tau + iq\tau] = \sum_k h(npq\tau - kp\tau + iq\tau)x_k + u(npq\tau + iq\tau). \tag{A5}$$

Note that, if p and q are relatively prime, as was assumed earlier, then h will be specified at intervals of τ in (A5), or equivalently at all time instants that are integer multiples of τ. At sample i within each group of p samples, only the values $h(kp\tau + iq\tau)$, where k is any integer, contribute to

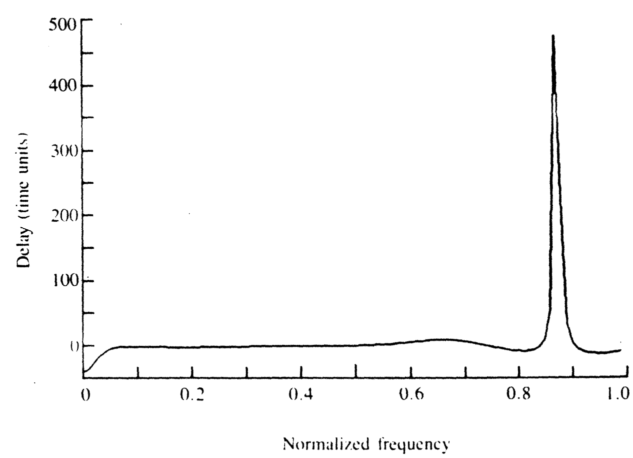


Figure 7

Delay characteristic for tape pulse response.

$d(npq\tau + iq\tau)$. Thus, d has p phases per group of p samples that can be independently modeled as

$$^{i}d_{np} = \sum_{k} x_k {}^{i}h_{nq-k} - {}^{i}u_{np} \qquad i = 0, \cdots, p-1, \tag{A6}$$

where, again,

$$^{i}h_n = h(nT + iT_d) \tag{A7}$$

and

$$^{i}d_n = d(nT_d + iT_d); \qquad ^{i}u_n = u(nT + iT_d) \tag{A8}$$

for $i = 0, \cdots, p-1$. Each of the subchannels can be identified independently and the resultant responses overlaid (with delays of τ with respect to one another). The overall response can then be used directly or decimated to $p\tau$ (the data rate), $q\tau$ (the sampling rate), or any other integer divisor of the rate $1/\tau$. An important point to note is that there is a loss in resolution of a factor of approximately q for any fixed data length l with respect to the case where $T_d = T/p$. This last fact makes the alternative of resampling the data or phase-locking the ADC used to acquire the data (set $q = 1$) very attractive from a performance viewpoint.

Appendix B: On-line efficiency

It is possible to implement the least-squares storage-channel identification method in a sample-recursive manner. The procedure becomes a special case of the one considered previously by this author for echo cancelers in data transmission in [8]. The storage identification procedure could be performed on line, for example, to initialize, and possibly update (see [15, 16], a Maximum-Likelihood Sequence Detection Circuit.

A brief summary of the procedure is, where k is the recursive time index,

$$W_{M,k} = W_{M,k-1} + \varepsilon^{P}_{M,k} \cdot C_{M,k}\,, \tag{B1}$$

and where

$$\varepsilon^{P}_{M,k} = d(k) - W'_{M,k-1}X_{M,k}\,. \tag{B2}$$

$C_{M,k}$ is an $M \times 1$ function of the input (presumably known or "training") data sequence and is given by

$$C_{M,k} = \Big(\sum_{m=0}^{k} X_{M,k}X'_{M,k}\Big)^{-1} X_{M,k}\,, \tag{B3}$$

and is presumably precomputed and stored prior to use. For more details on this procedure, and for an efficient recursive computation of $C_{M,k}$ when there is no prespecified training sequence, see [2, 8, 17, 19]. A final note is that, if the signal written just prior to the start of training is an erasure, then the prewindowed exact-initialization method of [8, 17] applies, rendering extremely low computational requirements: (B1) and (B3) simplify dramatically in that case.

Appendix C: Methods for nonlinear identification

The study of nonlinear identification of a data channel is an entire subject area in itself. For instance, one can refer to [20] and [21] for methods based on simplification of Volterra series under the constraints of a binary input. Here, a simple method suffices to verify the presence/absence of appreciable nonlinearities and to roughly quantify their magnitudes relative to the linear component of the channel response.

SNR measurement

Estimation of the SNR was discussed earlier. The minimized sum of squared errors, $\xi_{M,l}$, contains a component due to modeling error. If M is sufficiently large, most of this modeling error is due to nonlinearities. The size of the SNR is indicative of the level of nonlinearities. Generally speaking, SNRs well below those expected can be indicative of large modeling errors due to nonlinearities. Thus, one can use the size of the SNR as an indicator of nonlinearities, given that he has some prior experience with the particular media and head and knows what to expect in terms of a nominal SNR value. This type of procedure requires a very accurate phase-lock to the underlying data rate to ensure that nonlinearities are not artificially inserted by sampling-phase errors in the measurement process.

References

1. D. G. Messerschmitt, "A Study of Sampling Detectors for Magnetic Recording," University of California at Berkeley, private communication.
2. J. M. Cioffi, "The Block-Processing FTF Adaptive Algorithm," *IEEE Trans. Acoust., Speech, & Signal Proc.* **ASSP-34,** No. 1, 77–90 (February 1986).
3. C. L. Lawson and R. J. Hanson, *Solving Least-Squares Problems*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1974.
4. T. D. Howell, IBM Research Division, San Jose, CA, private communication.
5. G. Ungerboeck, IBM Research Division, Zurich, Switzerland, private communication.
6. J. Salz, "On the Start-Up Problem in Digital Echo Cancellers," *Bell Syst. Tech. J.* **6,** No. 2, Part 1, 1353–1364 (July–August 1983).
7. M. L. Honig, "Echo Cancellation of Voiceband Data Signals Using RLS and Stochastic-Gradient Algorithms," *IEEE Trans. Commun.* **COM-33,** No. 1, 65–73 (January 1985).
8. J. M. Cioffi and T. Kailath, "An Efficient, RLS, Data-Driven Echo Canceller for Fast Initialization of Full-Duplex Data Transmission," *IEEE Trans. Commun.* **COM-33,** No. 7, 601–611 (July 1985). See also *Proceedings of ICC'85*, June 1985, Chicago.
9. C. M. Melas, IBM Research Division, San Jose, CA, private communication.
10. M. K. Haynes, "Experimental Determination of the Loss and Phase Transfer Functions of a Magnetic Recording Channel," *IEEE Trans. Magnetics* **MAG-13**, No. 5, 1284–1286 (September 1977).
11. G. C. Goodwin and R. L. Payne, *Dynamic System Identification*, Academic Press, Inc., New York, 1977.
12. D. F. Elliot and K. Ramamohan Rao, *Fast Transforms: Algorithms, Analyses, Applications*, Academic Press, Inc., New York, 1982.
13. R. E. Blahut, *Fast Algorithms for Digital Signal Processing*, Addison-Wesley Publishing Co., Reading, MA, 1985.

14. H. J. Nussbaumer, *Fast Fourier Transforms and Convolution Algorithms*, Springer-Verlag, Berlin, 1981.
15. G. Ungerboeck, "Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems," *IEEE Trans. Commun.* **COM-22**, No. 5, 624–636 (May 1974).
16. F. R. Magee and J. G. Proakis, "Adaptive Maximum-Likelihood Sequence Estimation for Digital Signaling in the Presence of Intersymbol Interference," *IEEE Trans. Info. Theory* **IT-19**, No. 1, 120–124 (January 1973).
17. J. M. Cioffi and T. Kailath, "Fast, Recursive-Least-Squares, Transversal Filters for Adaptive Filtering," *IEEE Trans. Acoust., Speech, & Signal Proc.* **ASSP-34**, No. 2, 304–337 (April 1984).
18. J. Rissanen, "Modeling by Shortest Data Description," *Automatica* **14**, 465–471 (1978).
19. J. M. Cioffi and T. Kailath, "Windowed FTF Adaptive Algorithms with Normalization," *IEEE Trans. Acoust., Speech, & Signal Proc.* **ASSP-33**, No. 3, 607–625 (June 1985).
20. O. Agazzi, D. G. Messerschmitt, and D. A Hodges, "Nonlinear Echo Cancellation of Data Signals," *IEEE Trans. Commun.* **COM-30**, No. 11, 2421–2433 (November 1982).
21. A. Gersho and E. Biglieri, "Adaptive Cancellation of Channel Nonlinearities for Data Transmission," *Proceedings of ICC'84*, Amsterdam, May 1984, pp. 1239–1242.

Received September 28, 1985; accepted for publication December 10, 1985

John M. Cioffi *Stanford University, Department of Electrical Engineering, Stanford, California 94305.* Dr. Cioffi received the B.S. degree in electrical engineering from the University of Illinois, Urbana, in 1978 and the M.S. and Ph.D. degrees in electrical engineering from Stanford University in 1979 and 1984, respectively. From 1979 to 1981, he worked in the Advanced Data Communications Department of Bell Laboratories in Holmdel, New Jersey, and from 1984 to 1986 in the signal processing and coding theory group of the IBM Research Division in San Jose, California. He is currently an assistant professor of electrical engineering at Stanford University. Dr. Cioffi is a member of Eta Kappa Nu, Phi Eta Sigma, Phi Kappa Phi, Sigma Xi, Tau Beta Pi, and the Institute of Electrical and Electronics Engineers. He is the associate editor for adaptive filtering of the *IEEE Transactions on Acoustics, Speech, and Signal Processing.*

FOURIER TRANSFORM

$$x(t) \Leftrightarrow X(f)$$

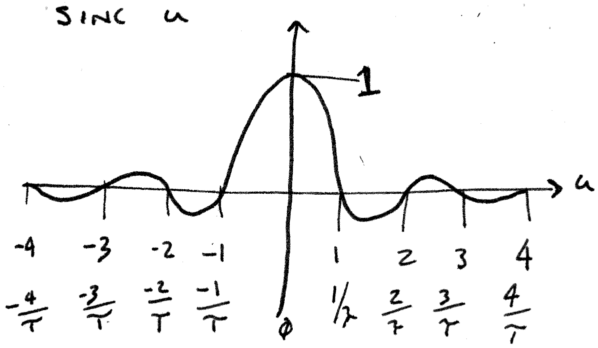
TIME DOMAIN — FREQUENCY DOMAIN

$$\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2j}$$

$$\cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

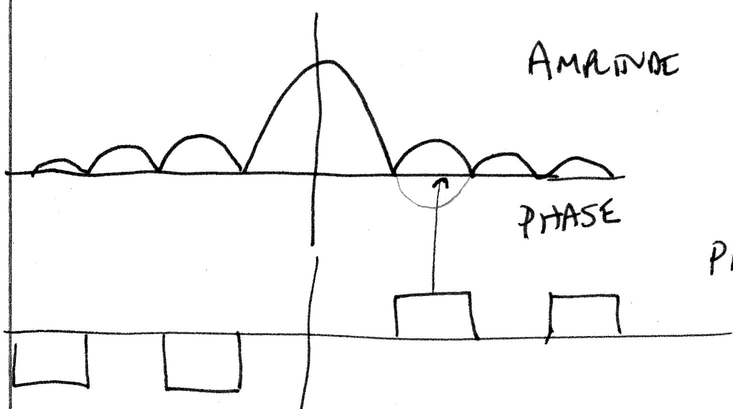
RECTANGULAR FUNCTION

SINC u

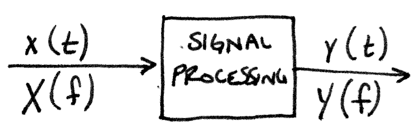


NARROW PULSE REQUIRES MORE HIGH FREQUENCIES

AMPLITUDE SPECTRUM HAS ABS VALUE



PHASE SPECTRUM CORRECTS NEGATIVE PORTIONS OF AMPLITUDE SPECTRUM



FOURIER TRANSFORM THEOREMS

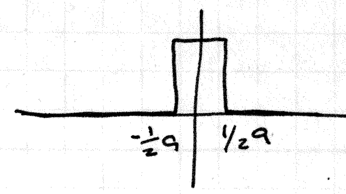
1. ADDITION $a_1x_1(t) + a_2x_2(t) = A_1X_1(f) + A_2X_2(f)$
2. PARSEVAL'S RELATIONSHIP (ENERGY)
3. DELAY
4. SCALE CHANGE
5. * DUALITY
6. INTEGRATION, DIFFERENTIATION

DELTA FUNCTIONS

1. AREA = 1
2. VERY NARROW

$$\int_{-\infty}^{\infty} \delta(t)\,dt = 1$$

$$\delta(t) = \lim_{a \to \infty} a \,\mathrm{RECT}\left(\frac{t}{a}\right)$$



a. CONVOLVING WITH δ GIVES ORIGINAL FUNCTION

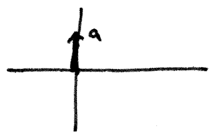
b. $\delta(-t) = \delta(t)$ SYMMETRICAL

$F(\text{CONSTANT}) = \delta(0)$

c. $F[\delta(t)] = 1$

$F(\delta(t-t_0)) = 1 \cdot e^{-2\pi f t_0}$

d. IF $x(t) = a\,\delta(t)$



e. SIFTING PROPERTY

$$\int_{-\infty}^{\infty} f(t)\,\delta(t-t_0)\,dt = f(t_0)$$

$$d[x(t), y(t)] = \sqrt{\int_{-\infty}^{\infty} [x(t)-y(t)]^2\,dt}$$

DISTANCE FUNCTION
$d(x,y) \geq 0$

$d(x,y) \leq d(x,z) + d(z,y)$ TRIANGULAR INEQUALITY

MEASURE OF SIMULARITY

RANDOM SIGNALS CANNOT BE DESCRIBED AS FUNCTIONS
DESCRIBED IN TERMS OF AVERAGE VALUE

LECTURE 3

TOPICS:

I. MAGNETIC RECORDING CHANNEL

II. CHANNEL IDENTIFICATION METHODS

III. PEAK DETECTION METHOD

I. MAGNETIC RECORDING CHANNEL:

— WRITE CURRENT: NRZI (non-return-to-zero inverted)

data: $\sum_n a_n \delta(t-nT)$

$$a_n \in \{-1, 1\}$$

e.g. $a_n = -(-1)^{d_n}$ where $d_n = 0$ or 1

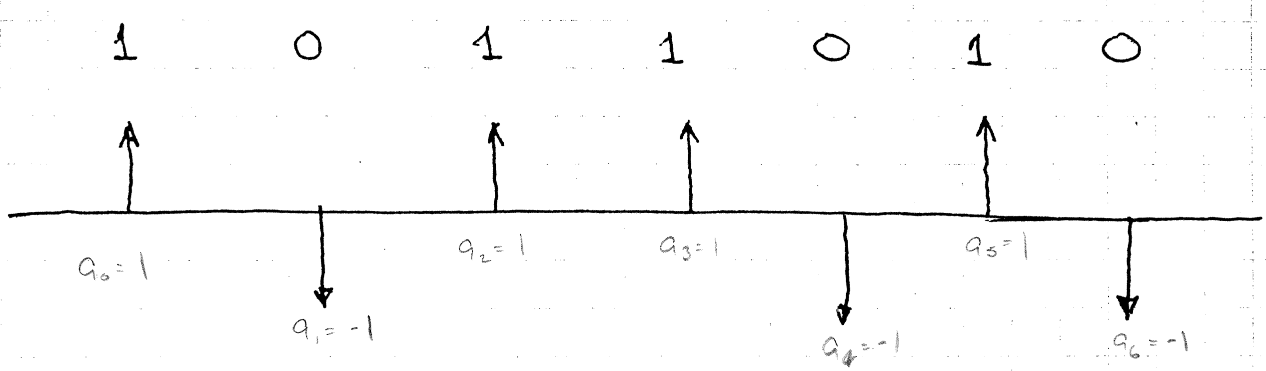
NEGATIVE CURRENT — DATA ZERO

$$\Rightarrow \quad a_n = -1 \quad \text{if } d_n = 0$$

POSITIVE CURRENT — DATA ONE

$$a_n = 1 \quad \text{if } d_n = 1$$

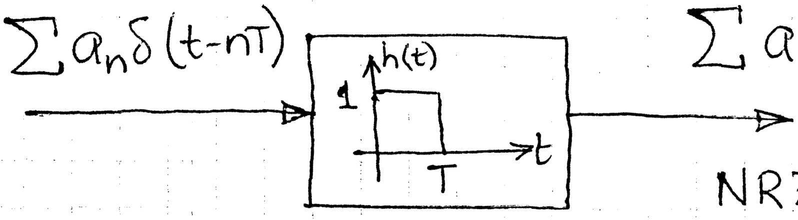
CHANNEL SYMBOLS



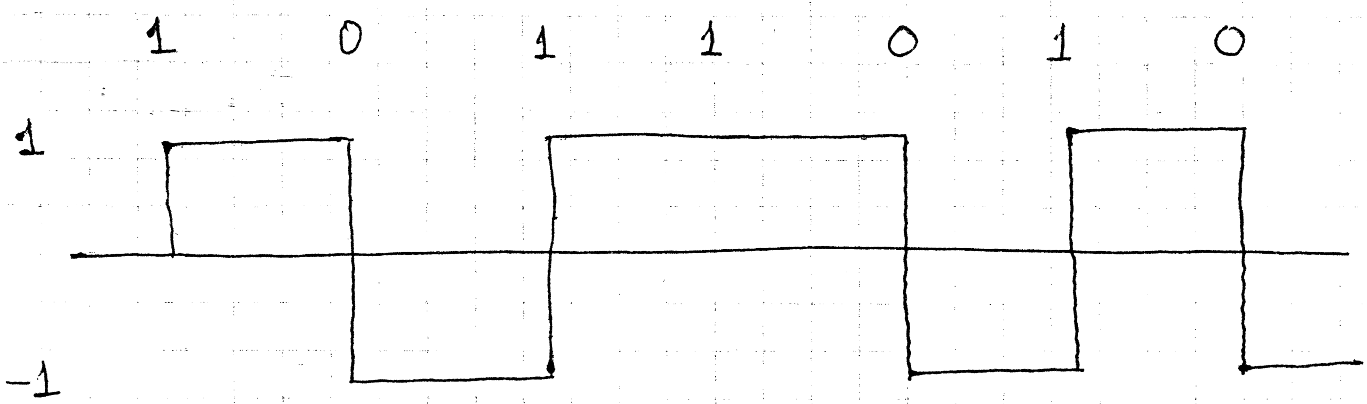
DATA SIGNAL

LINEAR RESPONSE OF $h(t-nT)$ FROM $\delta(t-nT)$

$\sum a_n \delta(t-nT)$ → $\sum a_n h(t-nT)$ CONVOLUTION

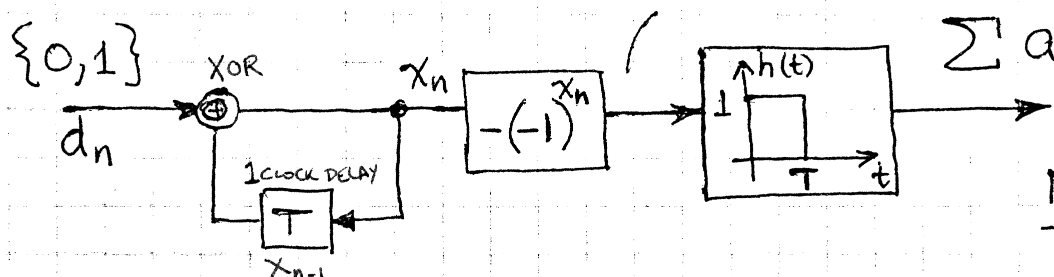


NRZ (non-return-to-zero)



$\sum a_n' \delta(t-nT)$

$\sum a_n' h(t-nT)$



NRZI

$$x_n = d_n \oplus x_{n-1}$$

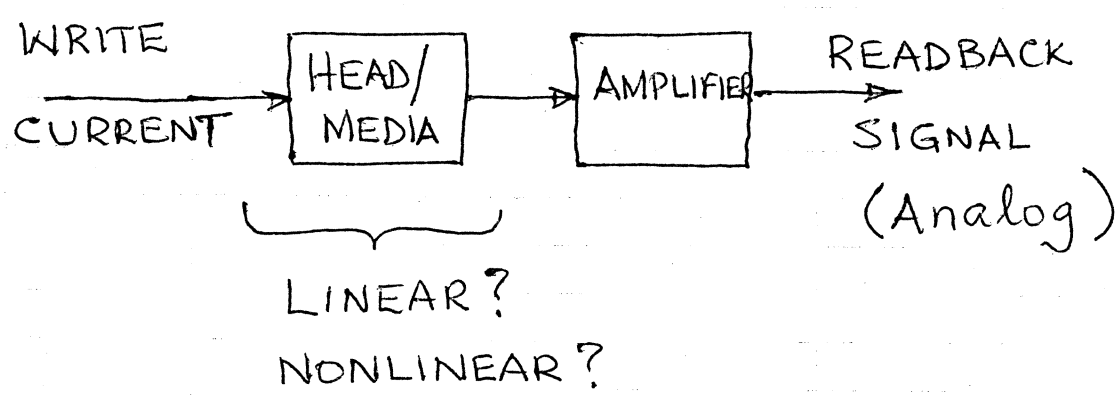
d_n:		1	0	1	1	0	1	0
x_n:	0	1	1	0	1	1	0	0
a_n':		1	1	-1	1	1	-1	-1

WRITE CURRENT POLARITY CHANGES ON EVERY 1 INPUT DATA

NRZI IS A FORM OF DIFFERENTIAL ENCODING.

IT PROVIDES 180° PHASE INVARIANCE.

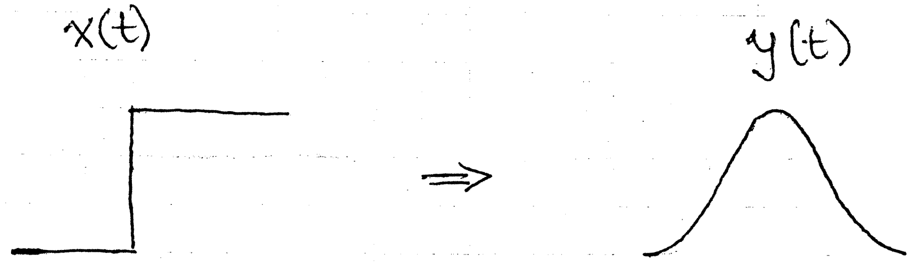
NRZI = 180° NRZI = -NRZI



- LINEARITY

$$x_1(t) \rightarrow y_1(t) \qquad x_2(t) \rightarrow y_2(t)$$

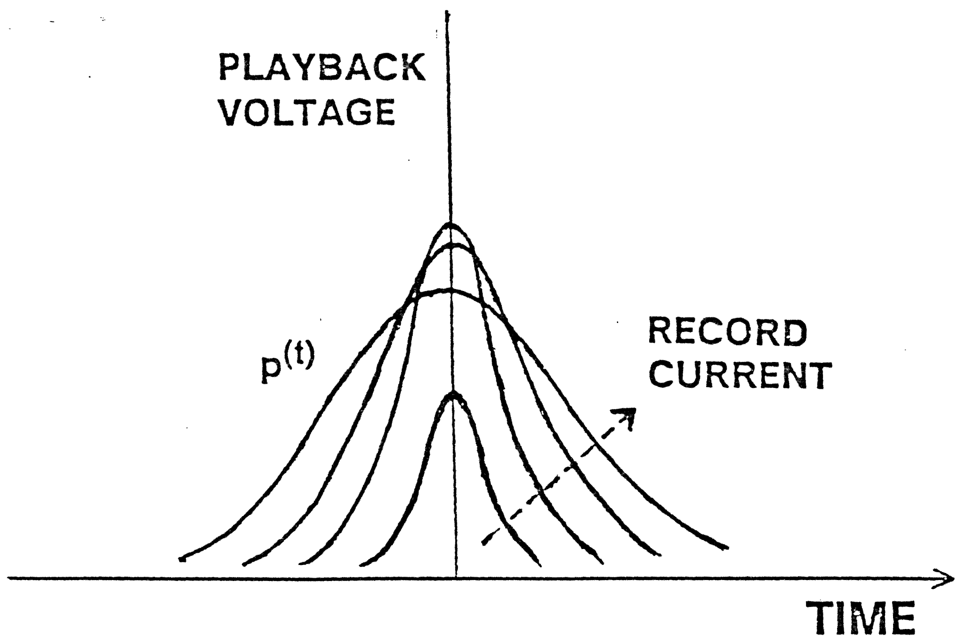
$$a_1 x_1(t) + a_2 x_2(t) \rightarrow a_1 y_1(t) + a_2 y_2(t)$$

- 

$a\,x(t)$, in general, does not give $a\,y(t)$. However, if we apply $x(t) - x(t-T)$, then the response is $y(t) - y(t-T)$ over a fairly wide range of recording density. Thus, the recording channel is linear in a narrow sense.

DIGITAL MAGNETIC RECORDING CHANNEL

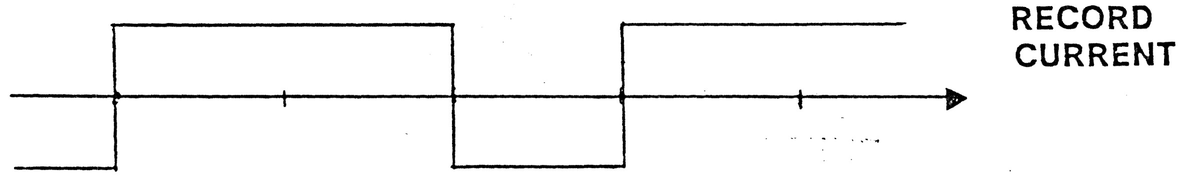
BINARY SATURATION RECORDING

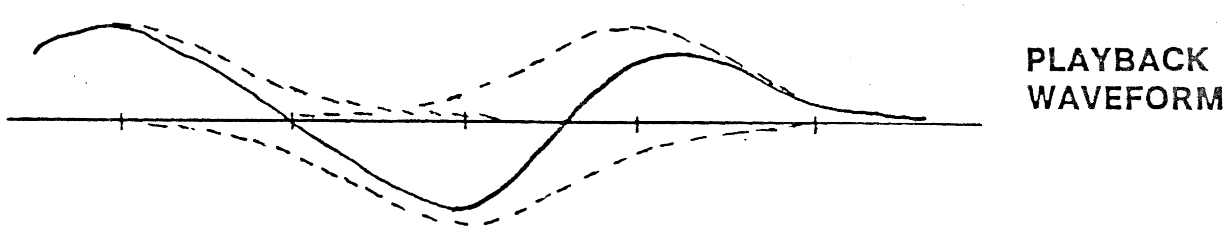


THE TRANSITION RESPONSE AS A FUNCTION OF RECORD CURRENT

SUPERPOSITION

DESPITE THE INTENSELY NONLINEAR BEHAVIOR OF THE MAGNETIC RECORDING PROCESS, THE RESPONSE TO BINARY WAVEFORMS CAN BE BUILT UP BY SUPERPOSITION OF THE TRANSITION RESPONSES.

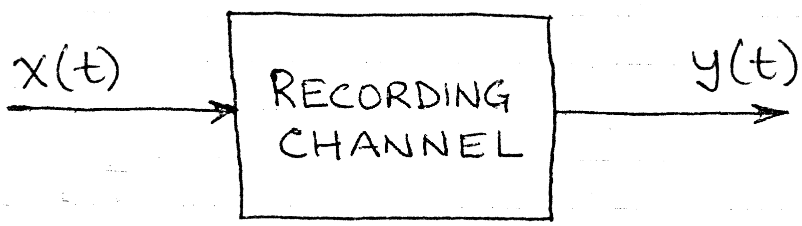


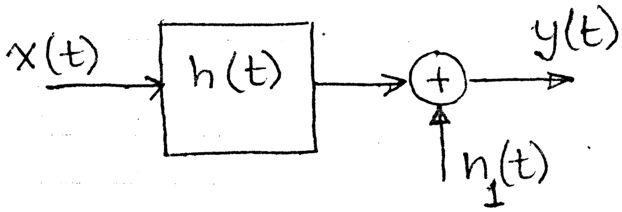


SUPERPOSITION IS REMARKABLY SUCCESSFUL EVEN UP TO VERY HIGH DENSITIES.

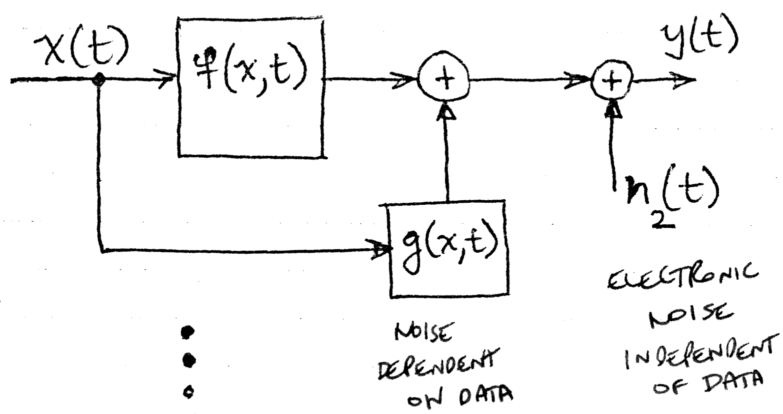
LECTURE # 3

CHANNEL IDENTIFICATION





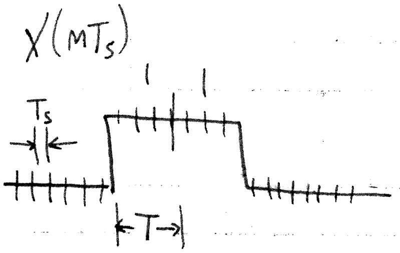
LINEAR, ADDITIVE NOISE MODEL



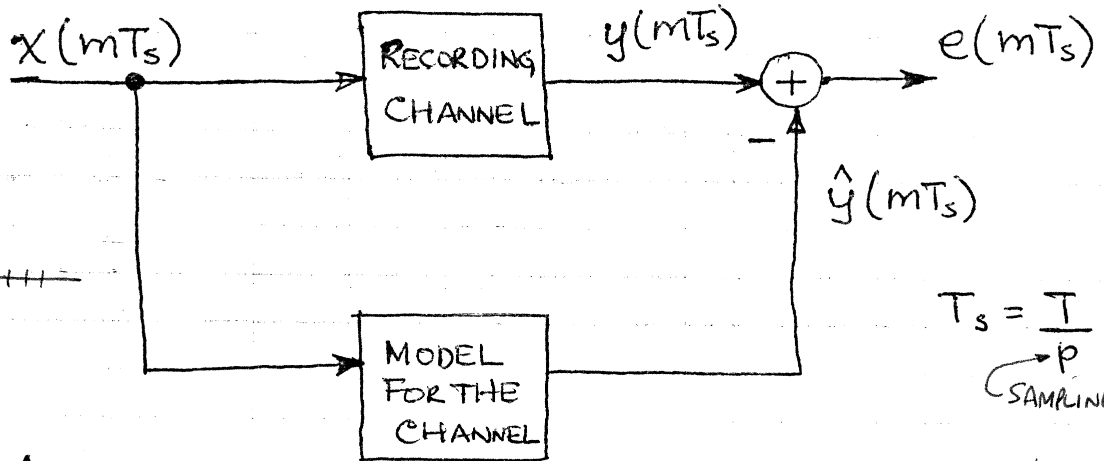
NONLINEAR, DATA-DEPENDENT NOISE MODEL

1) EVALUATE PERFORMANCE OF DIFFERENT DETECTION METHODS
2) CHOOSE DETECTION METHOD
3) DETECTOR DESIGN (FILTERING, COMPENSATE CHANNEL DISTORTION, etc.)
4) VARIATION IN RECORDING CHANNEL RESPONSE
5) BOUNDS ON PERFORMANCE OF THE RECORDING CHANNEL

- TIME-DOMAIN IDENTIFICATION



m = SAMPLE NUMBER



$T_s = \frac{T}{p}$ $(p \geq 1)$

p: SAMPLING RATE

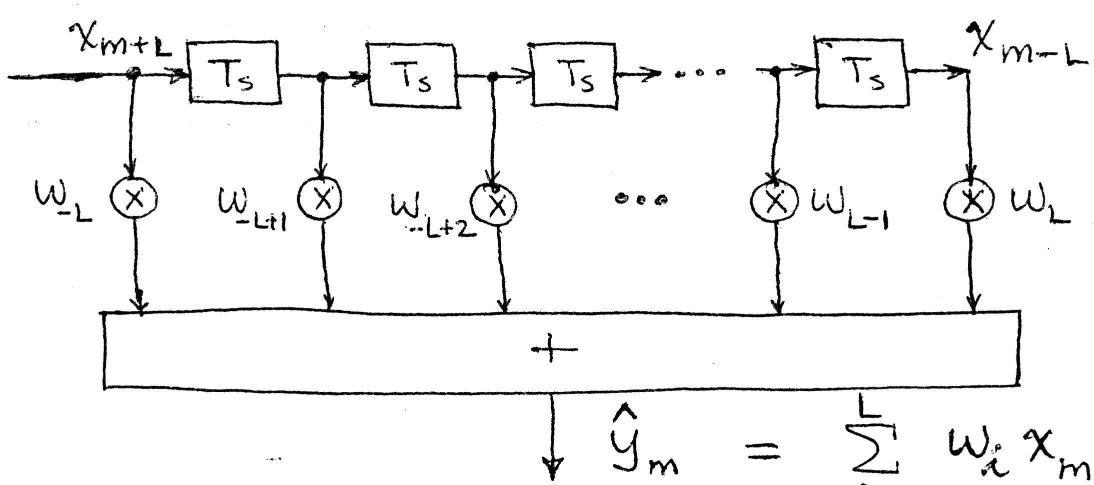
$x_m \triangleq x(mT_s)$ $\qquad e_m \triangleq e(mT_s) = y(mT_s) - \hat{y}(mT_s)$

$y_m \triangleq y(mT_s)$ $\qquad \hat{y}_m \triangleq \hat{y}(mT_s)$

y_m: SAMPLE VALUE; m: SAMPLE # 0, 1, 2, 3; T_s: SAMPLE PERIOD

ADJUST PARAMETERS OF THE CHANNEL MODEL SO THAT $E[e_m^2]$ (ERROR) IS MINIMIZED.

- LINEAR MODEL FOR THE CHANNEL: FINITE IMPULSE RESPONSE (FIR) MODEL



$$\hat{y}_m = \sum_{i=-L}^{L} w_i x_{m-i}$$

LINEAR COMBINATION

DISCRETE CONVOLUTION

$1 * 2L+1$

Row vector

$$\underline{x}_m \triangleq [\,x_{m+L}\ \ x_{m+L-1}\ \ x_{m+L-2} \cdots x_m\ \ x_{m-1}\ \ x_{m-2} \cdots x_{m-L}\,]$$

Col vector

$$\underline{w} \triangleq [\,w_{-L}\ \ w_{-L+1}\ \ w_{-L+2} \cdots w_0\ \ w_1\ \ w_2 \cdots w_L\,]^T$$

$2L+1 * 1$

$(2L+1)$ unknowns

TRANSITION RESPONSE SPANS 7-10 BIT PERIODS

$$\hat{y}_m = \underline{x}_m \underline{w}^{\cancel{*}}$$

ESTIMATED OUTPUT

- Let $\hat{y}_m = y_m$; that is,

$$\underline{x}_m \underline{w}^T = y_m \qquad m = 0, 1, 2, \ldots K$$

or, equivalently, where $K \geq (2L+1)$

$$\begin{bmatrix} \underline{x}_0 \longrightarrow \\ \underline{x}_1 \longrightarrow \\ \underline{x}_2 \longrightarrow \\ \vdots \\ \underline{x}_K \longrightarrow \end{bmatrix} \begin{bmatrix} w_{-L} \\ w_{-L+1} \\ w_{-L+2} \\ \vdots \\ w_0 \\ \vdots \\ w_L \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ \\ y_K \end{bmatrix}$$

OVER-DETERMINED SYSTEM OF LINEAR EQUATIONS

$$\underset{\sim}{X}\, \underline{w}' = \underline{y}$$

THE LEAST-SQUARES SOLUTION IS GIVEN BY

$$\underline{W} = \underset{\sim}{R}^{-1} \underline{P}$$

$$\text{WHERE} \quad \underset{\sim}{R} = \left[\underset{\sim}{X}^T \underset{\sim}{X} \right]_{(2L+1)\times(2L+1)}$$

$$\underline{P} = \left[\underset{\sim}{X}^T \underline{y} \right]_{(2L+1)\times 1}$$

$\underset{\sim}{X}$ and $\underline{y}$ are formed from known input and its digitized (possibly resampled) response.

- USE SUB-CHANNEL FORMULATION TO REDUCE THE DIMENSIONALITY OF $\underset{\sim}{R}$ AND $\underline{P}$ BY A FACTOR OF p WHERE $p = \frac{T}{T_s} \geq 1.$ NOT REDUCED! $\underline{P}$ IS NOW $\frac{(2L+1)}{P} \times P$
  - IN THE ABOVE FORMULATION, EACH ROW OF THE $\underset{\sim}{X}$ MATRIX IS REPEATED p TIMES; ONLY THE RIGHT-HAND SIDE (i.e., THE $\underline{y}$ VECTOR) CHANGES.

$$\underset{500}{}\qquad\underset{500}{}$$

- INSTEAD OF $\underset{\sim}{X}\,\underline{w} = \underline{y}$, WE CAN FORMULATE

$$\underbrace{\underset{\sim}{X}_R}_{50}\,\underbrace{\underline{W}}_{(10\times10)} = \underset{\sim}{Y}$$

$\underline{w} = (R_R)^{-1} P_R$

WHERE $\underset{\sim}{X}_R$ IS THE REDUCED VERSION OF $\underset{\sim}{X}$ AND $\underset{\sim}{Y}$ IS ALSO A MATRIX WITH p COLUMNS AND $\frac{K}{p}$ ROWS (ASSUME $\frac{K}{p}$ IS AN INTEGER).

- THE LEAST-SQUARES SOLUTION IS GIVEN BY

$$\underset{\sim}{W} = \begin{array}{c} \xleftrightarrow{\quad p \quad} \\ \begin{bmatrix} W_{-L} & W_{-L+1} & \cdots & W_{-L+p} \\ W_{-L+p+1} & \cdots & & \\ \cdots & & & \\ W_{L-p} & & & W_L \end{bmatrix} \end{array} = \underset{\sim}{R}_R^{-1}\,\underset{\sim}{P}_R$$

WHERE

$$\underset{\sim}{R}_R = \overset{50\times10}{(\underset{\sim}{X}_R^T\,\underset{\sim}{X}_R)}$$

$$\underset{\sim}{P}_R = (\underset{\sim}{X}_R^T\,\underset{\sim}{Y})$$

$X = 50$ BITS INPUT
$Y = 50$ BITS RESPONSE
$T_S = \frac{T}{10}$ $\quad P = 10$

500 INPUT SAMPLES
500 OUTPUT SAMPLES

$W = 10^T = 100$ UNKNOWNS

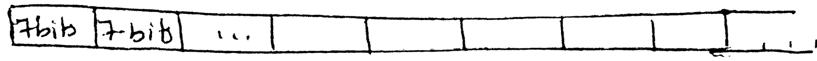
$$X_0 W = \overset{\xleftrightarrow{\text{P COLS}}}{[Y_0\ Y_1\ Y_2\ \ldots]}$$

$$T_S = \frac{T}{P}$$

$$\underline{X_0}\,\underline{W} = \overset{\xleftrightarrow{\text{P COLS}}}{[Y_0, Y_1, Y_2 \ldots Y_{P-1}]}$$

Homework #2 EECS458

1. A periodic input sequence is generated by repeating the 7-bit PRBS ~~as~~ 1 0 0 1 1 1 0 as



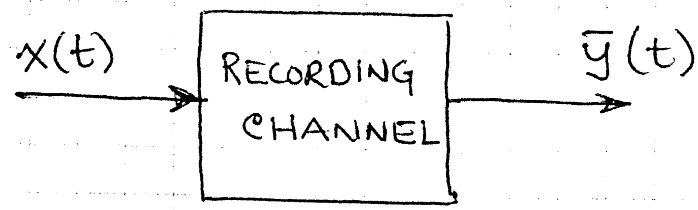
~~Where~~ This sequence is recorded and the readback signal is digitized at the rate of 2 samples per bit (2samples/T). The output is given by $y_0\ y_1\ y_2 \ldots y_{13}\ y_0\ y_1 \ldots y_{13}\ y_0\ y_1 \ldots$. It is known that the channel response spans 3 bit intervals. (6 SAMPLES)

a) Formulate the $\underset{\sim}{R}$ matrix and the $\underset{\sim}{P}$ vector to model the channel in terms of an FIR filter with 6 coefficients (each coefficient spaced T/2 seconds apart).

b) Repeat (a) using the sub-channel response.

(In both (a) and (b), set-up the equations to solve for the ~~step~~ pulse (dibit) response).

○ FREQUENCY-DOMAIN APPROACH



FOURIER TRANSFORM OF AVG OUTPUT

$$\bar{Y}(f) = H(f)\,X(f)$$

COMPLEX CONJUGATE

or $$\bar{Y}(f)\,X^*(f) = H(f)\,X(f)\,X^*(f) = H(f)\,|X(f)|^2$$

or $$\boxed{H(f) = \frac{\bar{Y}(f)\,X^*(f)}{|X(f)|^2}} \Rightarrow h(t) = \mathcal{F}^{-1}[H(f)]$$

AVOIDS DIV BY COMPLEX NUMBERS

$$\bar{Y}(f) = \mathcal{F}[\bar{y}(t)]$$

$$X(f) = \mathcal{F}[x(t)]$$

– $\dfrac{\bar{Y}(f)\,X^*(f)}{|X(f)|^2}$ defines the cross-spectrum of output and input. Its inverse Fourier transform is the cross-correlation function of the output and the input. If $|X(f)|^2 = 1$, then

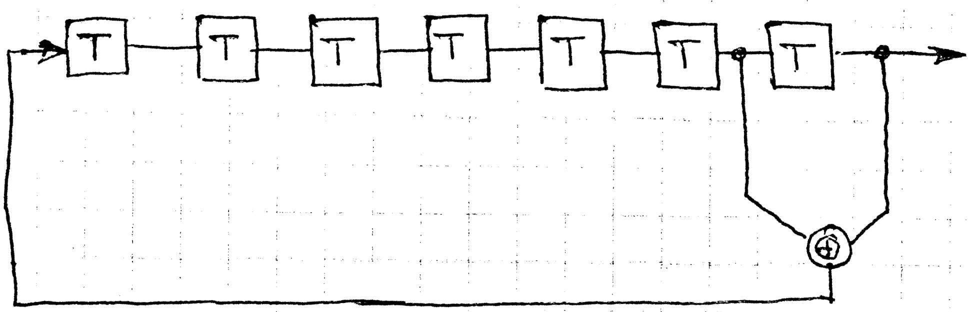
$$\boxed{h(t) = R_{yx}(t) = \int_{-\infty}^{\infty} y(\lambda)\,x(t+\lambda)\,d\lambda}$$

– CHOICE OF INPUT SEQUENCE :

SPECTRUM IS WHITE

Pseudo-random binary sequences (PRBS) of maximal length are preferred. They have the property that the fundamental and harmonics are all of the same amplitude (except for the roll-off due to $\frac{\sin(\omega T/2)}{(\omega T/2)}$ inherent in NRZ signalling)

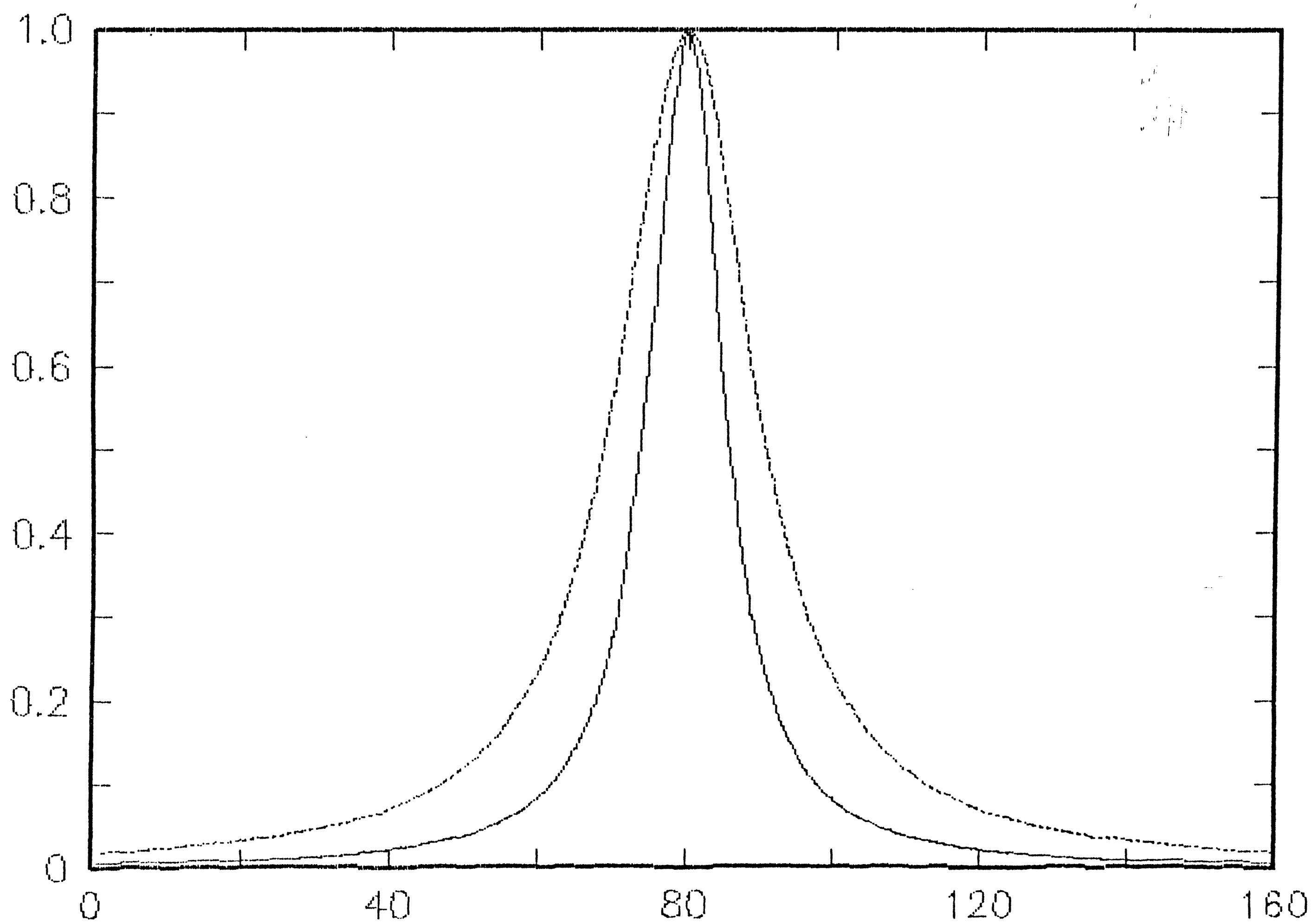
e.g.

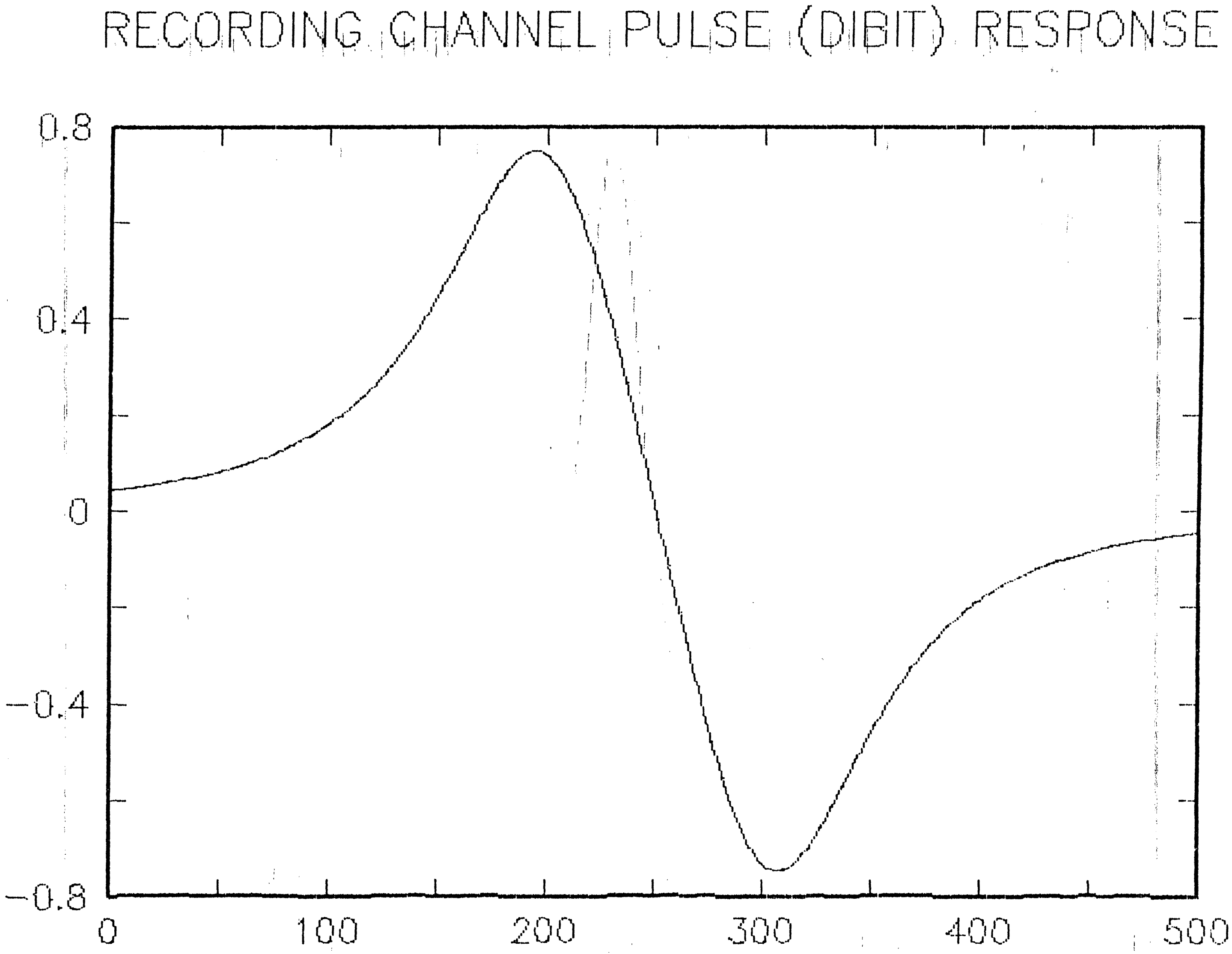


127 bit PRBS generator

$2^{n}-1$ PERIOD

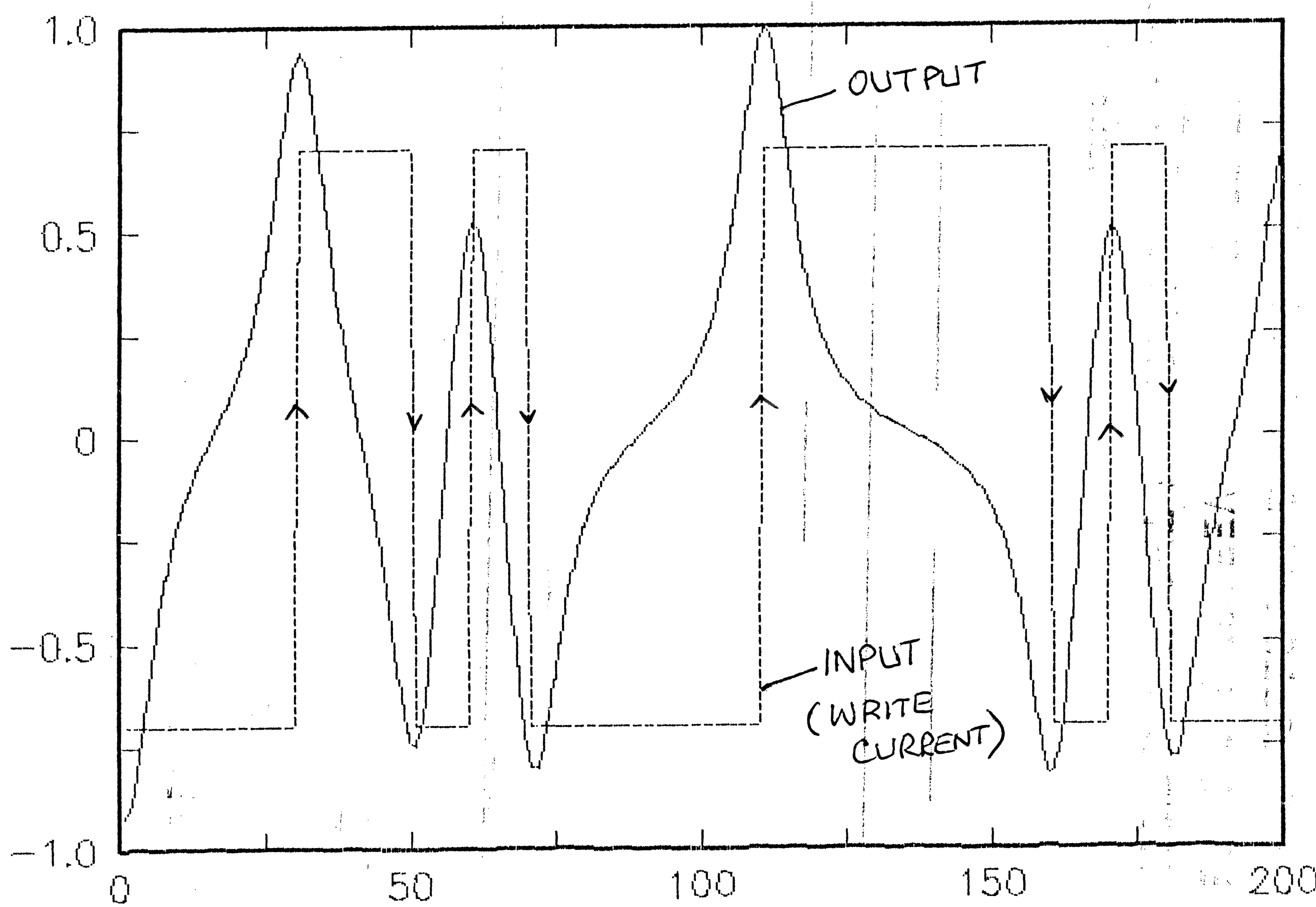
RECORDING CHANNEL STEP (TRANSITION) RESPONSE




RECORDING CHANNEL PULSE (DIBIT) RESPONSE
0.8
0.4
0
−0.4
−0.8
0
100
200
300
400
500

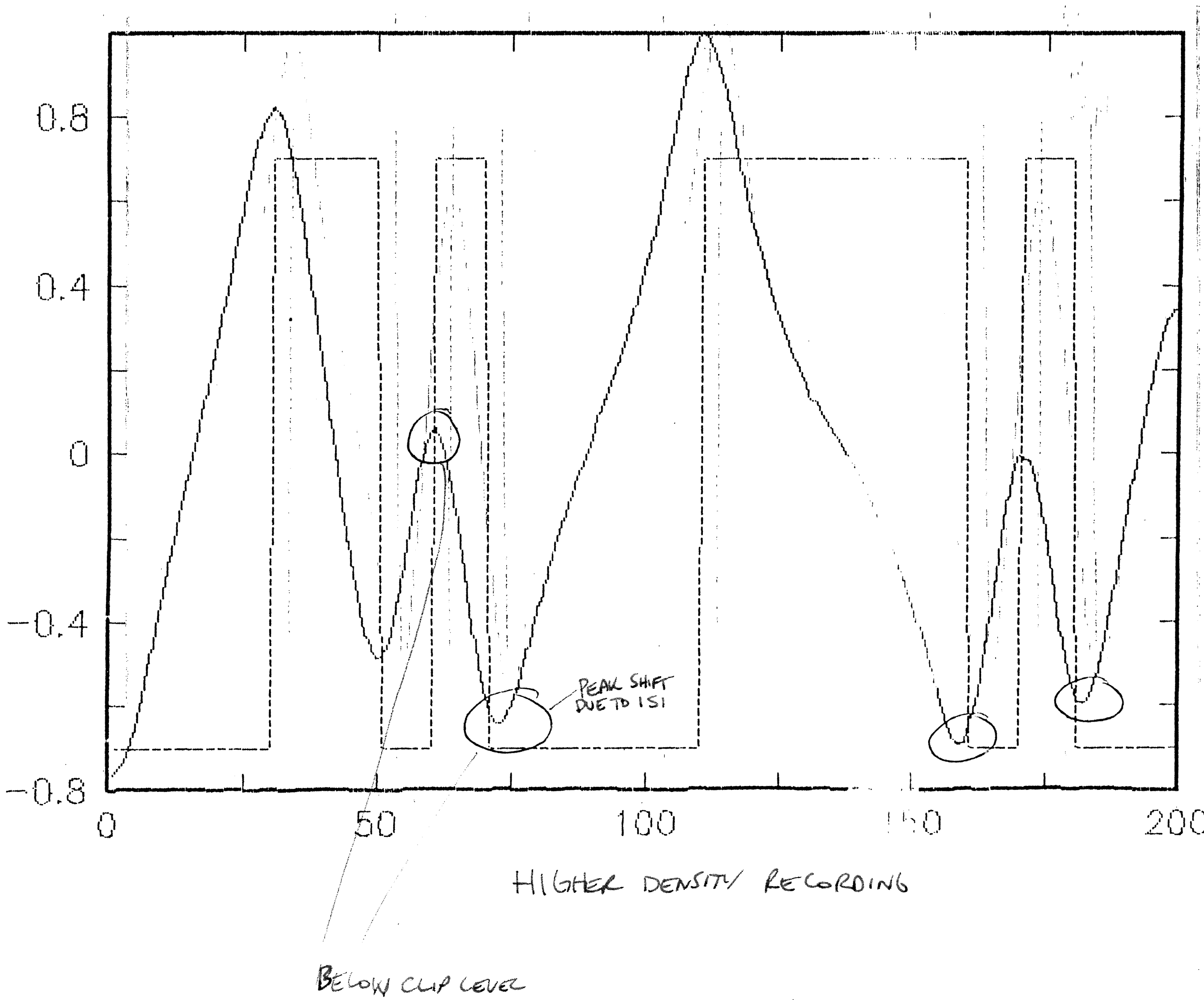
IBM - SJ - 221

INPUT/OUTPUT OF A MAGNETIC RECORDING CHANNEL



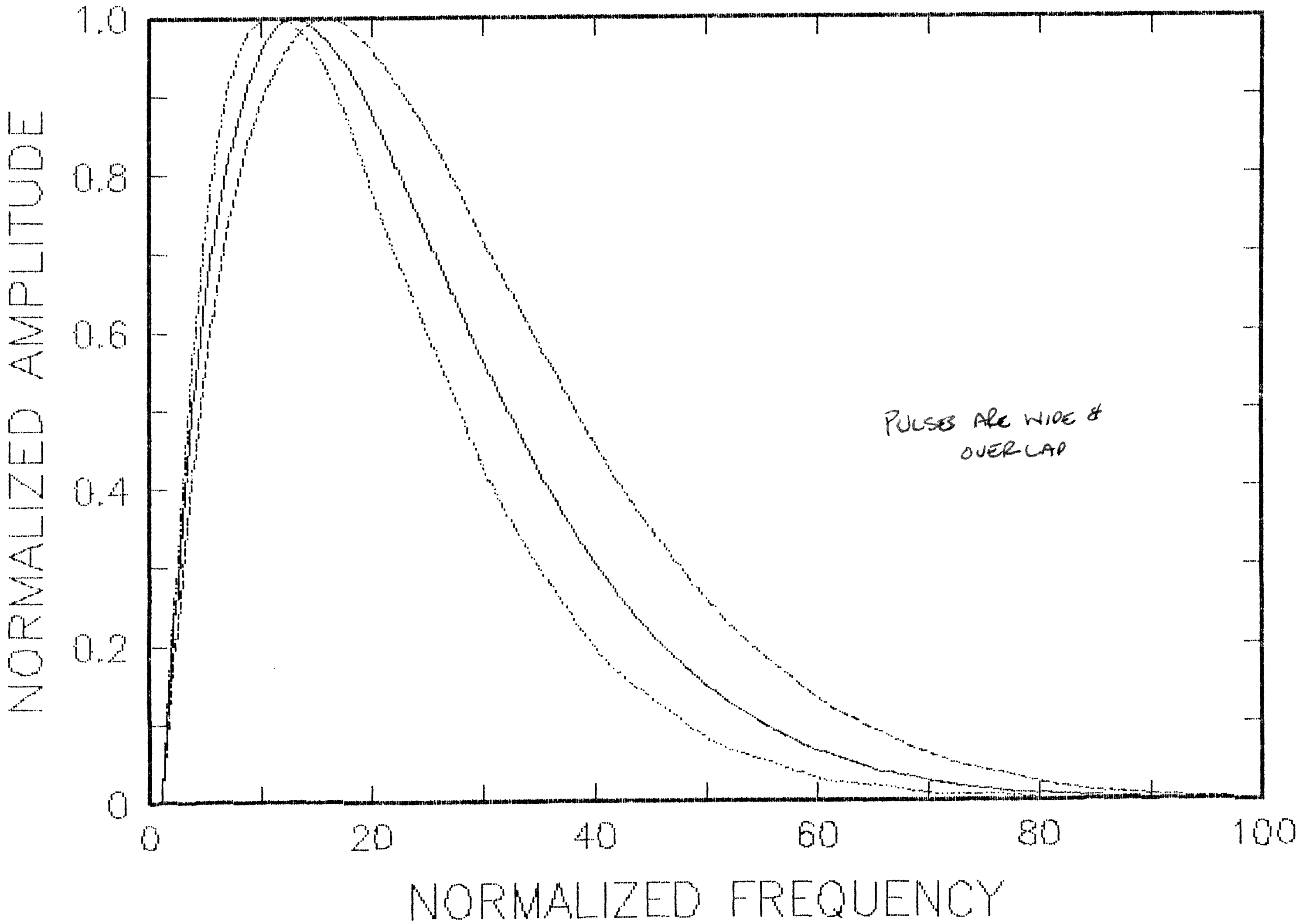
IBM - SJ - 221

INPUT/OUTPUT OF A MAGNETIC RECORDING CHANNEL



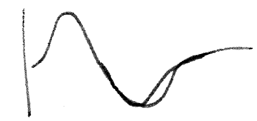
IBM SJ 221

AMPLITUDE SPECTRUM OF THE PULSE RESPONSE



BAND LIMITED CHANNEL

FOURIER TRANSFORM OF



IBM SJ 221

4.1
4/20/89

LECTURE 4:

RECORDING CHANNEL (REVIEW)

a) Readback waveform is generally well approximated by the sum of appropriate sequence of isolated transition pulses; i.e.,

$$y(t) = \sum_n b_n s(t-nT)$$

where $s(t)$ is the isolated transition response and $b_n = a_n' - a_{n-1}'$. a_n' represents the input data symbols.

$a_n' = \begin{smallmatrix}1\\0\end{smallmatrix}$

$b_n = \begin{smallmatrix}+2\\0\\-2\end{smallmatrix}$

b) b_n can take on 3 possible values: 0 and ± 2 if $a_n' \in \{1,-1\}$. Moreover, the non-zero values of b_n always alternate in sign. At low to moderate densities, the readback signal has pulses that also alternate in polarity.

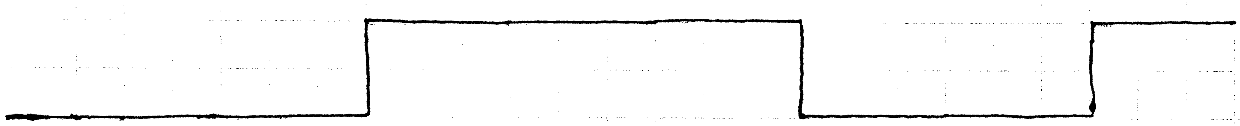
EXAMPLE:

data d_n: 0 1 0 0 1 0 1

a_n' (NRZI): -1 -1 1 1 1 -1 -1 1

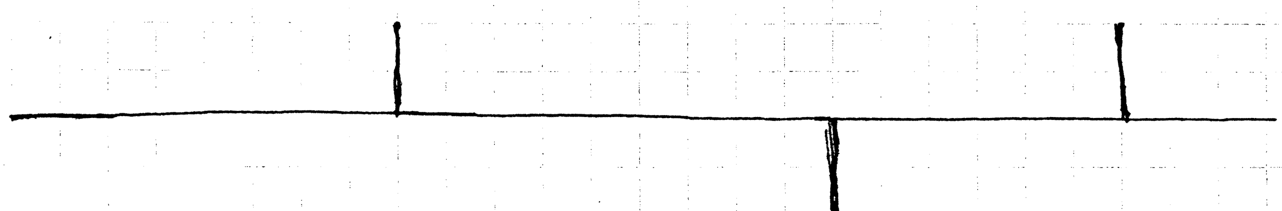
NO CHANGE CHANGE NO CHANGE CHANGE

write current:

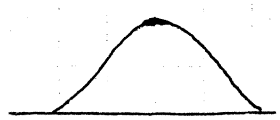


b_n : 0 0 2 0 0 -2 0 2

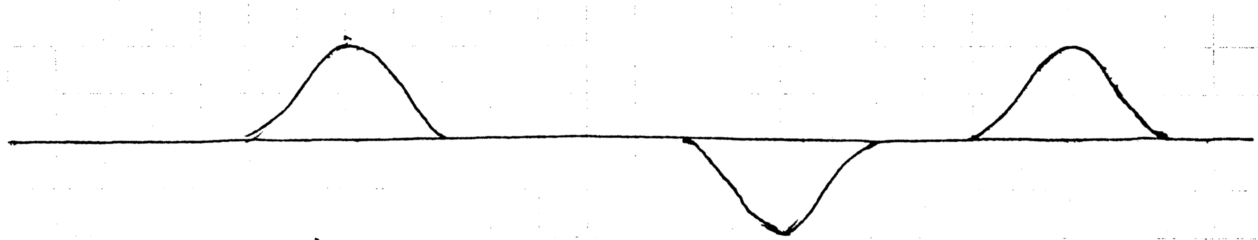
$= a_n' - a_{n-1}'$



$s(t)$:



$y(t)$:



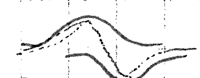
$$\sum_n b_n s(t-nT)$$

c) The recording channel is :

i) Peak amplitude limited (the maximum amplitude is associated with an isolated transition), No I.S.I.

ISI YIELDS CANCELLATION

ii) Bandwidth limited



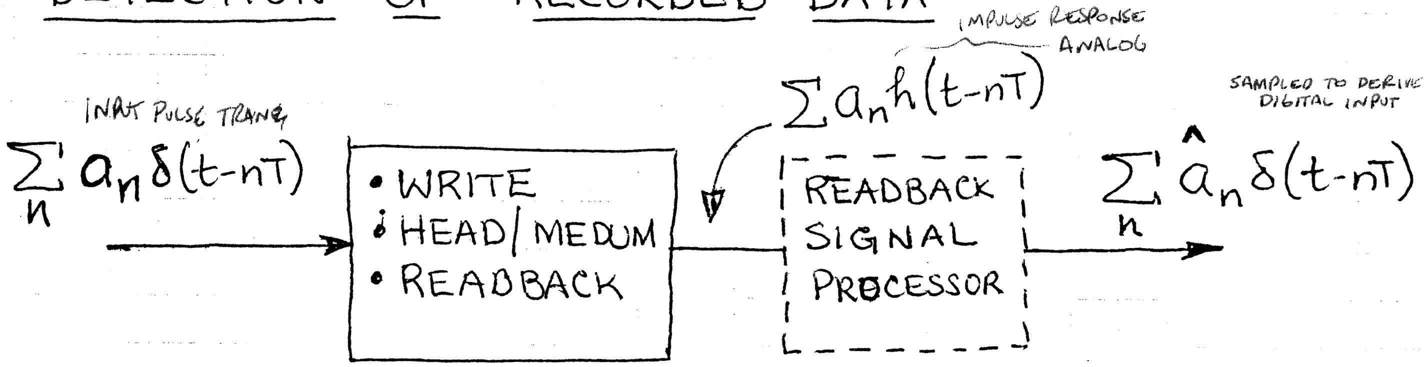
POWER DECREASES

d) When the duration of the transition response exceeds the minimum inter-transition spacing, intersymbol interference (isi) occurs.

e) The presence of isi reduces the peak amplitude. The average readback signal power reduces with increasing isi.

f) The overall channel response is determined by the geometric and magnetic properties of the head/medium interface.

DETECTION OF RECORDED DATA



want $\quad \Pr[\hat{a}_n \neq a_n] < \epsilon$ $\quad$ REDUCE ERROR RATE

READBACK SIGNAL PROCESSOR

- Analog methods (Peak detection)
- Digital methods (sampling detection methods: Partial response, decision-feedback equalization, maximum likelihood sequence detection (MLSD), ...)

PEAK DETECTION METHOD:

SLOPE IS ZERO $\frac{dx}{dt}$
ZERO CROSSING DETECTOR
COMPARE TO $X(t)$ CLIPPING LEVEL
TO ELIMINATE SPURIOUS TRANSITIONS

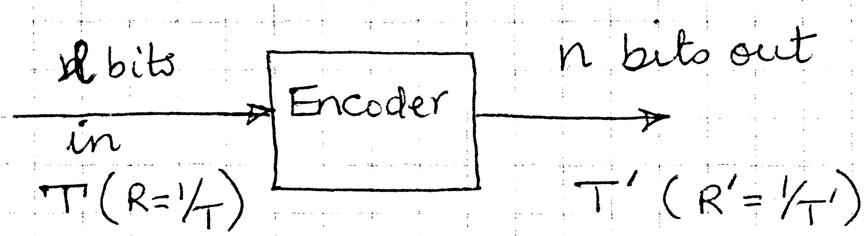
- Used in conjunction with runlength limited (RLL) codes.
  INCREASE TRANSITION SPACING.

- Commonly used RLL codes in digital storage systems are characterized by two parameters:

  d = minimum number of 0s between consecutive 1s

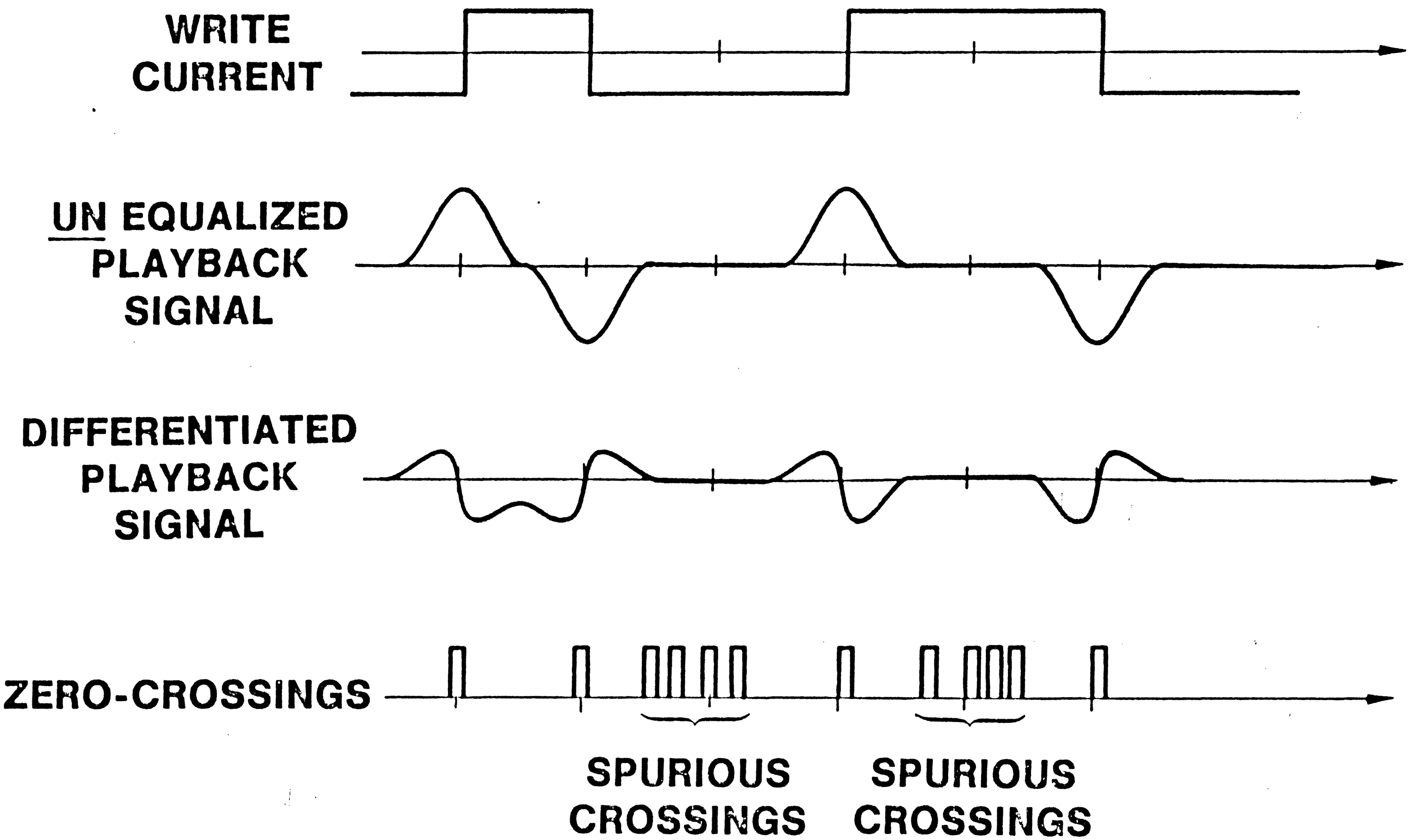
  k = maximum number of 0s between consecutive 1s.

  (d, k) codes.

- 

Rate of a code $\triangleq \dfrac{\ell}{n}$

$$T' = \frac{\ell}{n} T \Rightarrow R' = \frac{n}{\ell} R$$



PEAK DETECTION (DISCS)

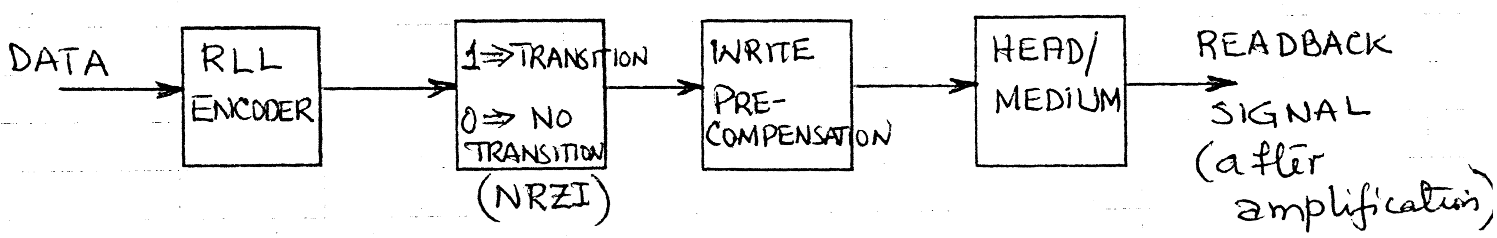
IBM · SJ · 221

- Commonly used codes:

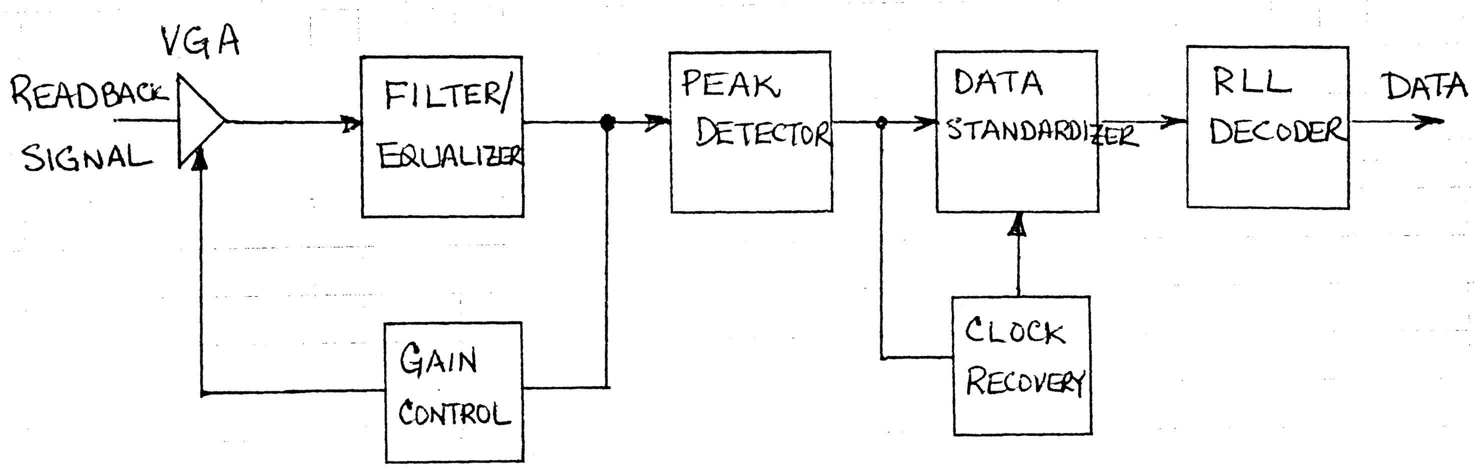
	d	k	rate	min. transition spacing	
MFM	1	3	1/2	T	
(2,7)	2	7	1/2	3T/2	REDUCED ISI
(1,7)	1	7	2/3	4T/3	MORE INTERTRANSITION SPACING + HIGHER CLOCK

2 DATA → 3 ENCODED

RECORDING:



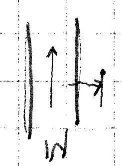
PEAK DETECTION:



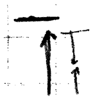
- PERFORMANCE ANALYSIS OF THE PEAK DETECTION METHOD:

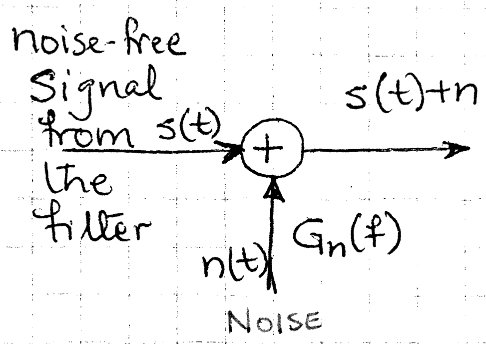
TWO WAYS IN WHICH ERRORS CAN OCCUR

– Peak-shift error (peak falls outside the correct detection window) ①



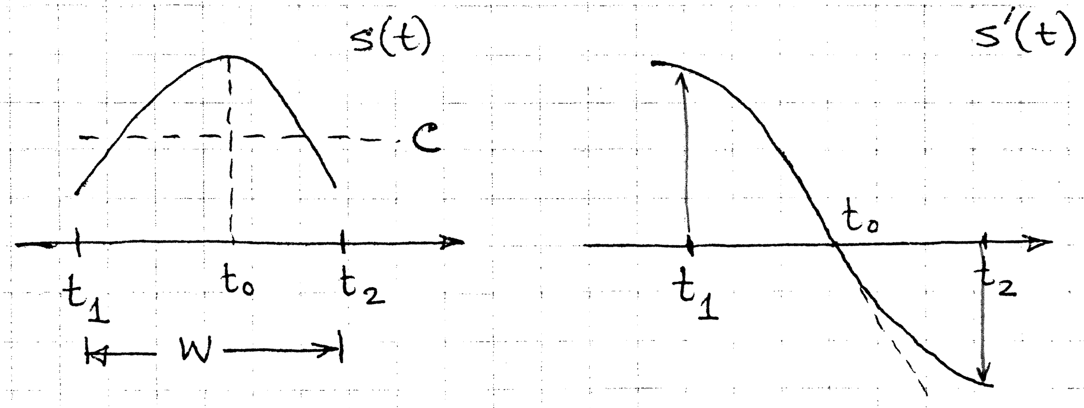
– Missing bit error (pulse amplitude falls below the clipping level) ②



- 

(linear model)

$\frac{1}{Hz^2}$

① 

$$\Pr\left[\, s'(t)+n'(t) < 0 \ \text{ or } \ s'(t)+n'(t) > 0 \,\right] \ (t \text{ in } W)$$

$$\Pr(A \text{ or } B) = P(A) + P(B) - P(AB)$$
$$\leq P(A) + P(B)$$

$$\Pr\left[s'(t) + n'(t) < 0 \right] \leq \Pr\left[n'(t_1) \leq -s'(t_1) \right]$$

t_1 is the point where $s'(t)$ is positive and has the largest value.

Similarly,

$$\Pr\left[s'(t) + n'(t) > 0 \text{, for } t \text{ in } W \right] \leq \Pr\left[n'(t_2) > -s'(t_2) \right]$$

t_2 is the point where $s'(t)$ is negative and has the largest absolute magnitude.

2 $\Pr\left[|s(t) + n(t)| < C \right]$ CLIPPING LEVEL for t in W

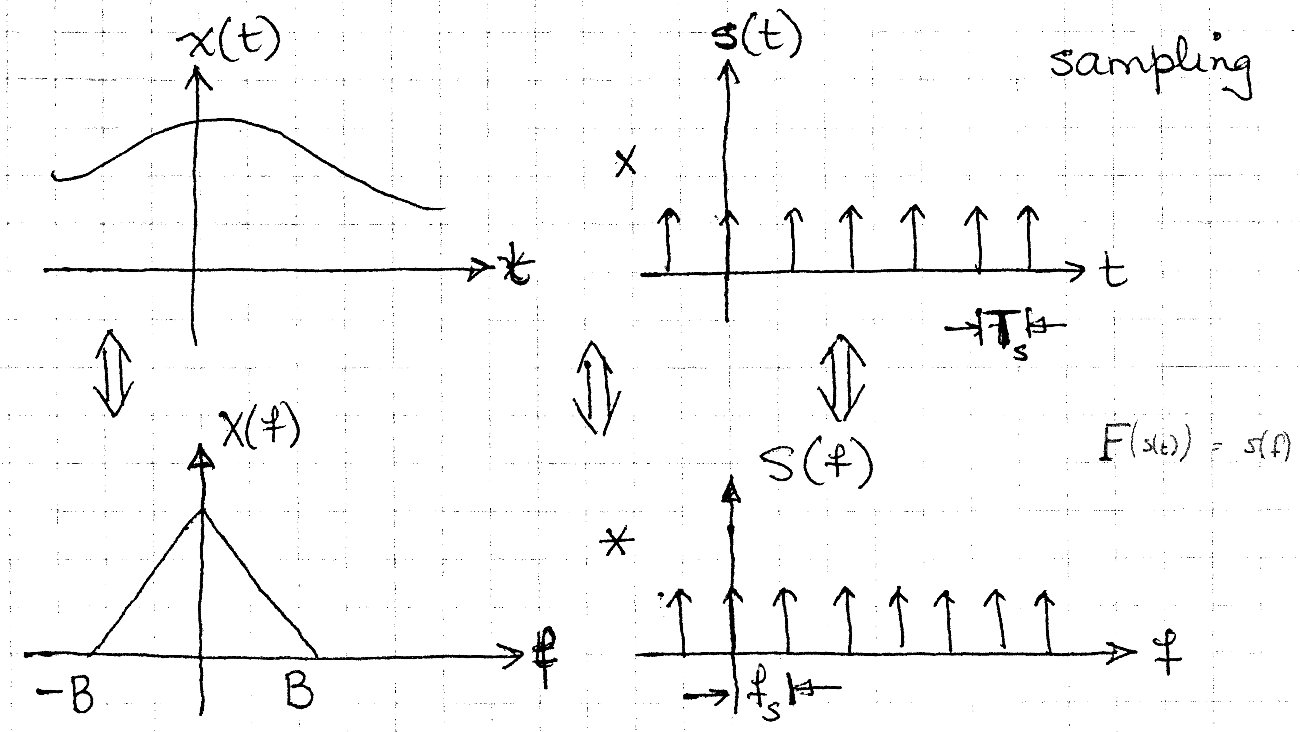
$$\approx \Pr\left[|s(t_0) + n(t_0)| < C \right]$$

where t_0 represents the point where $s(t)$ is the largest.

If $n(t)$ is Gaussian, the $n'(t)$ is Gaussian. Knowing $s'(t_1)$, $s'(t_2)$, and $s(t_0)$ for input data patterns, one can determine the above probabilities as a function of the noise statistics.

SAMPLING DETECTION METHODS :

A. Representation of bandlimited signals (pulses) :

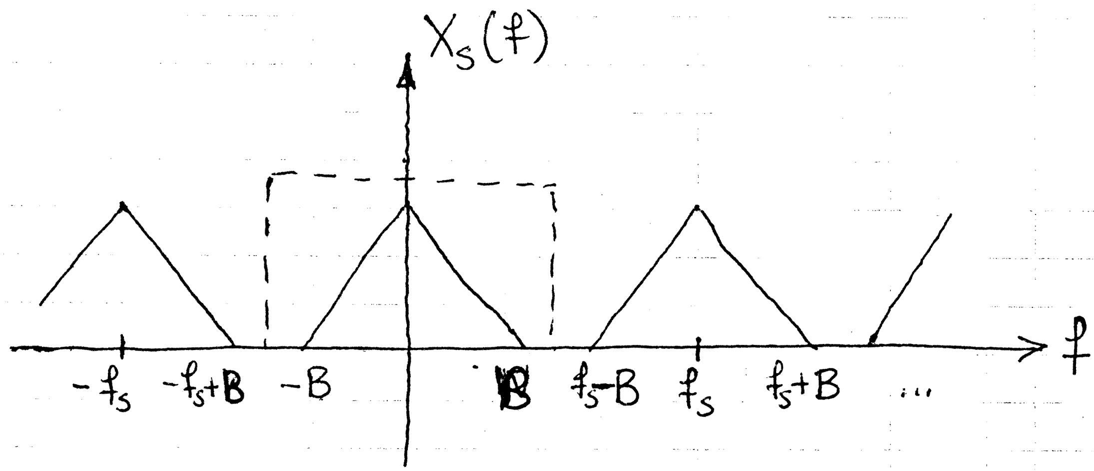


$$x_s(t) = x(t)\, s(t)$$

$$s(t) = \sum_k{}' \delta(t - kT) \iff S(f) = f_s \sum_k{}' \delta(f - f_s)$$

$$X_s(f) = X(f) * S(f)$$

$$= f_s \sum_k{}' X(f - k f_s)$$

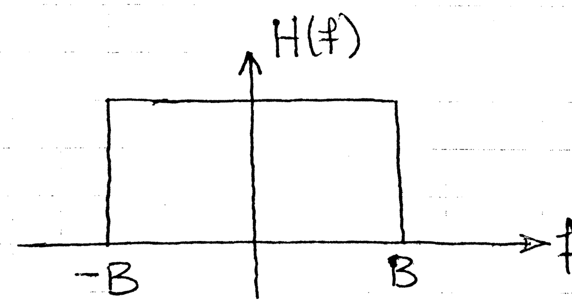


To avoid overlap of the various spectra

$$\boxed{f_s \geq 2B \quad \text{or} \quad T_s \leq \frac{1}{2B}}$$ Nyquist Sampling Theorem

min f_s is referred to as the Nyquist rate.

- Reconstruction:

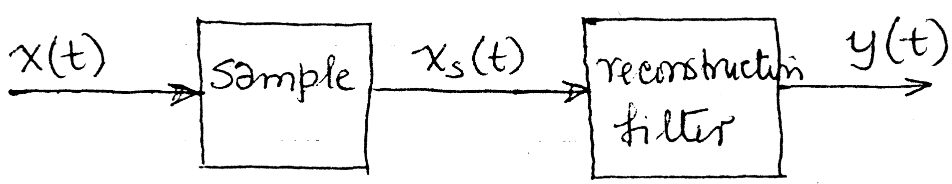


ideal LOW-PASS reconstruction filter:

$$H(f) = K\,\Pi\left(\frac{f}{2B}\right)e^{-j\omega t_0}$$

DELAY

RECTANGULAR FUNCTION



$$Y(f) = X_s(f) H(f)$$

$$= K f_s X(f) e^{-j\omega_0 t}$$

$\Longleftrightarrow$ $y(t) = K f_s \, x(t - t_0)$ orignal signal, scaled and delayed (no distortion)!

- look in the time domain

$$H(f) \Longleftrightarrow h(t) = 2BK \text{ sinc}\left[2B(t-t_0)\right]$$

$(t) \to (t - kT_s)$

$$y(t) = x_s(t) * h(t)$$

$$= \sum_k x(kT_s)\,\delta(t-kT_s) * h(t)$$

$$= \sum_k x(kT_s)\, h(t-kT_s)$$

$$= 2BK \sum_k x(kT_s) \text{ sinc } 2B(t - t_0 - kT_s)$$

The two $y(t)$ expressions should be equal; i.e.,

$$\cancel{K} f_s \, x(t-t_0) = 2B\cancel{K} \sum_k x(kT_s) \text{ sinc } 2B(t - t_0 - kT_s)$$

or

$$\boxed{x(t) = \frac{2B}{f_s} \sum_{k=-\infty}^{\infty} x(kT_s) \operatorname{sinc} 2B(t - kT_s)}$$

(replace $t - t_0$ by t)

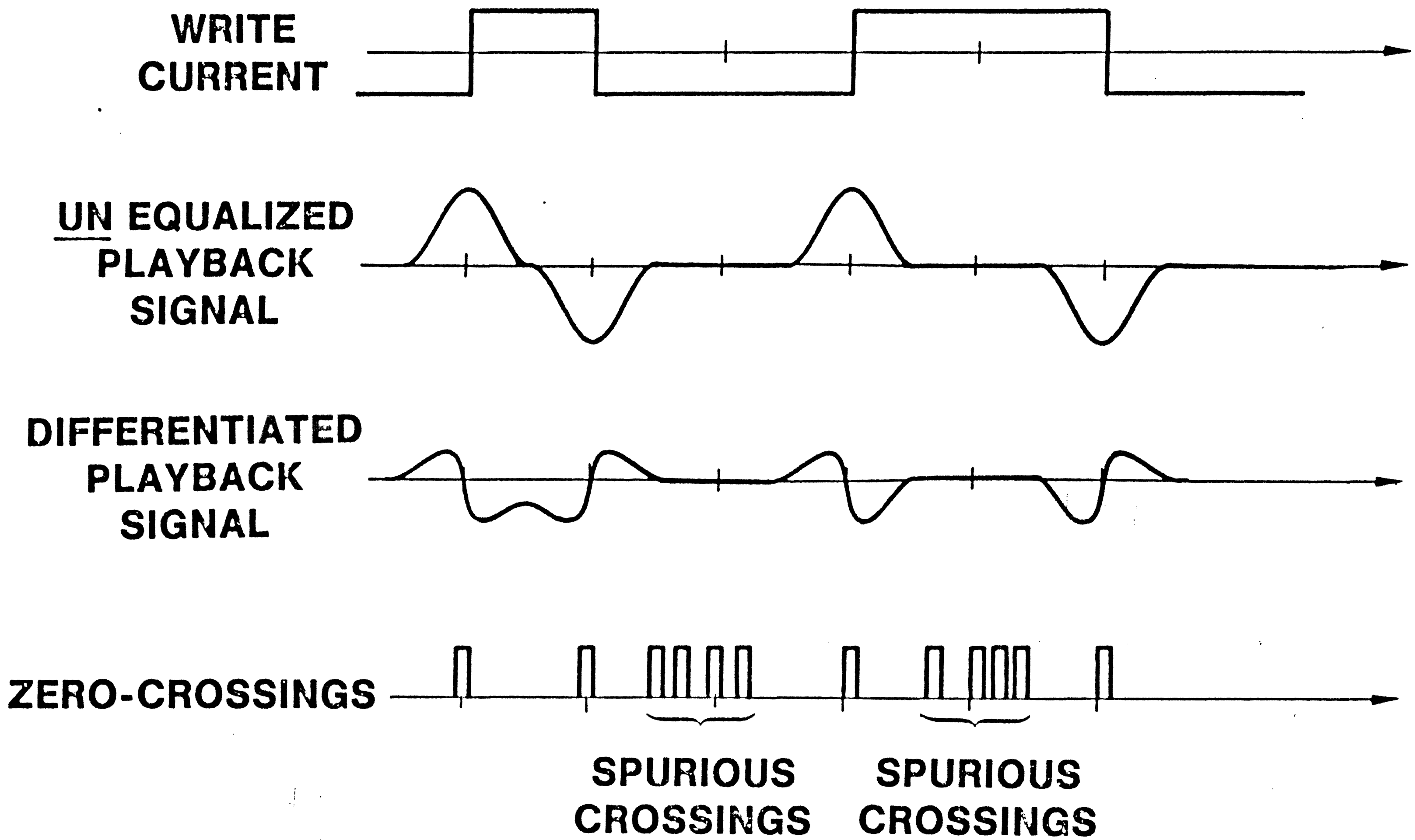
- if $B = f_s/2$ (i.e., f_s corresponds to the Nyquist rate) $f_s = \frac{1}{T_s}$

Then

$$\boxed{\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} x(kT_s) \operatorname{sinc} \frac{(t - kT_s)}{T_s} \\ &= \sum_{k=-\infty}^{\infty} x(kT_s) \frac{\sin \frac{\pi(t - kT_s)}{T_s}}{\frac{\pi(t - kT_s)}{T_s}} \end{aligned}}$$

$(f_s = 1/T_s)$

Thus, any bandlimited signal (pulse) can be represented by the above series. By choosing different values of $x(kT_s)$, we can synthesize different pulse shapes (and difference signal spectra).



PEAK DETECTION (DISCS)

APPLICATIONS OF A PEAK DETECTION CHANNEL MODEL

P. H. Siegel

Abstract - A computer model of a peak detecting magnetic recording channel has been implemented and used for channel design and performance evaluation. The model predicts raw error rate, ontrack and offtrack, as a function of linear density, run-length-limited (RLL) modulation code, write precompensation rules, and tapped-delay-line (TDL) equalizer. It assumes noise additivity and validity of linear superposition, and it bases calculations on a measured disk/electronics noise spectrum and digitized isolated transition readback signals from the data track and adjacent tracks. Details of the model are described, and illustrative applications to RLL (d,k) code selection and pulse slimming equalizer design for a specific channel are discussed.

INTRODUCTION

There are a number of signal processing options available which have the potential to increase areal density and reliability of peak detecting magnetic recording channels. Among these are modulation coding, write precompensation, and pulse slimming equalization. Assuming additivity of disk/electronics noise and adjacent track noise, and validity of linear superposition in the readback process, we have developed a model of a peak detection channel which predicts raw error rate as a function of linear density and specified signal processing. The basic methodology employed is similar to that suggested by Katz and Campbell [1].

Novel features of the model, in addition to the implementation of a general form of the Katz-Campbell theory, include the use of Shannon theory of source coding [2] to handle arbitrary RLL (d,k) codes [3], flexible write precompensation rules, and the incorporation of a TDL equalizer for general read equalization capability.

We discuss below some of the technical aspects of the model. We then address two applications to a specific disk channel: a comparison of RLL code performance, and the selection of a minimum noise pulse slimming equalizer.

INTERSYMBOL INTERFERENCE AND CODE PATTERNS

Intersymbol interference (ISI) affects the peak position and peak amplitude of the pulse resulting from a given transition. We compute an odd ISI length L, where L is the number of bits needed to account for pulse interactions. We calculate a cubic spline fit of a digitized readback pulse from the data track, as shown in Fig. 1. Then, using linear superposition, we simulate the readback signal corresponding to each (d,k) pattern of length L having a central transition. The differentiated signal is also calculated with the spline coefficients. The position of the central peak is located by use of a Newton-Raphson iterative search for a zero-crossing in the differentiated waveform, and the central peak amplitude is then found. The model next computes the values of the differentiated waveform at the edges of the detection window corresponding to the central transition. Two types of clocking are considered: an absolute clock and a mean-centered clock. The window for the mean-centered clock is centered around the average peak position described below, and represents an approximation to the window found in a channel with a PLL (phase locked loop). The waveform derivatives at the detection window edges are required for the bit shift error rate prediction. The average peak position is found by weighting the calculated peak positions for all patterns according to the Shannon pattern probabilities, and summing. The pattern probabilities indicate the frequency of occurrence of each pattern in encoded random data for an ideal (d,k) code. Since run-lengths are uncorrelated in an ideal code, the pattern probabilities are found by taking suitable products of run-length probabilities which we compute using techniques from information theory.

Manuscript received June 16, 1982.
The author is with IBM Research Laboratory, San Jose, California 95193, U.S.A.

Write precompensation rules can be specified in order to reduce the effects of intersymbol interference. The rules are pattern dependent adjustments of the recorded transition positions: a transition is advanced or delayed by a specified amount according to the code pattern context in which it occurs.

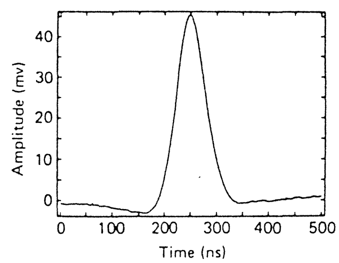


Fig. 1. Digitized isolated transition readback signal.

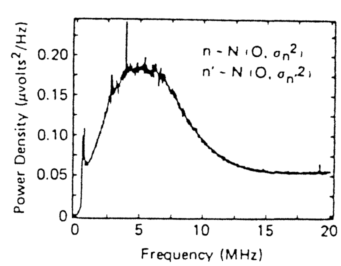


Fig. 2. Digitized disk/electronics noise power spectrum.

NOISE STATISTICS

The error rate calculation also requires a probability density function for noise and differentiated noise. For the disk/electronics noise, we digitize a noise power spectrum measured on a spectrum analyzer, as shown in Fig. 2. Normal probability plots of noise sample values measured from a dc-erased disk indicate that a Gaussian distribution fits the data out to at least three standard deviations. We take a Gaussian distribution for the disk/electronics noise, with mean zero and variance given by the numerical integral of the measured spectrum. A Gaussian distribution for the noise leads to simplifications in dealing with the differentiated noise as well. The derivative, n', of a Gaussian noise process n is again Gaussian [4], and the power spectrum T(f) of the differentiated noise is related to that of the original noise spectrum S(f) by the expression:

$$T(f) = (2\pi f)^2 \, S(f). \tag{1}$$

From the digitized S(f), we then compute the variance of the differentiated noise as the numerical integral of T(f). We model the distribution of n' as Gaussian with mean zero and with this variance.

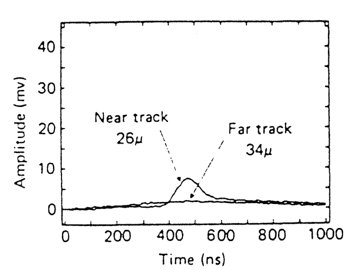


Fig. 3. Digitized adjacent track readback pulses, 4μ offtrack.

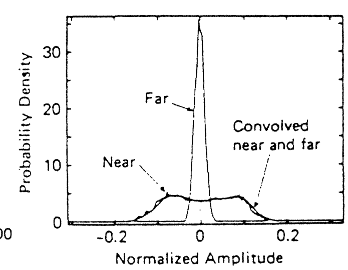


Fig. 4. Sample histograms from simulated adjacent track waveforms.

For adjacent track interference (cross-talk), we assume that side-reading of the nearest adjacent track on each side of the data track dominates the cross-talk signal. The readback signal from an isolated transition written on the adjacent track is digitized for the head position of interest. Figure 3 shows the readback pulses from the near and far adjacent tracks when the head is 4μ offtrack. Using a cubic spline fit and linear superposition, the readback waveform from several hundred bits of a pseudorandom (d,k) coded sequence is simulated and sampled up to 20 times per clock period. A histogram is made from the sample set as an estimate of the distribution density of samples from the adjacent track. If we assume no correlation between signals from different tracks, the total cross-talk density is estimated by taking the discrete

0018-9464/82/1100-1250$00.75©1982 IEEE

convolution of the histogram densities from the two tracks. Histograms from a 4μ head offset at a linear density of 18 kbpi with the (2,7) code are shown in Fig. 4, along with their convolution. The analogous calculation is then carried out for the samples of the waveform derivative. The distribution for combined disk/electronics and cross-talk noise can then be computed by discrete convolution.

ERROR RATE CALCULATION

Given that the signal has a pulse peak in the detection window W, the probability of noise-induced bit shift error is the probability that the differentiated signal plus differentiated noise waveform will fail to have a zero crossing within W:

$$Pr(\ s'(t) + n'(t) < 0 \ or \ s'(t) + n'(t) > 0 \ , for \ t \ in \ W\ ) \ . \quad (2)$$

Solving for this level-crossing probability exactly is a difficult mathematical problem, even when n is a Gaussian process. We use instead a convenient approximation, suggested by A. Milewski, which is a reasonably tight upper bound in the case of bandlimited noise. The probability is bounded above by the sum of the probabilities of the two events, for which simple upper bounds exist. For the first event, let t_1 be the time where the signal derivative is positive and of largest magnitude. Then,

$$Pr(\ s'(t) + n'(t) < 0 \ , for \ t \ in \ W\ ) \le Pr(\ n'(t_1) < -s'(t_1)\ ) \ . \quad (3)$$

Similarly, if t_2 is the time where the signal derivative is negative and of largest absolute value,

$$Pr(\ s'(t) + n'(t) > 0 \ , for \ t \ in \ W\ ) \le Pr(\ n'(t_2) > -s'(t_2)\ ) \ . \quad (4)$$

In practice, t_1 and t_2 have been found to lie at the detection window edges for the channels and densities studied. So, the approximations are evaluated at those points, using the waveform derivatives at the window edges and the noise distributions described above. We note that this upper bound has proven to be tighter than the approximation suggested in [1] which extrapolates the waveform derivatives at the window edges from the zero-crossing t_0 along a line of slope $s''(t_0)$. See Fig. 5.

The probability of missing bit error depends on the clip level C, which represents the minimum amplitude necessary to detect a peak in the readback signal. The probability of interest is

$$Pr(\ |s(t) + n(t)| < C, for \ t \ in \ W\ ) \ . \quad (5)$$

This represents a level-crossing probability which we approximate with the simple upper bound

$$Pr(\ |s(t_0) + n(t_0)| < C\ ) \ . \quad (6)$$

This probability is evaluated with the computed signal peak amplitude $s(t_0)$ and the noise distributions. See Fig. 6.

These error probability bounds may be used for approximate worst case pattern analysis. A weighted average using Shannon pattern probabilities provides an estimate of the overall error rate for encoded random data.

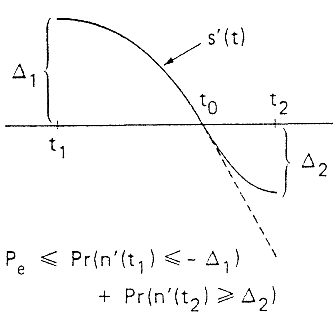


Fig. 5. Peak shift error rate approximation.

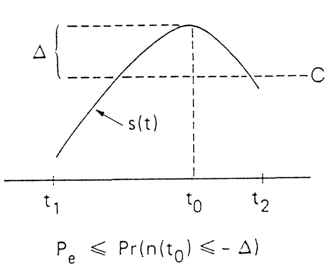


Fig. 6. Missing bit error rate approximation.

APPLICATIONS

We now discuss two applications of the model to a disk channel with a thin film head and particulate medium. Measurements were made at the inner diameter. Track pitch was 30μ.

RLL code comparison

We predicted the performance of (1,8) and (2,7) codes at three head positions - ontrack, 2μ offtrack and 4μ offtrack. The clip level was assumed to be 40% of the base-to-peak amplitude of the data track pulse. No write precompensation or pulse slimming equalization was used. The resulting error rate/linear density tradeoff curves are shown in Figs. 7 and 8. For the (1,8) code, peak shift errors dominated at densities less than 20 kbpi, while missing bits were the major error mechanism at higher densities. For the (2,7) code, however, peak shift errors were the primary determinant of error rate at all densities considered. The model indicates that at densities less than 20 kbpi, the (1,8) code has lower average error rates than the (2,7) code. At higher densities, the loss of signal amplitude degrades the (1,8) performance. In the range of error rates from 1E-12 to 1E-8, the (1,8) code provides a density advantage of slightly more than 5%. This result is consistent with the conclusions reached by Fisher and Newman in [5].

Table I shows a list of worst case patterns with L = 15 for the density 18 kbpi, as calculated by the model, along with peak shift, peak amplitude, and probability of error for ontrack operation. In general, the worst case patterns highlight features of the digitized pulses and can be used to assess the impact of peculiarities of pulse shape on error rate. Here, patterns with a minimum length run followed by a long run clearly affect the performance most severely, reflecting the pulse asymmetry.

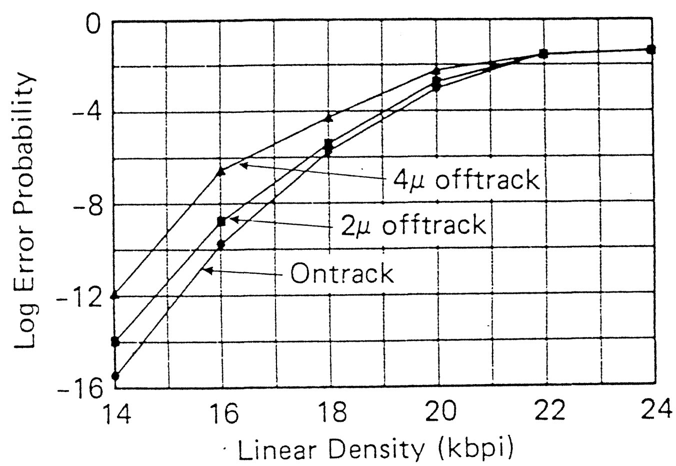


Fig. 7. Simulated performance of (1,8) code.

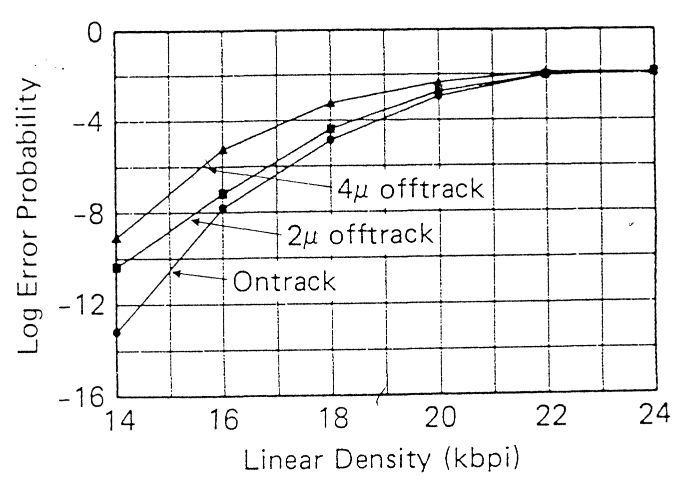


Fig. 8. Simulated performance of (2,7) code.

Peakshift (ns)	Relative Amplitude	Probability of Error	Pattern
7.1	.85	4.13E-5	100001010000101
7.0	.84	3.14E-5	010001010000101
7.1	.86	2.94E-5	100001010000100
7.2	.83	2.59E-5	000101010000101
7.0	.86	2.23E-5	010001010000100

(1,8) code, 25.1 ns window

5.6	.90	1.73E-3	010010010000001
5.3	.89	8.83E-4	000010010000001
5.5	.91	7.40E-4	100010010000001
5.6	.92	5.77E-4	010010010000010
5.2	.87	4.22E-4	010010010000000

(2,7) code, 18.9 ns window

TABLE I: Worst case patterns at 18 kbpi.

Pulse slimming equalizer evaluation

Barbosa [6] has reported on a design method for minimum noise pulse slimming equalizers. For a given channel and linear density, he constructs a one-parameter family of TDL equalizers, each of which maximizes the degree of slimming subject to a noise penalty constraint. At densities from 14 to 24 kbpi, we used the model to select the noise penalty for which the corresponding equalizer gives the smallest average ontrack error rate. The (2,7) code was used, and no cross-talk was considered. The ontrack and offtrack performance of the selected equalizer was then calculated, with cross-talk included. The results for the equalized channel with (2,7) code are shown in Fig. 9.

The conclusion based on the ontrack performance is that these equalizers can increase linear density between 10% and 20% in the range of ontrack error rates from 1E-12 to 1E-8. The equalized channel is not sensitive to small offtrack excursions, but the offtrack performance deteriorates as offtrack distance increases from 2μ to 4μ because of the enhancement of the cross-talk signal by the equalizer.

The worst case patterns were found to reflect the positions of the sidelobes of the equalized pulse. For example, at 20 kbpi, with a detection window of 17.05 ns, and with the equalized pulse shown in Fig. 10, the worst case patterns had runs of 4 zeros preceding and following the central transition, that is, 1 0 0 0 0 1 0 0 0 0 1 .

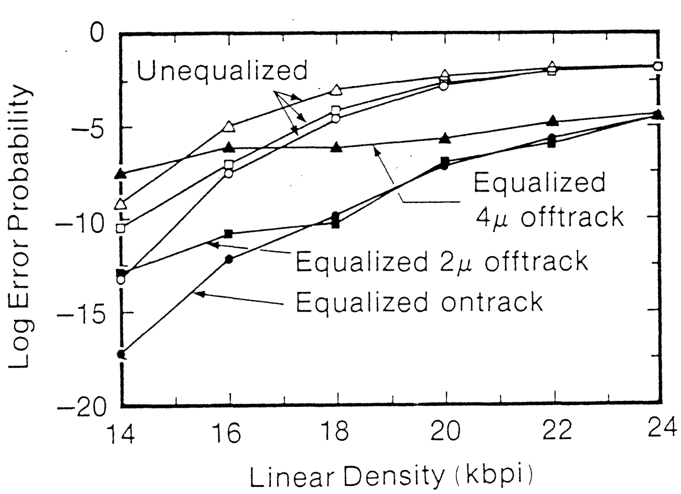


Fig. 9. Simulated performance of TDL equalizer with (2,7) code.

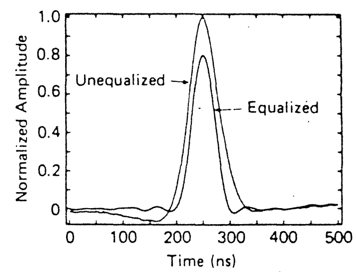


Fig. 10. Comparison of unequalized pulse and 20 kbpi equalizer pulse.

CONCLUSIONS

We have described a computer model which predicts raw error rates for a peak detecting magnetic recording channel. Offtrack performance is predicted by inclusion of adjacent track interference effects. Calculations are based on measured channel characteristics: step responses from the data and adjacent tracks, and a disk/electronics noise spectrum. The model also permits the evaluation of several signal processing options, individually and in combination: RLL code, write precompensation, and pulse slimming equalization. In addition to error rates, the model provides useful information about dominant error mechanisms and error sources both for worst case code patterns and for random coded data. Results of (d,k) code comparison and TDL equalizer evaluation for a specific disk channel were discussed.

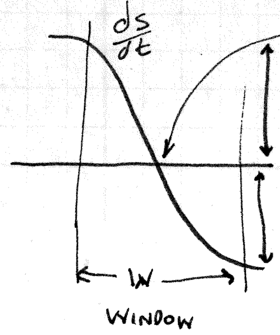
ACKNOWLEDGMENTS

The author would like to acknowledge the early work on peak detection channel models by T. A. Schwarz.

REFERENCES

1. E. R. Katz and T. G. Campbell, "Effect of bitshift distribution on error rate in magnetic recording," IEEE Trans. Magn., vol. MAG-15, pp.1050-1053, May 1979.
2. C. E. Shannon, "A mathematical theory of communication," Bell System Tech. J., vol. 27, pp. 379-423, July 1948, and pp. 623-656, October 1948.
3. P. A. Franaszek, "Sequence-state methods for run-length-limited coding," IBM J. Res. Dev., vol. 14, pp. 376-383, July 1970.
4. D. Middleton, Introduction to Statistical Communication Theory, New York: McGraw-Hill Book Co., 1960, pp. 370-372.
5. R. D. Fisher and J. J. Newman, "Code performance and head/media interface," IEEE Trans. Magn., vol. MAG-17, pp. 1452-1454, July 1981.
6. L. C. Barbosa, "Minimum noise pulse slimmer," IEEE Trans. Magn., vol. MAG-17, pp. 3340-3342, Nov. 1981.

4/26/89



ZERO CROSSING IS PEAK

TAKE HOME MIDTERM

SIGNAL AMPLITUDE
NOISE PUSHES GRAPH UP
LOW AMPLITUDE PEAKS MOST SUCEPTIBLE TO NOISE

LINING UP PEAKS WITH CLIPPED TRANSITIONS BOOST HIGH FREQ RESPONSE.
(LIKE EQUALIZATION)

PULSE AMPLITUDE MODULATION

PULSE POSITION MODULATION

Electronics and Communications in Japan, Vol. 59-C, No. 10, 1976

Peak Shift Caused by Gaussian Noise in Digital Magnetic Recording

Yasuhiro Tahara, Yoshimasa Miura and Yoshinori Ikeda

Fujitsu Laboratories Ltd., Kawasaki, Japan 211

SUMMARY

The peak shift generated in digital magnetic recording processes is one of the most important obstacles to high-density recording. The principal causes of peak shifts are waveform interference effects and noise. Of these causes, only the noise components have been subjected to empirical treatment. In this paper we developed a probabilistic analysis of the peak shift due to noise. Generations of the peak shift are treated as probabilistic distributions and the corresponding distribution functions and contribution to the phase margin are theoretically derived. The results show that when Gaussian noise is superimposed on read-out signals from the head, generation of peak shifts due to the noise also exhibits a Gaussian distribution. With the variance of the distribution as σ^2, the maximum peak shift is 55 ~ 7 σ and the loss of phase margin is 11 ~ 14 σ. The theory is applied to the MFM recording system and the peak shifts of 2F, 1F and |110| patterns due to the white noise are obtained. The ratio of these peak shifts takes an almost constant value of 1:1.3:1.2 in the region where the resolving power is 50 to 70%. It is found that the theoretical prediction and the experimental data agree very well for |110| patterns.

1. Introduction

Improvement of recording density is an important problem in magnetic devices such as magnetic disks and drums. One of difficulties encountered in high-density recording systems is the generation of peak shifts. Information written on disks and drums as magnetic reversal patterns is read out at magnetic heads and regenerated by detecting the peaks of read-out waveforms. When the recording density on disks and drums is increased, peak shifts in the read-out waveforms become larger due to interbit waveform interference, resulting in degradation of the SN ratio. If peak shifts become excessively large, read-out errors occur and it is no longer possible to derive recorded informations from the read-out waveforms.

Principal causes of peak shifts are waveform interference and waveform jitter due to noise. A number of theoretical and experimental studies have been conducted on the waveform interference effects [1-3] and some efforts to reduce peak shifts have been tested using circuit techniques, such as waveform equalization, that make use of waveform characteristics [4, 5]. On the other hand, only empirical treatments have been done on peak shifts caused by noise and no specific quantitative analysis has been conducted. In present-day magnetic recording devices, SN ratios are steadily decreasing because of the reduction of read-out voltages in the head as accompanied with increased recording densities and because of the increase in noise bandwidth due to higher recording and regenerating frequencies. As a result, the effect of the noise has increased and methods must be developed for quantitative analysis of peak shifts caused by the noise. Furthermore, a design procedure for recording and regenerating systems is needed by which the overall peak shifts caused by both waveform interference and noise effects may be minimized.

One of the characteristics of the peak shift caused by the noise is its randomness. This is because the noise generation is also random. Hence, for quantitative treatment of the peak shifts caused by noise it is necessary to introduce a probabilistic approach. Although Mallinson [6] and Kobayashi [7] analyzed noise in NRZ recording systems in a probabilistic manner, they have not considered peak shifts at all.

In this paper, a method previously proposed by the authors [8] is extended to the probabilistic analysis of peak shifts caused by noise, and the probability distribution of peak shifts and their contribution to the phase margin are quantitatively derived. The theory is applied to the case of MFM recording systems, and the amounts of peak shifts caused by the noise are computed for several practical patterns. Finally, experimental results are compared with theoretical predictions.

2. Peak Shifts Caused by Noise

2.1 Noise

In magnetic disks and drums, information written on the recording medium as magnetization reversal patterns is read out at the magnetic head and immediately amplified by a preamplifier located near the head. The amplified signal is then sent to a peak detector at the later stage of the system. Principal causes of noise are (see Fig. 1):

1. Preamplifier noise
2. Head impedance noise
3. Medium noise

The first of these causes arises from semiconductor noise generated in the preamplifier and consists of thermal noise, shot noise and 1/f noise. At the 1 to 10 MHz used in magnetic disks and drums, 1/f noise is negligible, and hence thermal noise and shot noise are predominant. The spectrum of the latter two is almost identical to that of white noise. The head impedance noise is a kind of thermal noise caused by the composite impedance of the head and head termination circuit as seen from the preamplifier. The spectrum distribution is concentrated near the resonance frequency of the head [9]. The medium noise is caused by nonuniform dispersion of magnetic particles in the recording medium [10, 11]. It is read out with the information signal by the head.

These kinds of noise are generated randomly and hence the noise distribution can be treated as Gaussian.

Let us expand the noise voltage $V_n(t)$ into Fourier series at $-T \leq t \leq T$

$$V_n(t) = \sum_{n=1}^{\infty} (a_n \sin \omega_n t + b_n \cos \omega_n t) \tag{1}$$

where

$$\omega_n = 2\pi f_n = \pi n / T \tag{2}$$

and a_n and b_n are probability variables that independently obey Gaussian distributions. Distribution functions of a_n and b_n are identical and their mean values are zero.

$$\left.\begin{aligned} \overline{a_n^2} &= \overline{b_n^2} \\ \overline{a_n} &= \overline{b_n} = 0 \end{aligned}\right\} \tag{3}$$

The power spectral density of this noise (defined as the mean square noise voltage per unit bandwidth at frequency of f_n) is given by

$$N(f_n) = \frac{\frac{1}{2}\overline{(a_n^2 + b_n^2)}}{\Delta f} = \frac{\overline{a_n^2}}{\Delta f} = \frac{\overline{b_n^2}}{\Delta f} \tag{4}$$

where $\Delta f = 1/2T$.

2.2 Peak shift caused by noise—sinusoidal waves

Let us consider what kind of peak shifts will be produced when the noise described above is superposed on the regenerated signals. We first examine the most fundamental case, in which the read-out waveforms are described in terms of sinusoidal waves. The total regenerated signal voltage V(t) is

$$V(t) = \frac{1}{2} V_0 \cos \omega_0 t + \sum_{n=1}^{\infty} (a_n \sin \omega_n t + b_n \cos \omega_n t) \tag{5}$$

where V_0 and ω_0 are the amplitude and angular frequency of the regenerated signal, respectively.

We shall now obtain the shift of the peak, originally located at t = 0, caused by the noise. First V(t) is differentiated and expanded around t = 0, assuming the amount of the peak shift is small. The result is

$$\frac{d}{dt} V(t) = -\frac{1}{2} \omega_0^2 V_0 t + \sum_{n=1}^{\infty} \omega_n a_n \tag{6}$$

From this equation the amount of the peak shift is

$$\Delta t = \frac{2}{\omega_0^2 V_0} \sum_{n=1}^{\infty} \omega_n a_n \tag{7}$$

The mean square value of σ^2 of Δt is given by

$$\begin{aligned} \sigma^2 = \overline{(\Delta t)^2} &= \frac{4}{\omega_0^4 V_0^2} \overline{\left(\sum_{n=1}^{\infty} \omega_n a_n \right)^2} \\ &= \frac{4}{\omega_0^4 V_0^2} \sum_{n=1}^{\infty} \omega_n^2 \overline{a_n^2} \\ &= \frac{4}{\omega_0^4 V_0^2} \sum_{n=1}^{\infty} \omega_n^2 N(f_n) \Delta f \\ &= \frac{4}{\omega_0^4 V_0^2} \cdot \frac{1}{2\pi} \int_0^{\infty} \omega^2 N(\omega) d\omega \end{aligned} \tag{8}$$

where $N(\omega) = N(f_n)$.

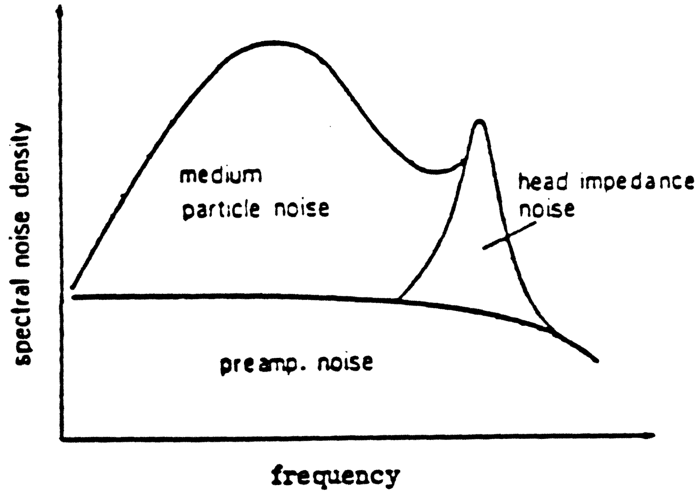


Fig. 1. Noise spectrum in magnetic recording file.

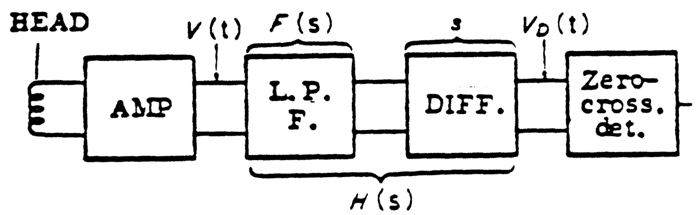


Fig. 2. Read circuit.

Next, let us examine the distribution function of the peak shift Δt. Notice that a_n in (7) has a Gaussian distribution. In general, when variables x_i ($i = 1, \ldots, N$) have independent Gaussian distributions the variable x given by

$$x = \sum_{i=1}^{N} c_i x_i \tag{9}$$

where the c_i are constants, also has a Gaussian distribution and its variance is given by

$$\sigma_x^2 = \sum_{i=1}^{N} c_i^2 \sigma_i^2 \tag{10}$$

where σ_i^2 is the variance of x_i [12]. Hence, if a_n has a Gaussian distribution, so does Δt. The variance of Δt is of course given by (8). The distribution function $p(\Delta t)$ is now

$$p(\Delta t) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(\Delta t)^2}{2\sigma^2}\right) \tag{11}$$

where σ^2 is given by (8).

2.3 Peak shift caused by the noise—general case

Let us now derive a method for computing peak shifts in general cases. The read circuit of a digital magnetic recording device generally consists of an amplifier, low-pass filter, differentiation circuit, and zero-crossing detector as shown in Fig. 2. Since operation of these circuits is not ideal, the effects of their frequency characteristics on the peak shift cannot be neglected.

For simplicity, we represent the frequency characteristics of the entire read circuit by that of a low-pass filter. When the transfer function of the low-pass filter is F(s), the transfer function H(s) of the read circuit, which contains a differentiation circuit, is given by

$$H(s) = s \cdot F(s) \tag{12}$$

as is seen from Fig. 2. If an RC approximate differentiation circuit shown in Fig. 3 is used in place of a true differentiation circuit, the transfer function H(s) becomes

$$H(s) = \frac{s}{1+\frac{s}{\omega_d}} \cdot F(s) = s \cdot \frac{s}{1+\frac{s}{\omega_d}} F(s) \tag{13}$$

where $\omega_d = 1/RC$. Hence, if another first-order low-pass filter is inserted, the transfer function can be reduced to that of (12).

Consider now the case in which a read-out signal f(t) is indicent at the circuit. Since the total input signal including the noise is

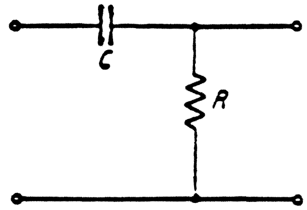


Fig. 3. Approximate differentiation circuit.

$$V(t)=f(t)+\sum_{n=1}^{\infty}(a_n\sin\omega_n t+b_n\cos\omega_n t) \tag{14}$$

the output $V_D(t)$ of the differentiation circuit is

$$V_D(t)=\overset{\text{Signal}}{h(t)}+\sum_{n=1}^{\infty}\omega_n\cdot|F(j\omega_n)|$$

$$\times\{a_n\cos(\omega_n t+\theta_n)-b_n\sin(\omega_n t+\theta_n)\} \tag{15}$$

where h(t) is the output signal of the differentiation circuit for the read-out signal f(t) and is given by

$$\mathcal{L}[h(t)]=H(s)\cdot\mathcal{L}[f(t)] \tag{16}$$

using the transfer function H(s). θ_n is the phase of $F(j\omega_n)$.

Let the zero-crossing point of h(t) be t_0 and the slope of h(t) near t_0 be G_0, i.e.,

$$G_0=\frac{d}{dt}h(t)\Big|_{t=t_0} \tag{17}$$

Then the zero-crossing point of the output $V_D(t)$ is given by

$$t=t_0+\frac{1}{G_0}\sum_{n=1}^{\infty}\omega_n\cdot|F(j\omega_n)|$$

$$\times\{a_n\cos(\omega_n t_0+\theta_n)-b_n\sin(\omega_n t_0+\theta_n)\} \tag{18}$$

The first term t_0 represents the peak shift caused by the waveform interference and the phase delay in the circuit, whereas the second term corresponds to that caused by the noise. Hence, the mean square value of the peak shift caused by the noise is

$$\sigma^2=\frac{1}{G_0^2}\cdot\frac{1}{2\pi}\int_0^{\infty}\omega^2|F(j\omega)|^2N(\omega)d\omega \tag{19}$$

or, using (12), is given by

$$\sigma^2=\frac{1}{G_0^2}\cdot\frac{1}{2\pi}\int_0^{\infty}|H(j\omega)|^2N(\omega)d\omega \tag{20}$$

where the term

$$\frac{1}{2\pi}\int_0^{\infty}|H(j\omega)|^2N(\omega)d\omega$$

represents the noise power contained in the output of the differentiation circuit. Therefore, the peak shift caused by the noise can be described in terms of the noise power in the output and the slope at the zero-crossing point.

3. Peak Shifts of Various Patterns in MFM Recording Systems Due to Noise

In conventional digital magnetic recording devices, the recording and read-out of information are performed using PM, FM or MFM recording processes in which self-locking can be incorporated. In most recent large-capacity recording devices, the MFM process is employed. In this section, the effect of waveform interference on the peak shift caused by noise is investigated. To this end, peak shifts caused by white noise are calculated and compared for 2F, 1F and {110} patterns (Fig. 4).

3.1 2F pattern

In the MFM recording processes, the 2F patterns have the highest magnetization reversal frequency. Ordinarily, the recording density in the MFM process is such that the resolution is 60 to 70%, within which range the read-out signals of the 2F patterns are almost sinusoidal and the contribution of harmonics is negligible. Hence, the read-out signal waveform $f_2(t)$ is

$$f_2(t)=\frac{1}{2}V_{2F}\cos\omega_0 t \qquad (.$$

where V_{2F} is the peak-to-peak amplitude. Here ω_0 ($=2\pi f_0$) is the recording and read-out angular frequency and is related to the bit cell time T via

$$\omega_0=\pi/T \tag{22}$$

Let the power spectrum of the white noise be

$$N(\omega)=n_0^2 \tag{23}$$

We assume that the low-pass filter is an ideal with the characteristics

$$F(j\omega)=\begin{cases}1 & : f\leq f_c\\ 0 & : f>f_c\end{cases} \tag{24}$$

Then the peak shift is computed from (19).

$$\sigma_{2F}=\frac{f_c^{3/2}n_0}{\sqrt{3}\,\pi f_0^2 V_{2F}} \tag{25}$$

If the noise power is written as N_{rms}^2

$$N_{rms}^2=f_c\cdot n_0^2 \tag{26}$$

then σ_{2F}

$$\sigma_{2F}=\frac{f_c}{\sqrt{3}\,\pi f_0^2}\cdot\frac{N}{V_{2F}} \tag{27}$$

Notice that the amount of the peak shift is proportional to $f_c^{3/2}$, as shown in (25). This is because the higher frequency content of the noise contributes more to the peak shift. Hence, in the read-out of magnetic recording systems the importance of elimination of high-frequency noise by the low-pass filter is more than just for improvement of the SN ratio.

3.2 1F patterns

The 1F patterns have the lowest magnetization reversal frequency in MFM recording processes. As seen from Fig. 4 (b), the read-out signals contain the 3F harmonics. Letting the amplitude of the read-out signal be V_{1F} and the ratio of the 3F components be β, we can write the read-out waveforms of 1F patterns as

$$f_{1F}(t)=\frac{1}{2}V_{1F}\left\{(1-\beta)\cos\frac{1}{2}\omega_0 t+\beta\cos\frac{3}{2}\omega_0 t\right\} \quad (28)$$

where β is usually 0.1 to 0.2. The amount of peak shifts of 1F patterns caused by the white noise can be obtained in a manner similar to the case of 2F patterns and is given by

$$\sigma_{1F}=\frac{4f_c}{\sqrt{3}\,\pi f_0^2}\cdot\frac{N_{rms}}{(1+8\beta)V_{1F}} \quad (29)$$

The peak shift of 1F patterns depends on V_{1F} as well as β. This is because the amount of shift is not a function of the amplitude but of the sharpness of the peak. Hence, the peak shift caused by the noise is less likely to occur for the case with larger resolving power and β.

Next, the magnitudes of peak shifts of 1F and 2F patterns will be compared. If the noises are identical in both cases, the ratio of σ_{1F} and σ_{2F} is obtained from (27) and (29) as

$$\frac{\sigma_{1F}}{\sigma_{2F}}=\frac{4R}{1+8\beta} \quad (30)$$

where R is the resolving power given by

$$R=V_{2F}/V_{1F} \quad (31)$$

Let an isolated read-out waveform be

$$e(t)=A\cdot\frac{a}{a^2+t^2} \quad (32)$$

where A and a are constants for expressing waveforms. If we assume that the read-out signal waveforms can be expressed in terms of a superposition of isolated waveforms, β is given by

$$\beta=\frac{1-\sqrt{1-R^2}}{2} \quad (33)$$

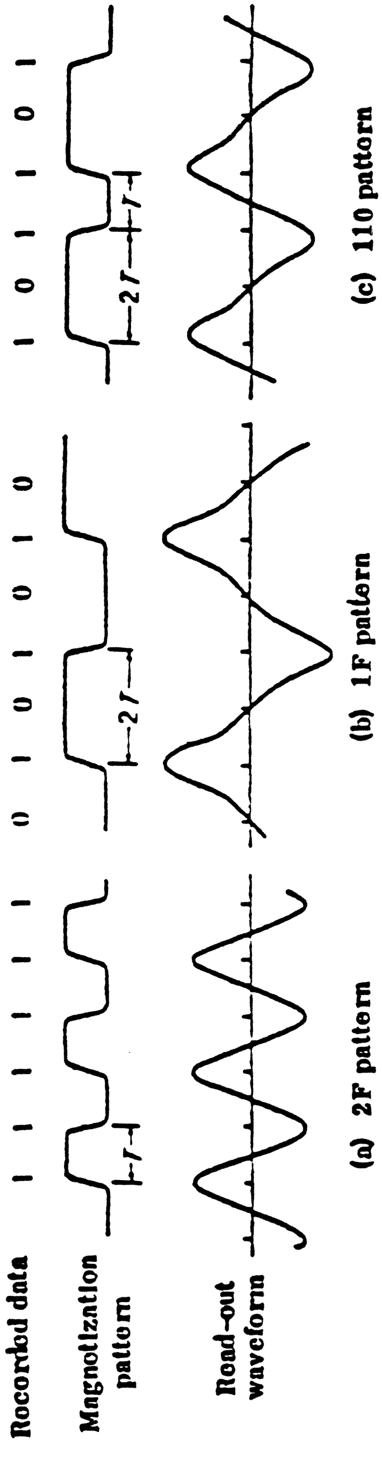


Fig. 4. Read-out waveforms of various patterns in MFM recording.

(see the Appendix). Hence, (30) becomes

$$\frac{\sigma_{1F}}{\sigma_{2F}} = \frac{4R}{5-4\sqrt{1-R^2}} \tag{34}$$

Figure 5 shows σ_{1F}/σ_{2F} versus R. From the figure, it is clear that 1F patterns are approximately 30% more likely to be affected by noise 30%.

3.3 {110} patterns

In MFM recording systems the maximum peak shift caused by waveform interference occurs for "110110" patterns. The read-out waveforms of these {110} patterns can be represented by using (4/3) F and (8/3) F components as

$$f_{110}(t) = \frac{1}{2}V_{110}\left(\sin\frac{2}{3}\omega_0 t - \gamma\sin\frac{4}{3}\omega_0 t\right) \tag{35}$$

where V_{110} is the amplitude of (4/3) F components and γ is the amplitude ratio of (8/3) F to (4/3) F components. Peaks of these {110} patterns are located at

$$t = \frac{3T}{2\pi}\cos^{-1}\left(\frac{1-\sqrt{1+32\gamma^2}}{8\gamma}\right) \tag{36}$$

and the amount of peak shifts ΔT_{110} caused by waveform interference is

$$\Delta T_{110} = T\left\{1-\frac{3}{2\pi}\cos^{-1}\left(\frac{1-\sqrt{1+32\gamma^2}}{8\gamma}\right)\right\} \tag{37}$$

When white noise is superimposed on these {110} patterns, the amount of peak shift caused by the noise is

$$\sigma_{110} = \frac{3\sqrt{3}f_c}{4\pi f_0^2}\cdot\frac{N_{rms}}{\sqrt{A(\gamma)}\,V_{110}} \tag{38}$$

$$A(\gamma) = \frac{(1+32\gamma^2)(3+\sqrt{1+32\gamma^2})}{2\ (1+\sqrt{1+32\gamma^2})} \tag{39}$$

As in the case of 1F patterns, we consider the ratio of σ_{110} to σ_{2F}. If we assume that (32) represents the isolated read-out waveform, we obtain (see the Appendix)

$$\gamma = \left(\frac{1-\sqrt{1-R^2}}{R}\right)^{4/3} \tag{40}$$

$$\frac{V_{110}}{V_{2F}} = \frac{1}{\sqrt{3\gamma}} \tag{41}$$

Hence, if the noise is identical in both cases, we have

$$\frac{\sigma_{110}}{\sigma_{2F}} = \frac{9}{4}\sqrt{\frac{3\gamma}{A(\gamma)}} \tag{42}$$

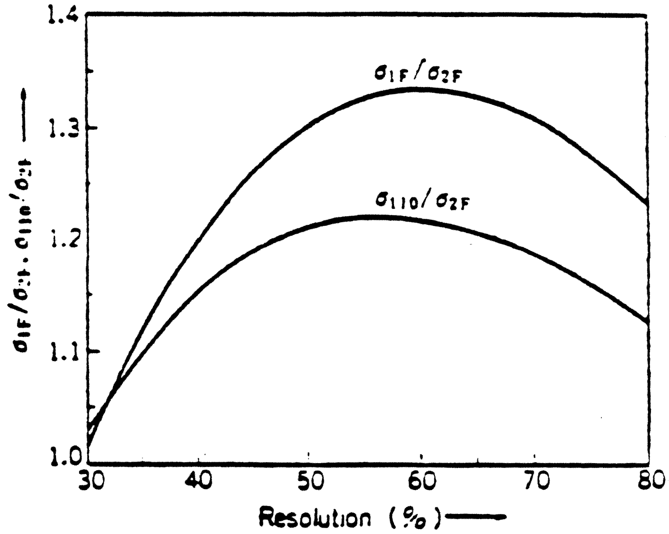


Fig. 5. Relative strengths of the peak shifts caused by noise as a function of resolution.

Figure 5 shows σ_{110}/σ_{2F} versus R. From the figure, we observe that {110} patterns are almost 20% more likely to be affected by noise.

The electromagnetic conversion characteristics of magnetic heads are often expressed in terms of the resolving power and the output voltage of 2F patterns. Actual peak shifts caused by the noise are larger for 1F and {110} patterns than for 2F patterns. For resolving power of 50 to 70%, the amount of shifts exhibits a constant ratio of 1 :1.3 :1.2. Therefore, the magnitude of phase shifts due to noise can be predicted from the output voltage, and hence from the SN ratio, of 2F patterns.

4. Phase Margin and Error Rate

The demodulation process in usual digital magnetic recording systems consists of the following steps. First, using a phase-lock loop, clock signals are generated from the data pulse train emerging from the peak detector. From these clock signals, window pulses are then created. Discrimination of 1 and 0 is done by detecting whether a particular data pulse is within the window pulse. The phase margin is defined as the difference between the maximum peak shift actually generated and the width of the window pulse. This quantity is viewed as a figure of reliability for magnetic recording devices. When the phase margin is sufficiently large, correct demodulation is possible even if new peak shifts are generated due to small defects on the recording medium or to tracking errors, as long as their magnitudes fall within the phase margin.

As shown in Fig. 6, measurements of the phase margin are performed by shifting the window pulse with respect to the data pulse and by detecting the error rate. The phase margin is the width of the

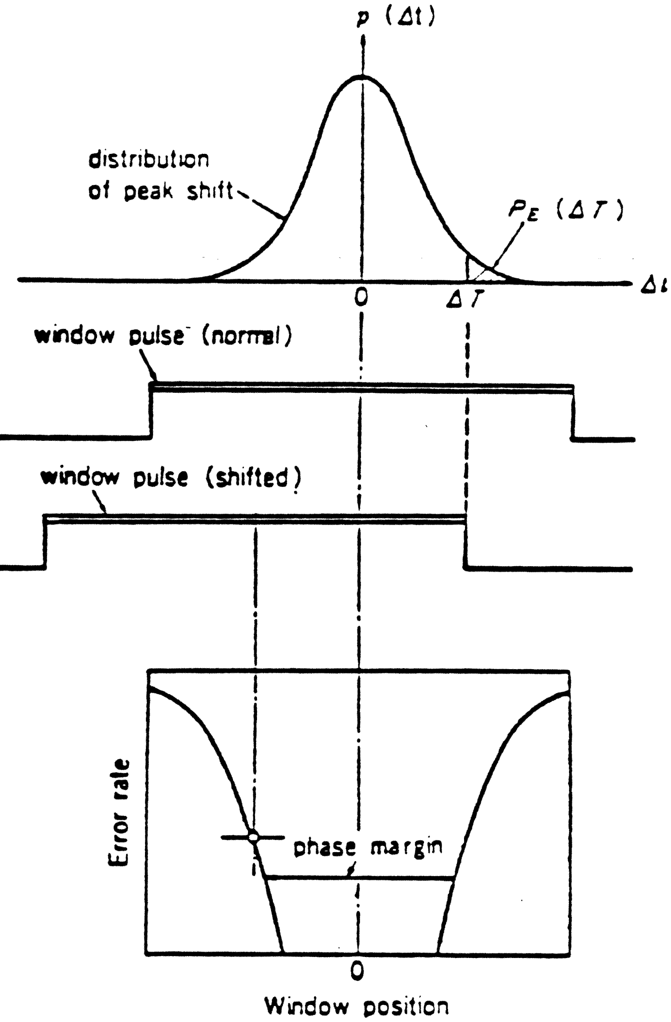


Fig. 6. Principle of the measurement of phase margin.

window pulse shift for which the error rate is below a certain value.

Let us derive the probability P(ΔT) at which a data pulse creates a peak shift larger than ΔT due to noise. The result is

$$P(\Delta T)=\int_{\Delta T}^{\infty} p(\Delta t)\,d\Delta t=\mathrm{erfc}\left(\frac{\Delta T}{\sigma}\right) \tag{43}$$

where erfc(x) is the error function. Figure 7 shows P(ΔT) versus ΔT/σ. Since the error rate in magnetic recording devices should be $10^{-8} \sim 10^{-12}$, we see that a maximum peak shift of 5.5 ~ 7σ is created. The phase margin is reduced by 11 ~ 14σ due to the noise.

In general, when a peak shift of ΔT_k (k = 1, ..., N) already exists due to waveform interference, the probability of generating peak shifts larger than ΔT is

$$P_E(\Delta T)=\sum_{k=1}^{N} w_k\,P(\Delta T-\Delta T_k) \tag{44}$$

where w_k is the ratio of pulses which cause the peak shift of ΔT_k.

5. Comparison with Experiment

In the experiment two kinds of magnetic heads and disks, A and B, were used to record and read out and the variation of the error rate was measured with respect to the location of the window pulse. The recording process was MFM and the recording and read-out frequency was 6.45 MHz. The cutoff frequency of the low-pass filter in the read-out circuit was 11.7 MHz. The read output of the head (V_{2F}), resolving power and noise were measured and are listed in Table 1. Although the head output of A is larger, so is the noise in A. The SN ratio of B is better by about 1 dB. Hence,

Table I. Recording and read-out characteristics of A and B

	A	B
Head output	1.32 mV_{p-p}	1.08 mV_{p-p}
Resolving power	61%	67%
Noise	31.2 μV_{rms}	22.7 μV_{rms}

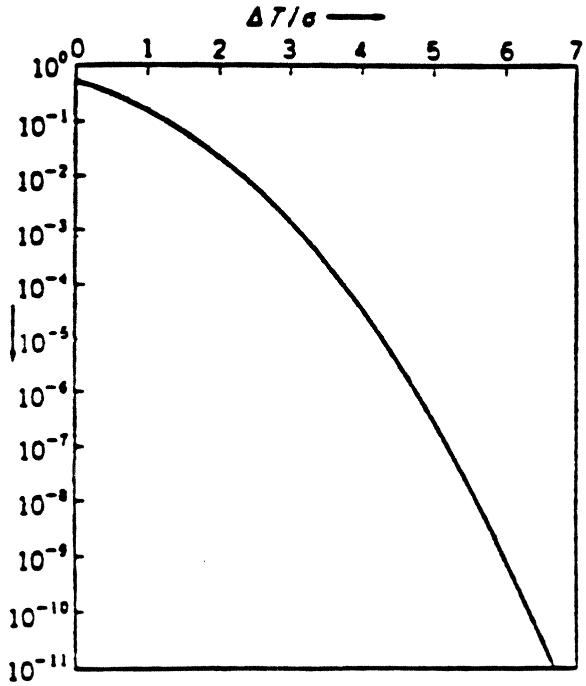


Fig. 7. Probability of peak shift larger than ΔT caused by noise.

it is expected that the resolving power and SN ratio are better and the peak shift smaller in B.

Figure 8 shows measured results of the error rate with respect to the window pulse location when {110} patterns are recorded and read out. As expected, the peak shift of A is much larger and the loss of phase margin is greater.

Values in Table 2 were calculated from the measured values in Table 1. In Table 2, the peak shift caused by waveform interference and the peak jitter due to noise are listed. Computed results of the error rate are solid lines in Fig. 8. Their agreement with experimental data is excellent. In the present experiment, the loss of phase margin due to noise was 30 to 40% of the window pulse width and we can see that the effect of noise on the peak shift is quite important.

6. Conclusion

The peak shift caused by noise was analyzed in a probabilistic manner and several examples were computed. The distribution and magnitude of the peak shift and its contribution to the phase margin were studied. Although, to date, peak shifts have been analyzed using empirical methods, the new method in this paper is capable of predicting more

Table 2. Computed values of peak shifts

	A	B
Waveform interfer. ΔT_{110}	6.0 ns	4.6 ns
Jitter by noise σ_{110}	1.46 ns	1.28 ns

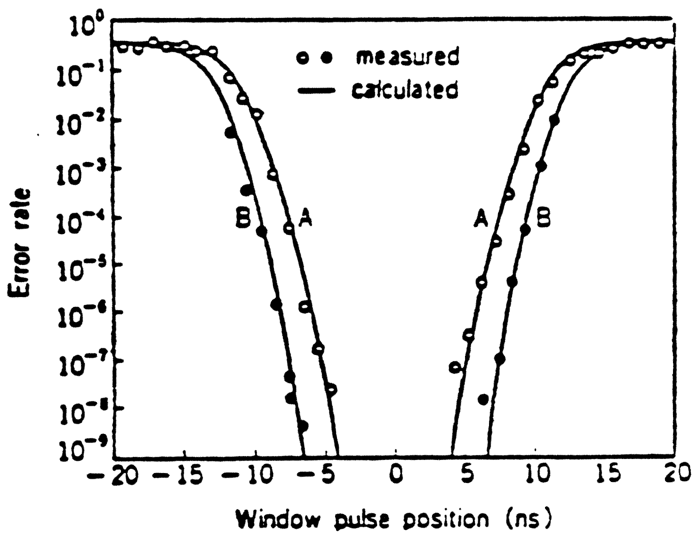


Fig. 8. Experimental results of the error rate for the {110} pattern.

accurate values. In future designs of magnetic heads or recording and reading circuits, the total peak shift caused both by waveform interference and by noise must be taken into account. The present method is useful for the optimum design of such devices.

One of the problems yet to be analyzed is waveform equalization by such circuits as pulse-narrowing networks. When the read-out waveforms go through a waveform equalizer, the peak shift due to waveform interference may be reduced, whereas that due to noise may increase. Waveform equalization is useful only when the reduction of the peak shift due to waveform interference is larger than the increase of the peak shift caused by noise. Since the SN ratio gradually decreases as the recording density is increased, the design of equalizers must be done with extreme caution.

In the present paper the peak shift was assumed small in order to simplify calculation was simplified. When the peak shift is extremely large, this simplification is no longer valid and the distribution is degraded from a Gaussian form. However, in conventional devices, the SN ratio is larger than 20 dB and the simplification described above is believed not to cause any problems.

Acknowledgement. The authors thank President Kojima and Vice President Utoro of Fujitsu Laboratories for their guidance. They also thank the members of Laboratory No. 4 of the Materials Department and Laboratory No. 2 of the Mechanism Research Department for their cooperation.

REFERENCES

1. Kameyama. Interference characteristics of high-density magnetic recording systems, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan (April 1969).
2. Koshimoto and Nishikawa. Wavelength characteristics of digital magnetic recording—read-out response, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan, MR69-2 (1969).
3. Hosokawa. Discussion on peak shift characteristics, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan, MR71-12 (1971).
4. H.M. Sierra. Increased magnetic recording read-back resolution by means of a linear passive network, IBM J., p. 22 (Jan. 1963).
5. Tachibana, Sata and Watanabe. A study of equalizers for digital magnetic recording read-out, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan, MR73-7 (1973).
6. J.C. Mallinson. On extremely high density digital recording, IEEE Trans., MAG-10, p. 368 (1974).
7. H. Kobayashi. Application of probabilistic decoding to digital magnetic recording systems, IBM J. Res., Develop., 15, p. 64 (1971).
8. Miura, Tahara and Ikeda. A study on noise in high-density digital magnetic recording systems, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan, MR74-39 (1975).
9. Nakano, Nagayama and Takanami. Required bandwidth of high-density recording heads, Natl. Conv. Record of I.E.C.E., Japan, 217 (1975).
10. J.C. Mallinson. Maximum signal-to-noise ratio of a tape recorder, IEEE Trans., MAG-5, p. 3 (1969).
11. Satake and Norihashi. Theory on erasing noise and a model of magnetic particle distribution in a tape, Papers of Technical Group on Magnetic Recording, I.E.C.E., Japan, MR74-23 (1974).
12. Miyawaki. Noise Analysis, Asakura Publishing (1969).

Submitted October 3, 1975

APPENDIX

Let the isolated read-out waveform be

$$e(t) = A \cdot \frac{a}{a^2 + t^2}$$

and the bit cell time be T. The read-out waveforms of 2F patterns in MFM recording systems are

$$f_{2F}(t) = \sum_{n=-\infty}^{\infty} (-1)^n e(t - nT)$$

Fourier transforming the above, we obtain

$$f_{2F}(t) = 2A\omega_0 \sum_{n=0}^{\infty} \exp\{-(2n+1)\omega_0 a\} \cos\{(2n+1)\omega_0 t\}$$

where

$$\omega_0 = \pi/T$$

Since contributions of the second- and higher-order terms are small, they are neglected

$$f_{2F}(t) = 2A\omega_0 \exp(-\omega_0 a)\cos\omega_0 t$$

In the case of 1F patterns, contributions of the third- and higher-order terms are similarly neglected,

$$f_{1F}(t) = A\omega_0\left\{\exp\left(-\frac{1}{2}\omega_0 a\right)\cos\frac{1}{2}\omega_0 t + \exp\left(-\frac{3}{2}\omega_0 a\right)\cos\frac{3}{2}\omega_0 t\right\}$$

Let

$$k = 2A\omega_0,\ \alpha = \exp\left(-\frac{1}{2}\omega_0 a\right)$$

Then, V_{2F} and V_{1F} become

$$V_{2F} = 2k\alpha^2$$
$$V_{1F} = k(\alpha+\alpha^3)$$

From these equations, we obtain

$$R = \frac{V_{2F}}{V_{1F}} = \frac{2\alpha}{1+\alpha^2}$$

$$\beta = \frac{\alpha^2}{1+\alpha^2} = \frac{1-\sqrt{1-R^2}}{2}$$

The read-out signal waveforms of {110} patterns are

$$f_{110}(t) = \frac{2}{\sqrt{3}}A\omega_0\left\{\exp\left(-\frac{2}{3}\omega_0 a\right)\sin\frac{2}{3}\omega_0 t - \exp\left(-\frac{4}{3}\omega_0 a\right)\sin\frac{4}{3}\omega_0 t\right\}$$

$$= \frac{1}{\sqrt{3}}k\left(\alpha^{4/3}\sin\frac{2}{3}\omega_0 t - \alpha^{8/3}\sin\frac{4}{3}\omega_0 t\right)$$

Hence,

$$V_{110} = \frac{2}{\sqrt{3}}k\alpha^{4/3}$$

$$\gamma = \alpha^{4/3} = \left(\frac{1-\sqrt{1-R^2}}{R}\right)^{4/3}$$

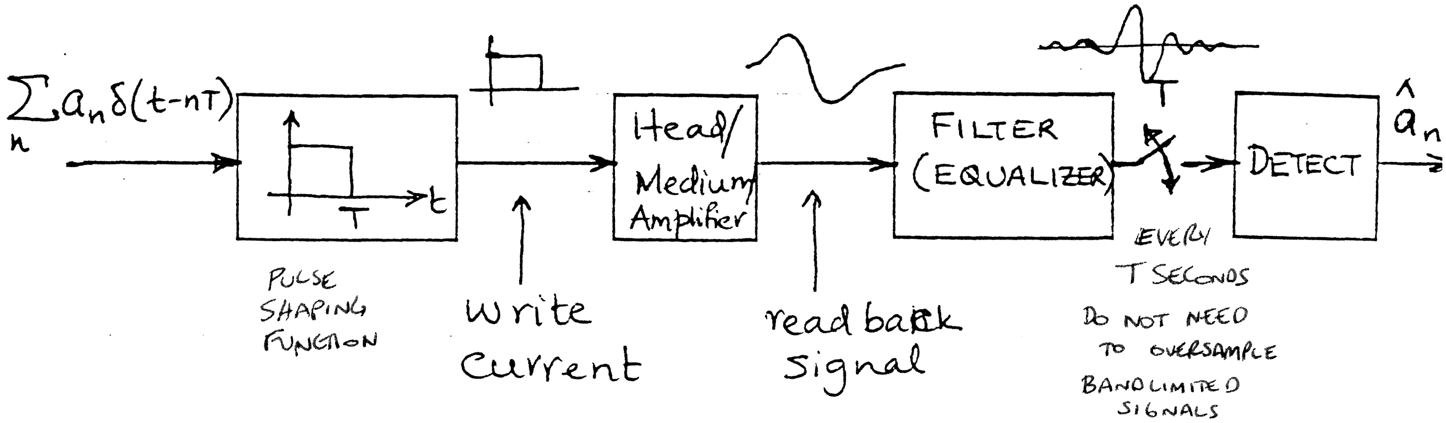
From the above, we have

$$\frac{V_{110}}{V_{2F}} = \frac{1}{\sqrt{3}\,\alpha^{2/3}} = \frac{1}{\sqrt{3\gamma}}$$

LECTURE 5 — EECS 458

I. DIGITAL (SAMPLING) DETECTION METHODS

AMPLITUDE IS IMPORTANT
DON'T CARE ABOUT PEAK SHIFT



Bandlimited channel

TO ½ DATA RATE (NYQUIST RATE)

$R = 1/T \qquad B = 1/2T$

- Any bandlimited signal can be represented as:

BY DUALITY

DELAYED BY KT

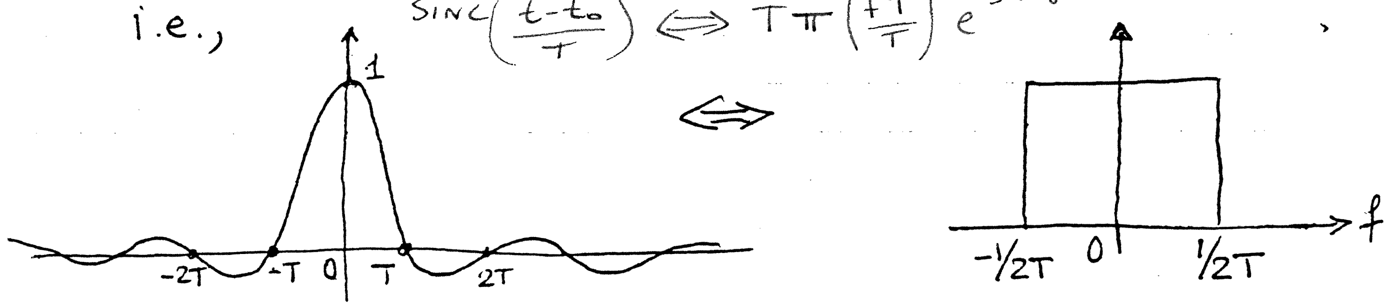
$$p(t) = \sum_{k=-\infty}^{\infty} p(kT)\, \mathrm{sinc}\left(\frac{t-kT}{T}\right)$$

now,

$$\mathrm{sinc}\,\frac{t}{T} \Longleftrightarrow T\, \Pi\left(\frac{fT}{T}\right)$$

$$\mathrm{sinc}\left(\frac{t-t_0}{T}\right) \Longleftrightarrow T\pi\left(\frac{fT}{T}\right) e^{-j\omega t_0}$$

i.e.,



DELAY

RECTANGULAR Π FUNCTION

$$\therefore \quad P(f) = \begin{cases} T \displaystyle\sum_{k=-\infty}^{\infty} p(kT)\, e^{-j\omega kT} & |f| \leq 1/2T \\ 0 & \text{otherwise} \end{cases}$$

Let $\quad D \triangleq e^{-j\omega T} = e^{-j2\pi fT}$ and dropping the T factor in the above equation, we can write $\qquad D^K = e^{-jwKT}$

$$P(f) = P(D)\Big|_{D=e^{-j2\pi fT}} = \underbrace{\sum_{k=-\infty}^{\infty} p(kT)\, D^{k}}_{\substack{\text{an algebraic equation} \\ \text{commonly used in the} \\ \text{literature to describe} \\ \text{bandlimited signals.}}} \qquad |f| \leq 1/2T$$

- By choosing different values for the coefficients $p(kT)$, we can construct different pulse shapes.

a) Let $p(kT) = \begin{cases} 1 & k=0 \\ -1 & k=1 \\ 0 & \text{otherwise} \end{cases}$

Then

$$p(t) = \operatorname{sinc}\left(\frac{t}{T}\right) - \operatorname{sinc}\left(\frac{t-T}{T}\right)$$

ALLOW TRANSITIONS TO OVERLAP IN ORDER TO LIMIT BANDWIDTH

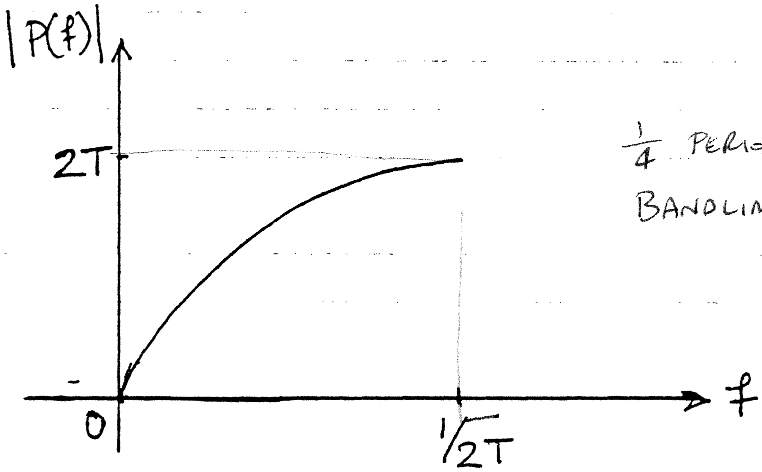
TRADE ISI FOR SPECTRAL RESPONSE

$$\Longleftrightarrow \quad P(f) = \frac{T}{\cancel{4}}\left[1 - e^{-j\omega T}\right]$$

$$= j2Te^{-j\frac{\omega T}{2}}\left[\frac{e^{j\frac{\omega T}{2}} - e^{-j\frac{\omega T}{2}}}{j2}\right]$$

$$= j2Te^{-j\frac{\omega T}{2}} \sin\left(\frac{\omega T}{2}\right)$$

$$= j2Te^{-j\frac{\omega T}{2}} \sin(\pi f T) \qquad |f| \leq 1/2T$$



$\frac{1}{4}$ PERIOD OF SIN

BANDLIMITED TO $\frac{1}{2T}$

(Note that the above pulse can be described by

$P(D) = 1 - D$)

b) Let $p(kT) = \begin{cases} 1 & k=0 \\ -1 & k=2 \\ 0 & \text{otherwise} \end{cases}$

Then

$$p(t) = \operatorname{sinc}\left(\frac{t}{T}\right) - \operatorname{sinc}\left(\frac{t-2T}{T}\right)$$

$$\Leftrightarrow \quad P(f) = T\left(1 - e^{-j2\omega T}\right)$$

$$= j2T e^{-j\omega T}\left[\frac{e^{j\omega T} - e^{-j\omega T}}{j2}\right]$$

$$= j2T e^{-j\omega T} \sin\omega T$$

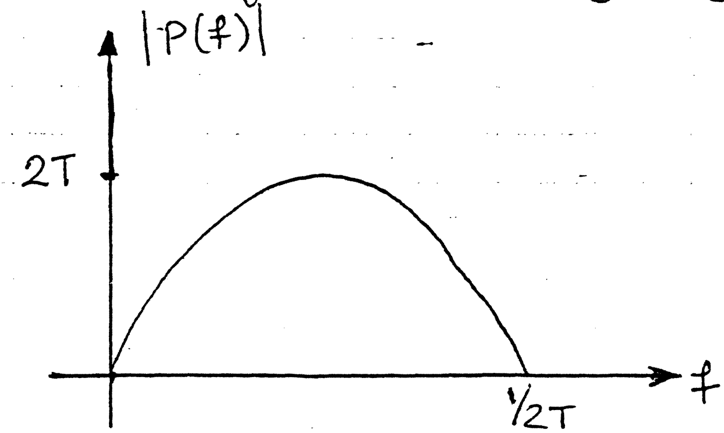
$$= j2T e^{-j\omega T} \sin(2\pi f T)$$

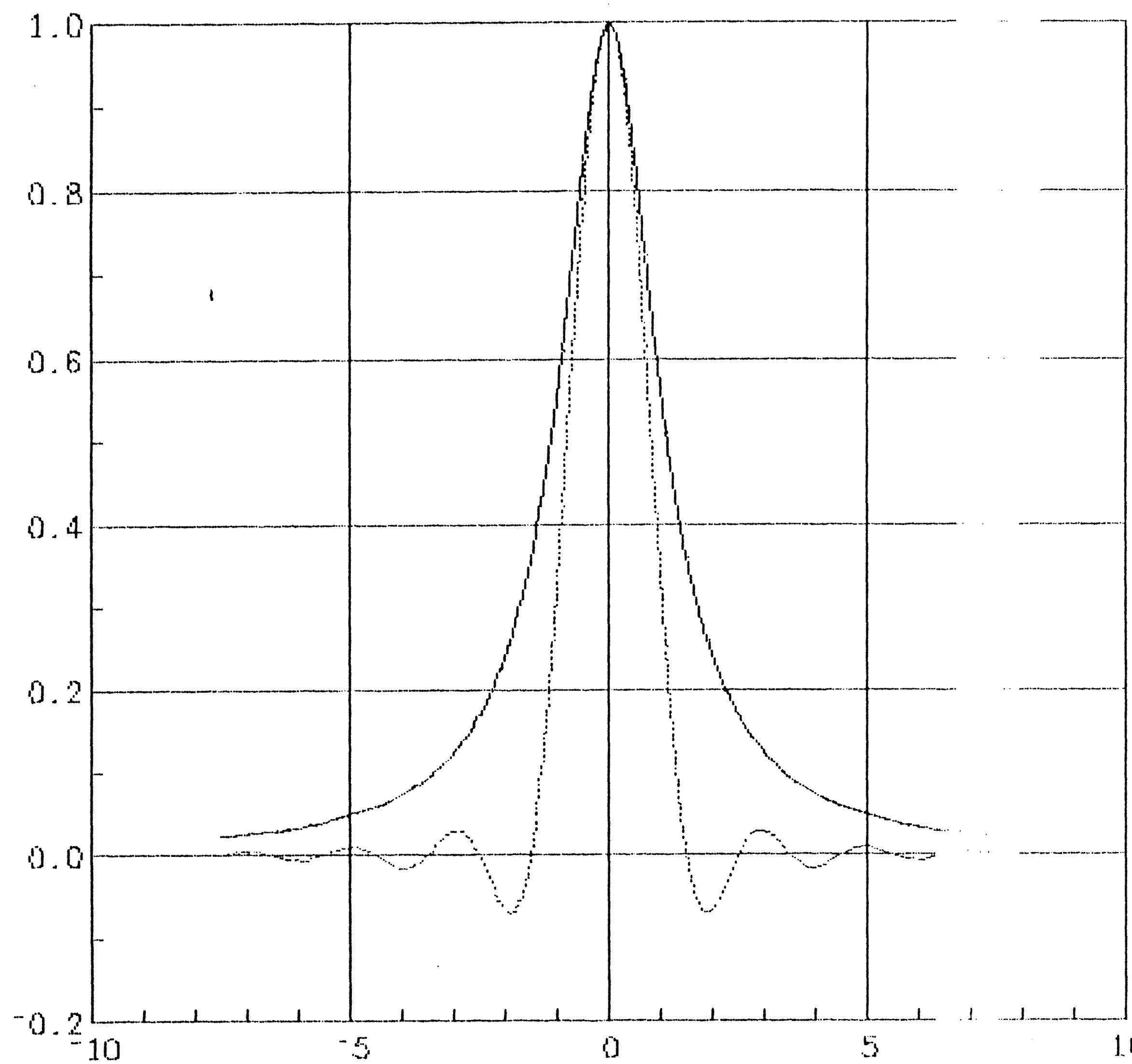
This pulse can be described by the polynomial

$$P(D) = 1 - D^2$$

This pulse is referred to as Class 4

PARTIAL RESPONSE signalling in the literature. It is also referred to as modified duobinary signalling.

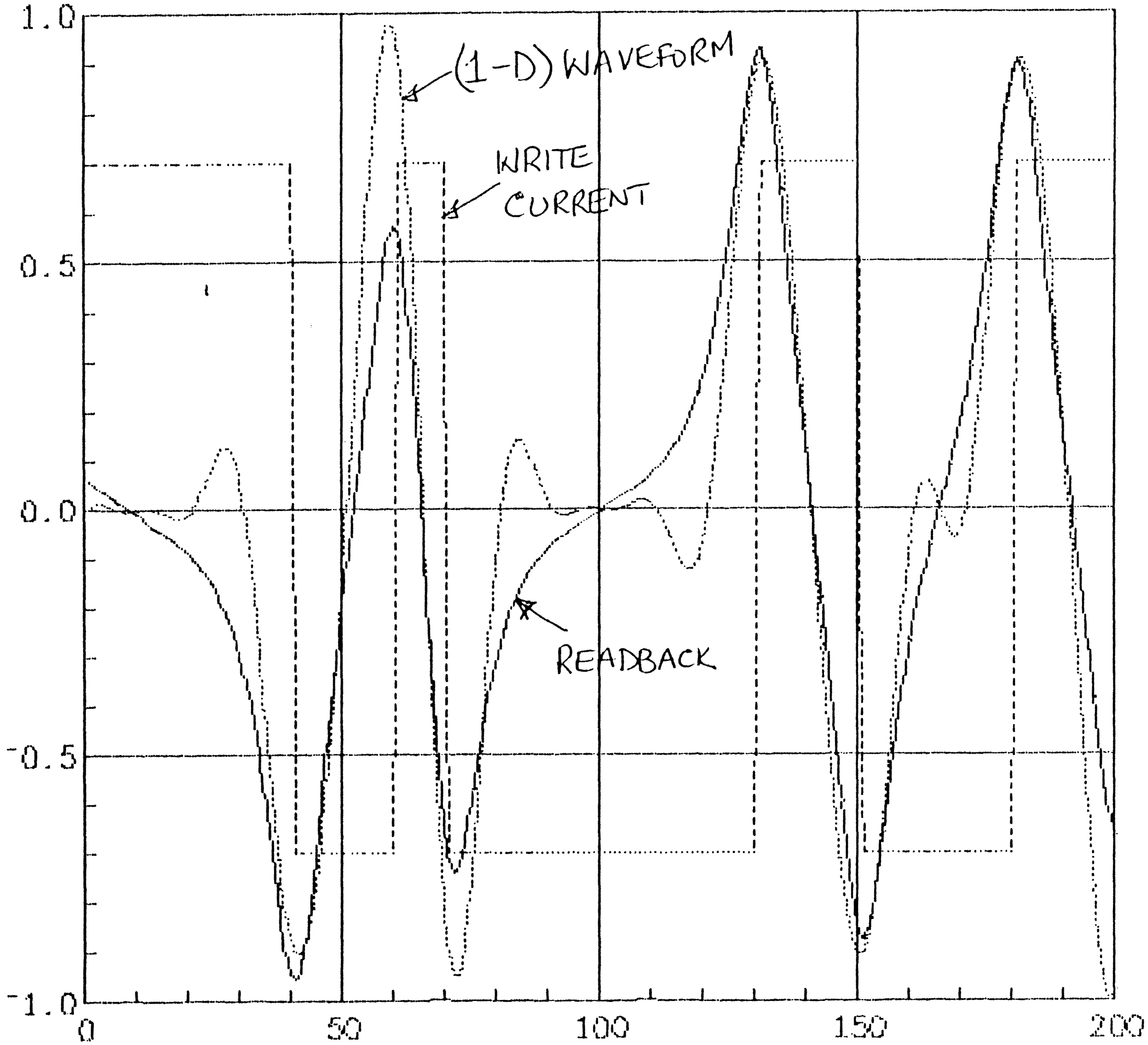




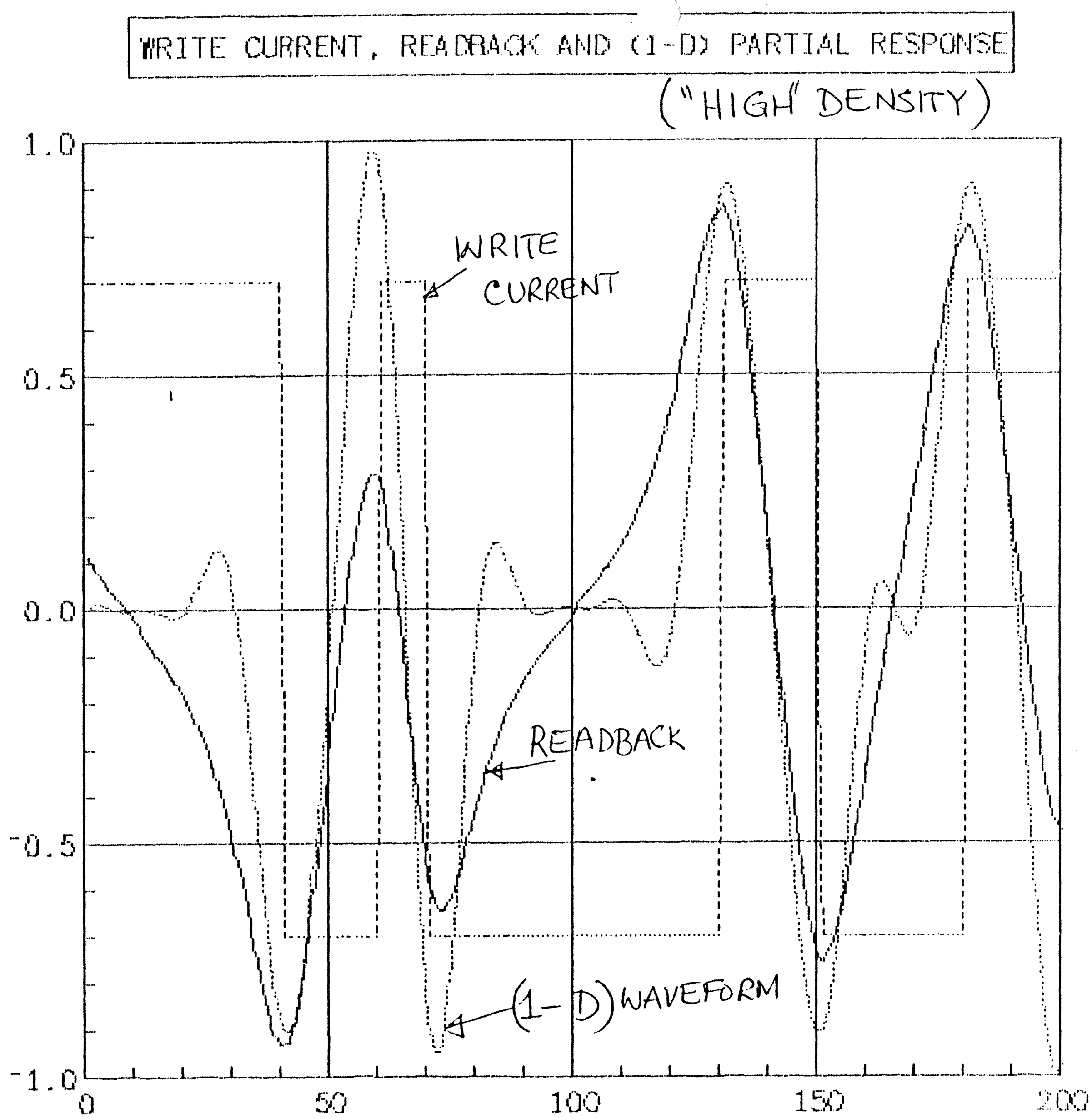
MODEL RECORDING CHANNEL TRANSITION RESPONSE AND MODIFIED DUOBINARY RESPONSE

WRITE CURRENT, READBACK AND (1-D) PARTIAL RESPONSE WAVEFORMS

("LOW" DENSITY)

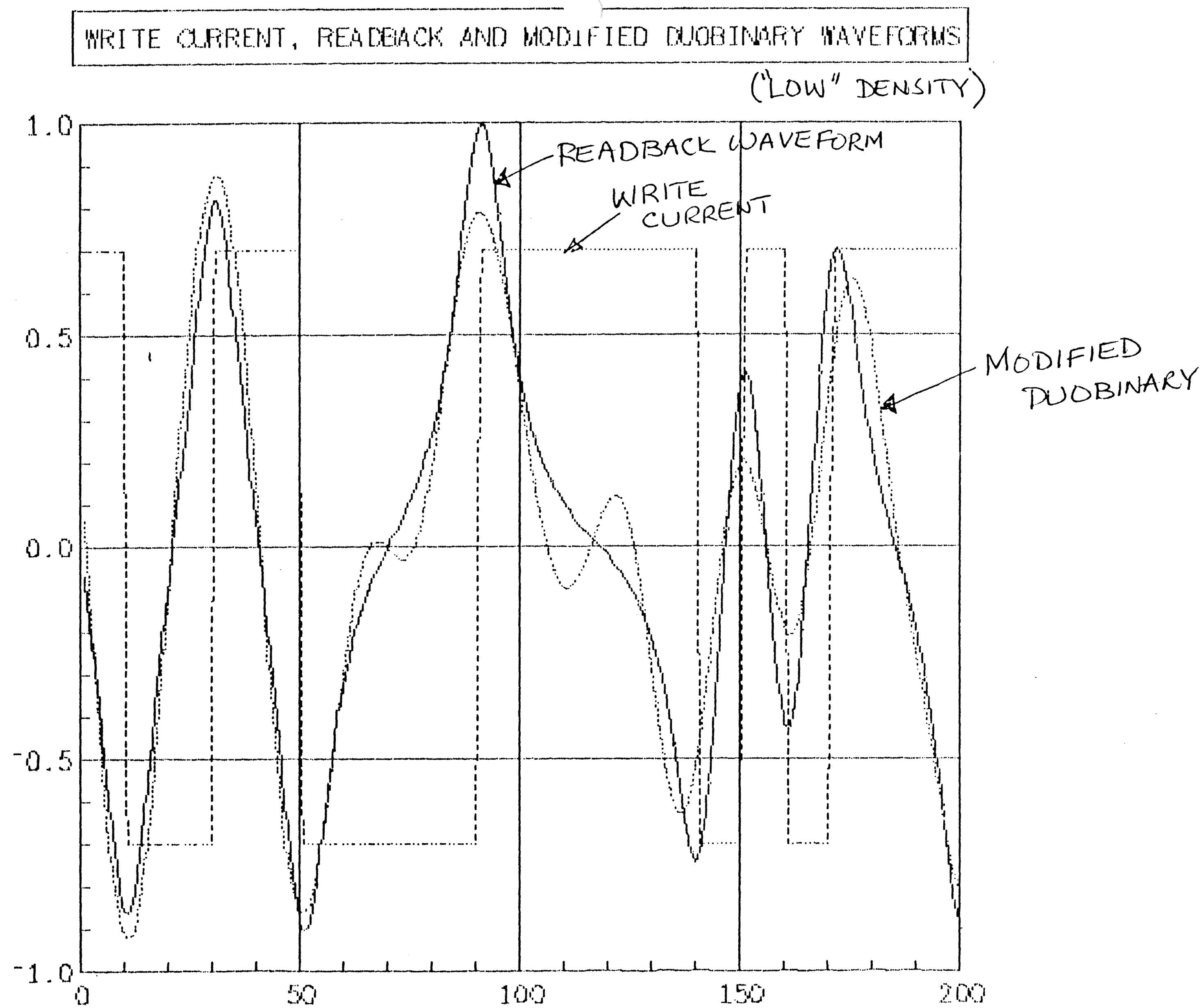


5.6



WRITE CURRENT, READBACK AND (1-D) PARTIAL RESPONSE
("HIGH" DENSITY)
1.0
0.5
0.0
-0.5
-1.0
0
50
100
150
200
WRITE CURRENT
READBACK
(1-D) WAVEFORM

5.7

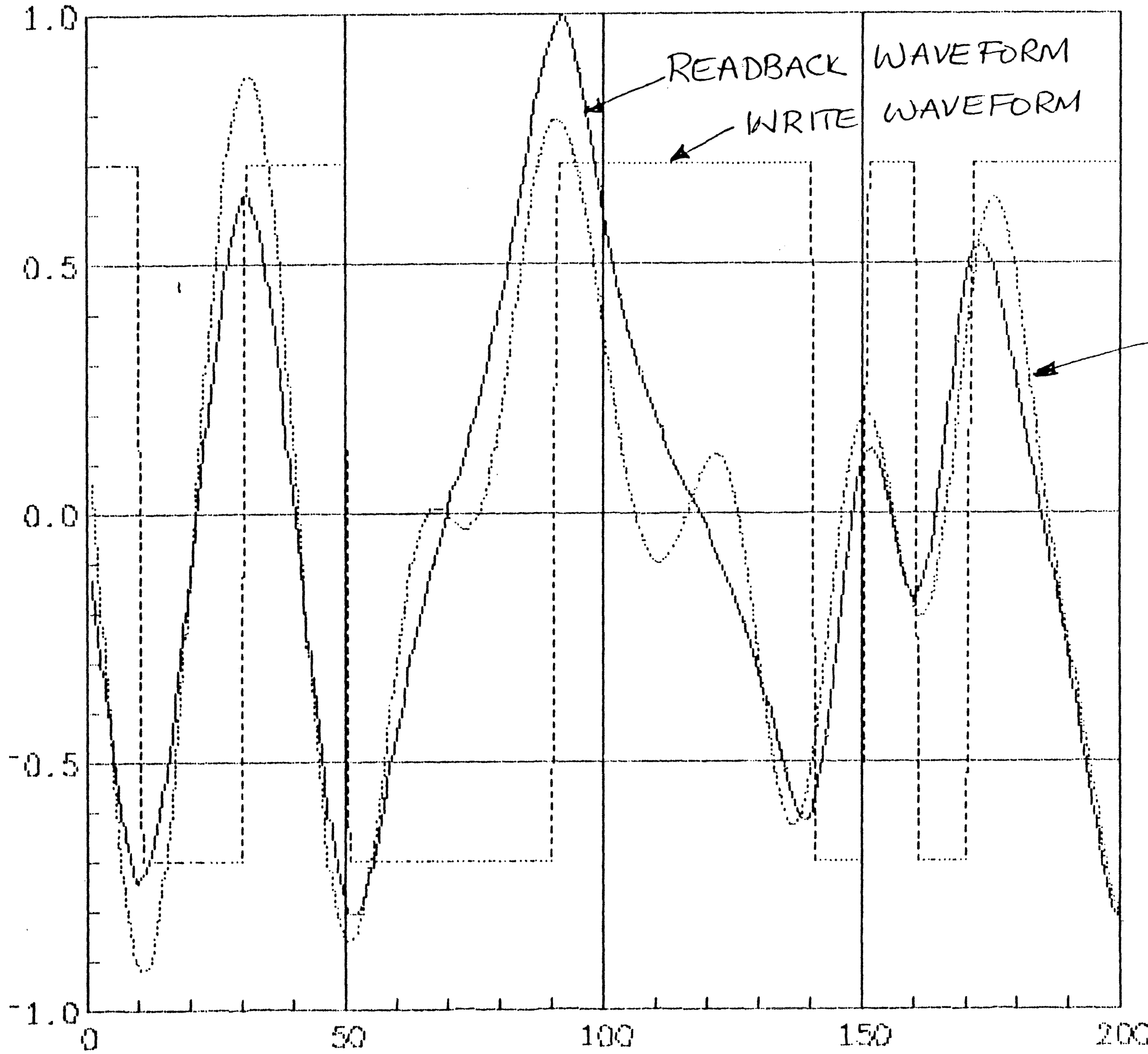


WRITE CURRENT, READBACK AND MODIFIED DUOBINARY WAVEFORMS
("LOW" DENSITY)
READBACK WAVEFORM
WRITE CURRENT
MODIFIED DUOBINARY
1.0
0.5
0.0
-0.5
-1.0
0
50
100
150
200

5.8

WRITE CURRENT, READBACK AND MODIFIED DUOBINARY WAVEFORMS

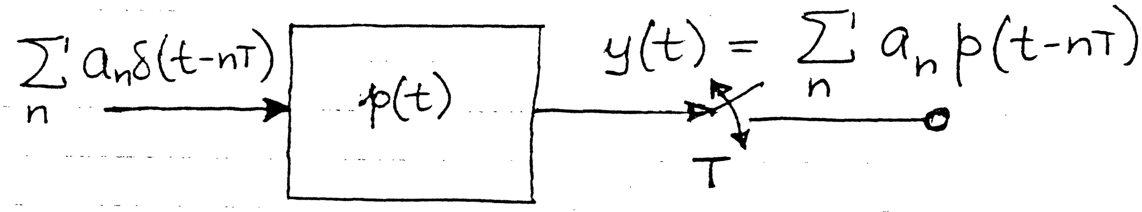
("HIGH" DENSITY)



NEED HIGH FREQUENCY BOOST TO SLIM PULSE WIDTHS

59

B. DETECTION WITH MODIFIED DUOBINARY SIGNALLING :



$$p(t) = \text{sinc}\left(\frac{t}{T}\right) - \text{sinc}\left(\frac{t-2T}{T}\right)$$

Note that $p(t)$ defines the overall channel impulse response after filtering.

Now, the filtered signal (or equalized signal) is

$$y(t) = \sum_n' a_n p(t-nT)$$

After the sampling process, the observed amplitude at $t = kT$ (where k is an integer)

$$y(kT) = \sum_n' a_n p(kT-nT)$$

$$= \sum_n' a_n p[(k-n)T]$$

But the modified duobinary pulse was constructed by letting

$p(0) = 1$, $p(2T) = -1$, and 0 at all other sample values. Thus, we get nonzero values only when $n = k$ and $(k-n) = 2$. (or $n = k-2$)

$$\therefore \quad y(kT) = \sum_n a_n \, p[(k-n)T]$$

$$= a_k - a_{k-2}$$

ERROR PROPAGATION

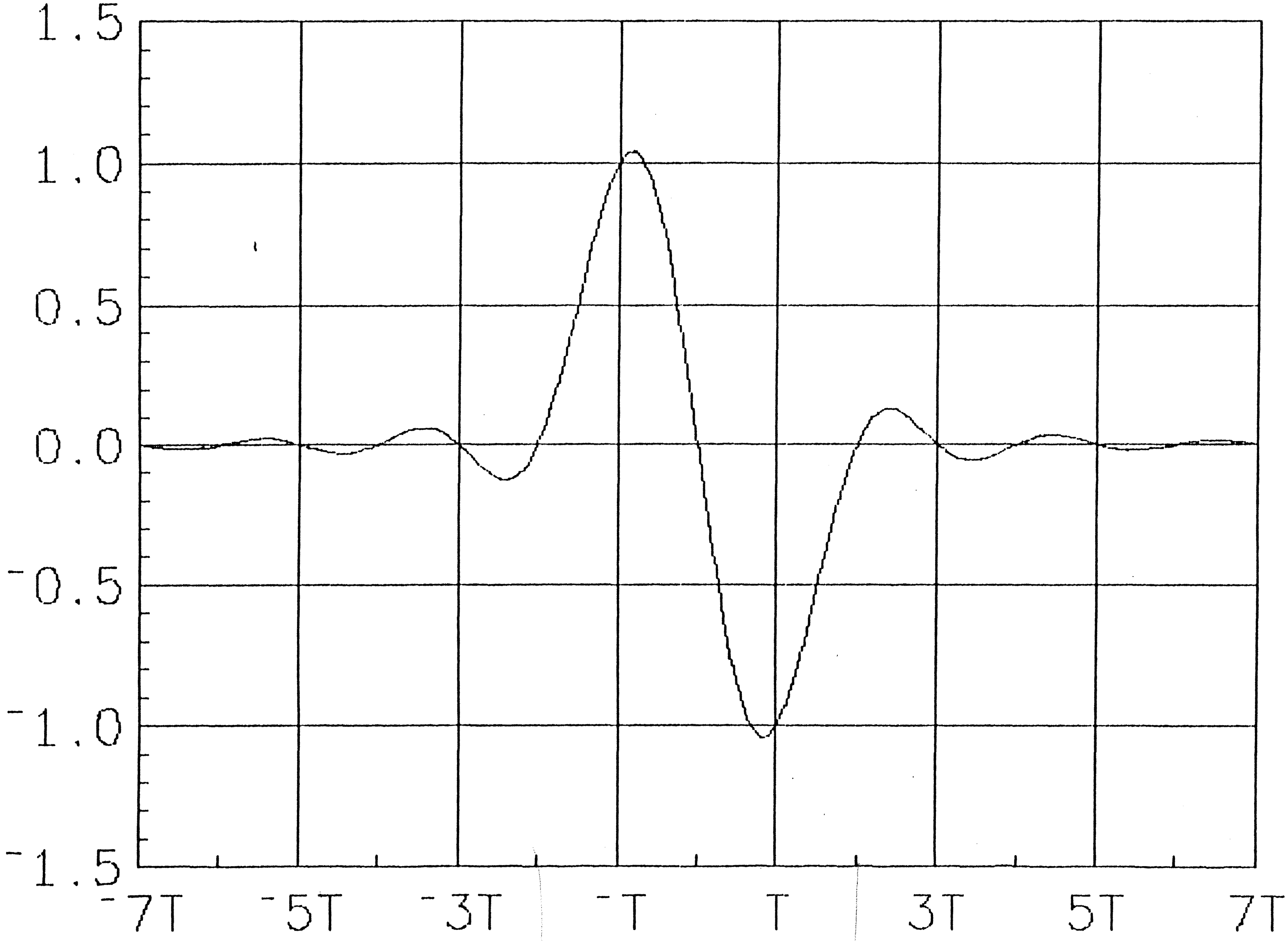
The above equation implies that the amplitude level at time kT is given by the difference between the input data at time k and $(k-2)$.

Thus,

$$y(kT) = \begin{cases} 2 & \text{if } a_k = 1 \text{ and } a_{k-2} = -1 \\ 0 & \text{if } a_k = \pm 1 \text{ and } a_{k-2} = \pm 1 \\ -2 & \text{if } a_k = -1 \text{ and } a_{k-2} = 1 \end{cases}$$

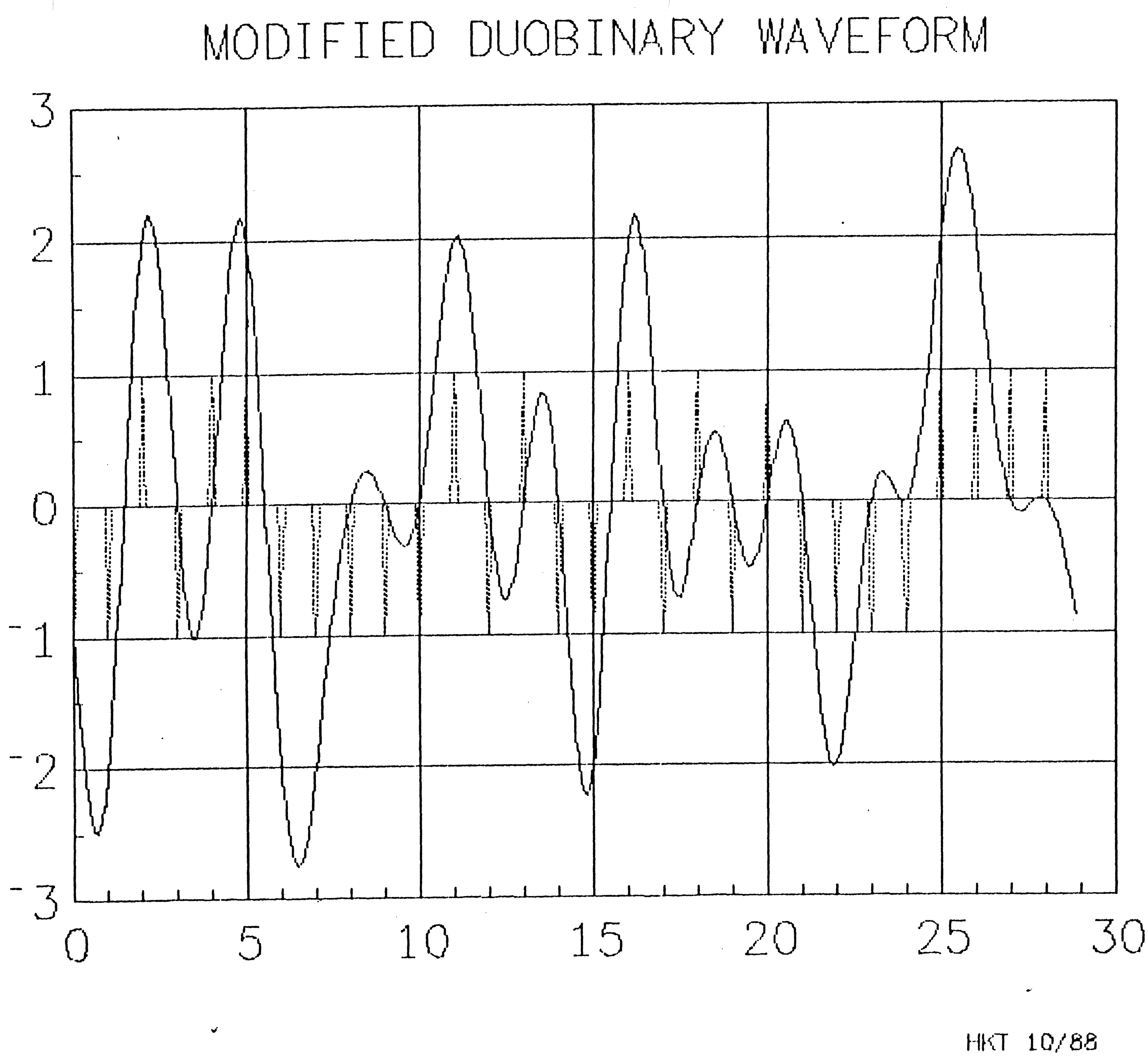
~~At~~ At the sampling instants, there are, therefore, 3 amplitude levels. Knowing a_{k-2} and $y(kT)$, one can detect a_k by using the above rules. $a_k = y(kT) + a_{k-2}$ Once a_k is known, one can detect a_{k+2} from the observed $y[(k+2)T]$. One can use a recursive relationship.

MODIFIED DUOBINARY PULSE



HK

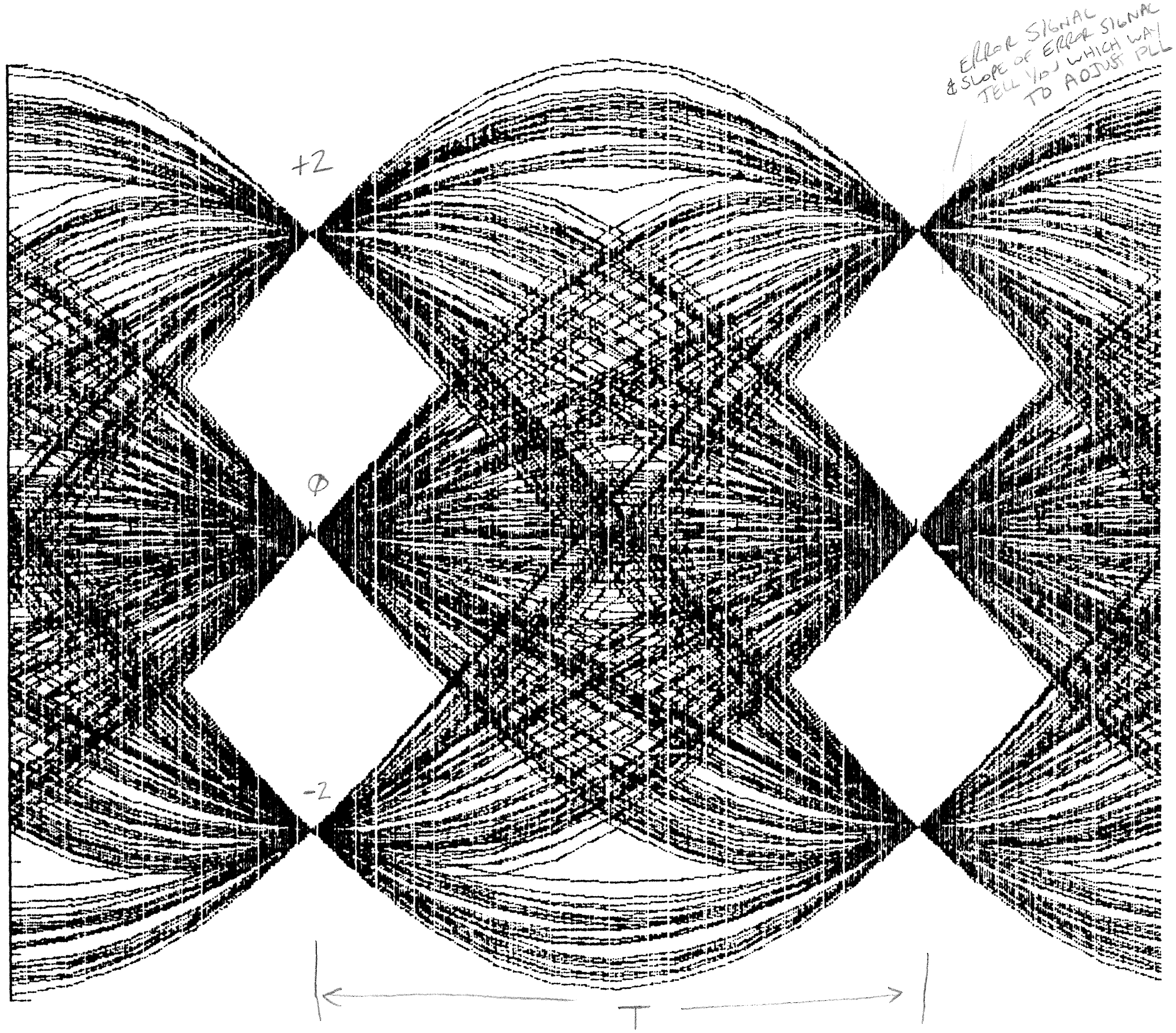
5.12



IBM SJ 221

5.13

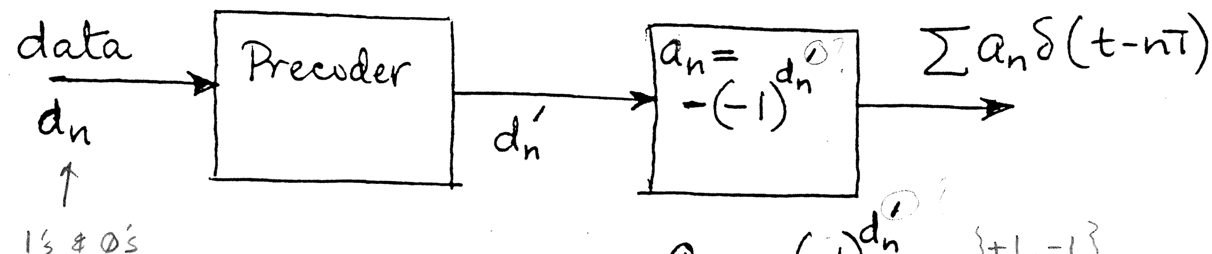
EYE PATTERN FOR (1-D*2) PARTIAL RESPONSE



3 DISTINCT LEVELS

5,14

- What if an error occurs in the detection of one of a_n's? Since a recursive relation is used, all subsequent detected bits will be in error until another error occurs in the detected value of $y(kT)$. To avoid error propagation, use <u>precoding</u> on the input data.

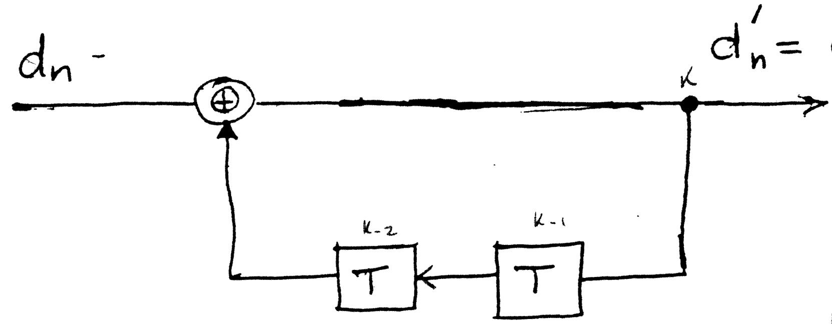


$$a_n = -(-1)^{d_n'} = \{+1, -1\}$$

PARTIAL RESPONSE

The precoder depends on the type bandlimited signal used. For modified duobinary, it is

$$d_n' = d_n \oplus d_{n-2}'$$



$Y_K = a_K - a_{K-2}$

$\boxed{-(-1)^{d_n'}}$

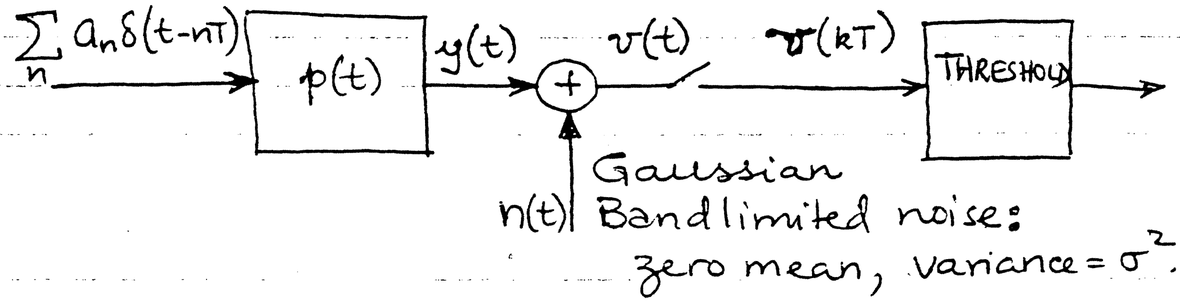
$= -(-1)^{d_n \oplus d_{n-2}}$

DECODER IS OPPOSITE

TRY INTRODUCING AN ERROR

$Y_K = -(-1)^{(a_K \oplus a_{K-2})} (a_K - a_{K-2})$

C. PERFORMANCE OF MODIFIED DUOBINARY WITH THRESHOLD DETECTION



In the absence of noise, $v(kT) = y(kT)$ will ideally have three values $\pm 2, 0$. Now,

$$v(t) = \underbrace{y(t)}_{\text{SIGNAL}} + \underbrace{n(t)}_{\text{NOISE}}$$

$1 - D^2$

a_n	a_{n-2}	$y(t)$
1	1	$1-1 = 0$
1	-1	$1-(-1) = 2$
-1	1	$-1-1 = -2$
-1	-1	$-1-(-1) = 0$

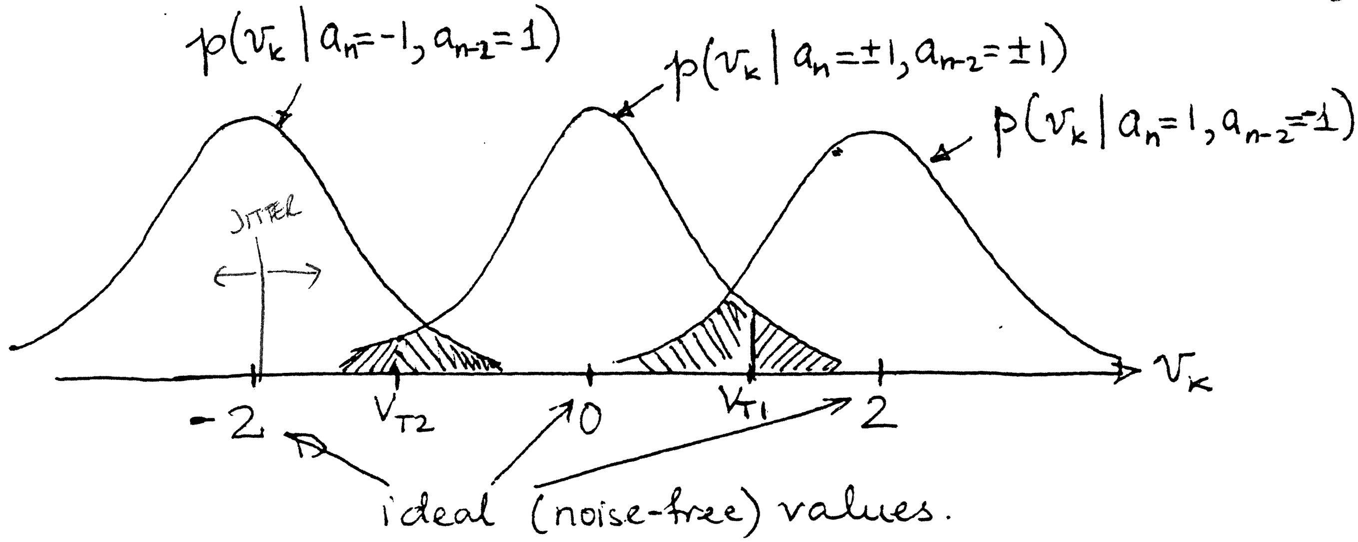
$$\text{and} \quad v(kT) = \begin{cases} 2 + n(kT) \\ 0 + n(kT) \\ -2 + n(kT) \end{cases}$$

Since $n(kT)$ is Gaussian, $v(kT)$ is also Gaussian. In fact, if $v_k \triangleq v(kT)$, then the probability density functions

$$p(v_k \mid a_n = 1, a_{n-2} = -1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(v_k-2)^2/2\sigma^2}$$

$$p(v_k \mid a_n = \pm 1, a_{n-2} = \pm 1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(v_k-0)^2/2\sigma^2},$$

$$p(v_k \mid a_n = -1, a_{n-2} = 1) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(v_k+2)^2/2\sigma^2}$$



$$P(E|A) = P(E \mid a_n=1, a_{n-2}=-1) = \int_{-\infty}^{V_{T1}} p(v_k \mid a_n=1, a_{n-2}=-1)\, dv_k$$

$$P(E|B) = P(E \mid a_n=-1, a_{n-2}=1) = \int_{V_{T2}}^{\infty} p(v_k \mid a_n=-1, a_{n-2}=1)\, dv_k$$

$$P(E|C) = P(E \mid a_n=\pm1, a_{n-2}=\pm1) = \int_{-\infty}^{V_{T2}} p(v_k \mid a_n=\pm1, a_{n-2}=\pm1)\, dv_k + \int_{V_{T1}}^{\infty} p(v_k \mid a_n=\pm1, a_{n-2}=\pm1)\, dv_k$$

$A \triangleq$ event $a_n=1,\ a_{n-2}=-1$

$B \triangleq$ event $a_n=-1,\ a_{n-2}=1$

$C \triangleq$ event $a_n=1,\ a_{n-2}=1$ or $a_n=-1,\ a_{n-2}=-1$.

$p(0)=1/2$
$p(1)=1/2$

	n	$n-2$	
A	1	0	$(\frac{1}{2})(\frac{1}{2}) = 1/4$
B	0	1	$(\frac{1}{2})(\frac{1}{2}) = 1/4$
C	0 / 1	0 / 1	$(\frac{1}{2})(\frac{1}{2})$, $(\frac{1}{2})(\frac{1}{2})$ $= 1/4 + 1/4 = 1/2$

If the input data is random; that is 0s and 1s are equally likely, then

$$P(A) = 1/4 \quad P(B) = 1/4 \quad P(C) = 1/2.$$

The average probability of error

TIMES A OCCURS

$$P(E) = P(A)P[E|A] + P(B)P[E|B] + P(C)P[E|C]$$

$\frac{1}{4}$ 1 $+$ $\frac{1}{4}$ 1 $+$ $\frac{1}{2}$ 2 $= \frac{3}{2}$

The threshold values V_{T1} and V_{T2} can be selected to minimize the average probability of error $P(E)$. Setting $\frac{dP(E)}{dV_{T1}} = 0$ and $\frac{dP(E)}{dV_{T2}} = 0$ yields,

$$V_{T1} = \frac{\sigma^2}{2} \ln 2 + 1$$

$$V_{T2} = \frac{\sigma^2}{2} \ln(1/2) - 1$$

Note that the optimum thresholds are not at ± 1 but are slightly biased. This is because the $P(C) > P(A)$ and $P(C) > P(B)$.
In practice, the noise variance is not known exactly. Furthermore, the data to be recorded may not be random. Because of these factors, the thresholds are generally set half-way between adjacent amplitude levels. Thus, in practice, V_{T1} will be set equal to 1, and $V_{T2} = -1$.

With $V_{T1} = 1$ and $V_{T2} = -1$, the average probability of error (assuming random data)

$$\boxed{P(E) = \frac{3}{2} Q\left(\frac{1}{\sigma}\right)}$$

To get $P(E) = 10^{-7} \Rightarrow \frac{1}{\sigma} \approx 5.2$ (see Tables for $Q(k)$)

$$\Rightarrow \sigma \leq \frac{1}{5.2}$$

The average power for the modified duobinary signal for the nominal amplitudes of $0, \pm 2$ is given by

$$E\left[\,|y_k^2(kT)|\,\right] = \overset{P(C)}{\frac{1}{2}}(0)^2 + \overset{P(A)}{\frac{1}{4}}(2)^2 + \overset{P(B)}{\frac{1}{4}}(-2)^2$$

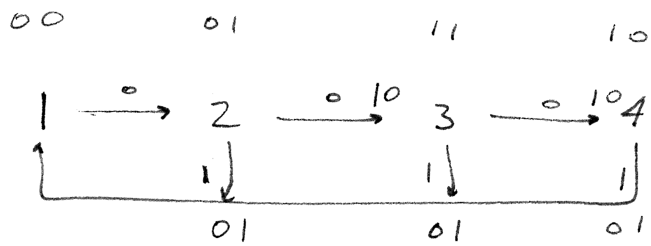
$$= 2$$

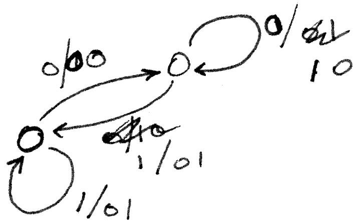
$\therefore$ the required $\left(\frac{\text{RMS ave. signal power}}{\text{noise power}}\right)$ for 10^{-7} error rate

$$SNR = \frac{E[y_k^2(kT)]}{\sigma^2} = \frac{2 \times (5.2)^2}{\cancel{(1/5.2)}} = 54.08$$

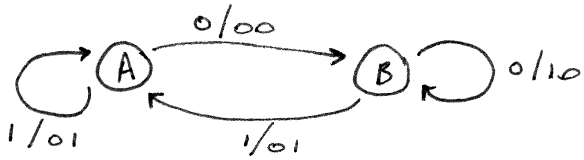
$$\Rightarrow 17.3 \text{ dB}.$$

This is the required signal-to-noise power at the input of the threshold detector.





0	0	0 0	1 0
0	1	0 0	0 1
1	0	0 1	0 0
1	1	0 1	0 1



0 0 0	1 0
0 0 1	1 0
0 1 0	0 1
0 1 1	0 1
1 0 0	0 0
1 0 1	0 0
1 1 0	0 1
1 1 1	0 1

DECODER	DATA
1 0	0
0 0	0
0 1	1

FM

ENCODER	DATA
0	1 0
1	1 1

DECODE - DROP CLOCK

5/11/89

Coding for Data Storage Channels
IIST, Spring Quarter 1989

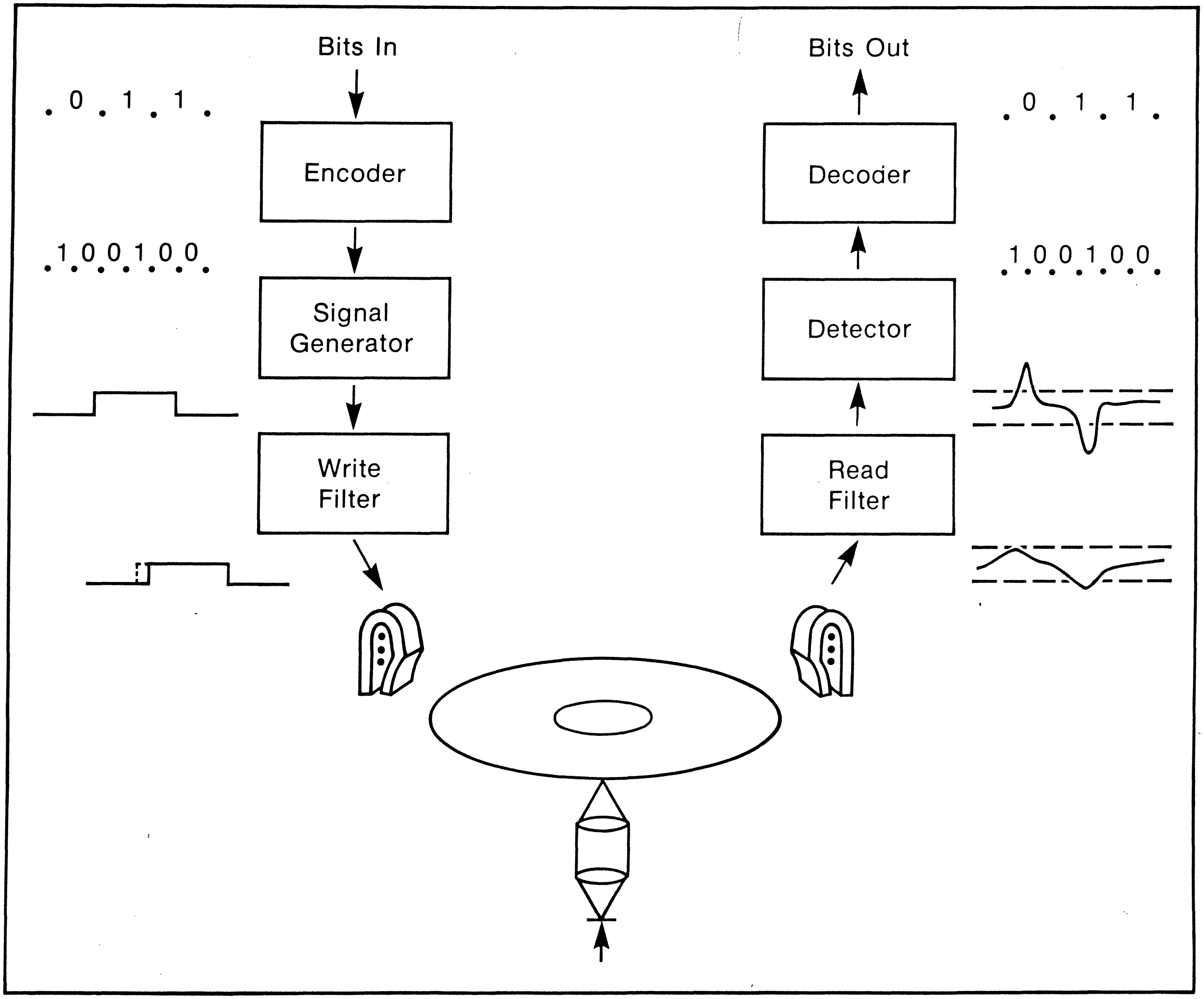
Paul H. Siegel

IBM Research Division
Almaden Research Center K65/802
San Jose, California 95120-6099
(408)927-2044

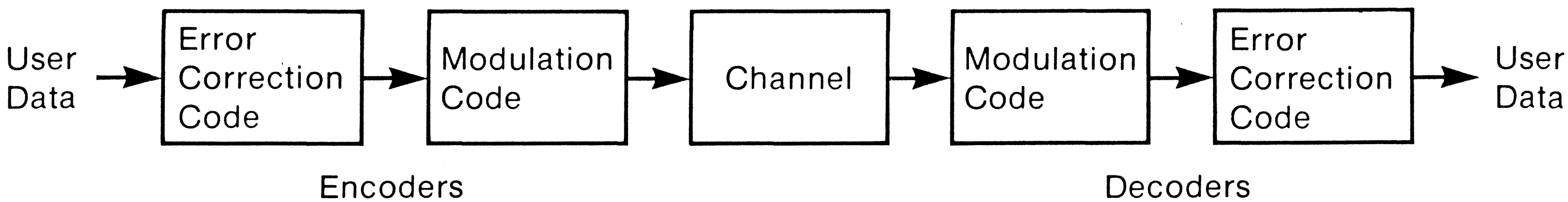
May 11, 1989

IBM

Digital Data Recording Schematic



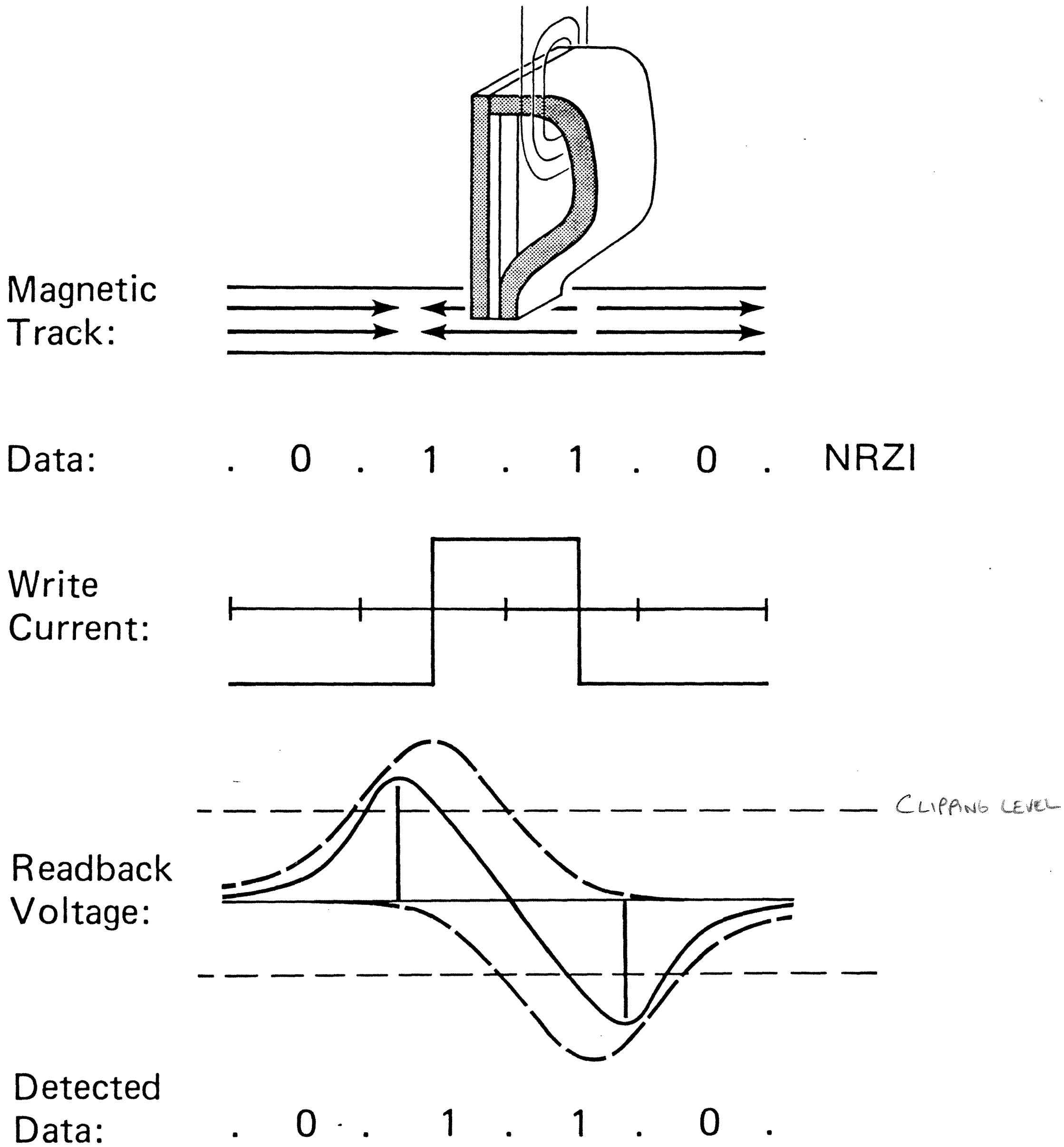
Configuration of Codes



Modulation Code: Matches recorded signal characteristics to channel bandwidth, detection method, read/write electronics, timing and tracking servo requirements

Error Correction Code: Detects and corrects data detection errors

Digital Magnetic Recording Channel

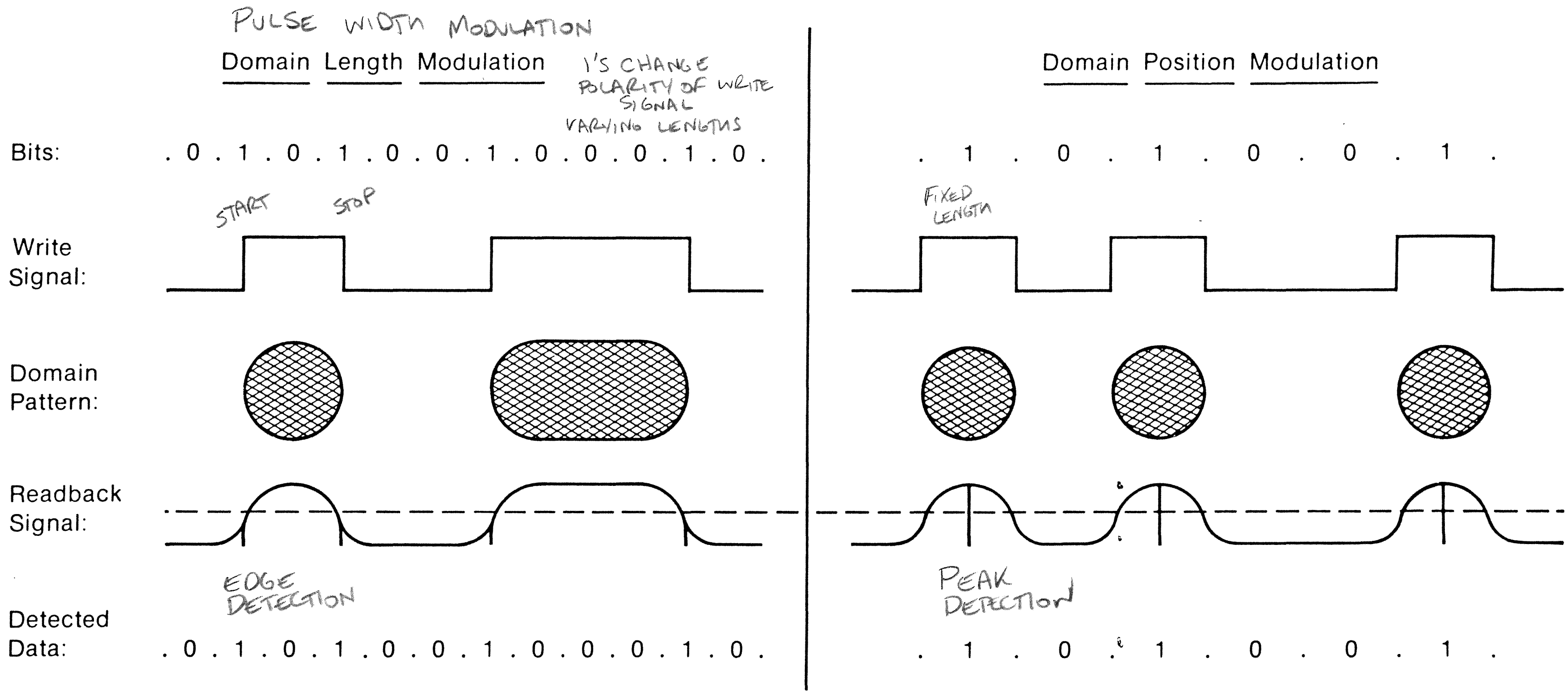


Causes of bit detection errors

- Random noise
- Intersymbol Interference
- Loss of clock synchronization

} Bit pattern related

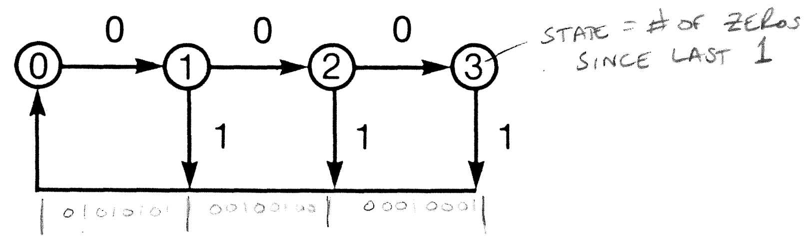
Digital Optical Recording Channel



Run-Length-Limited (RLL) Codes

- (d,k) constrained codes
  MINIMUM # 0's BETWEEN 1's
  — d $\leq$ # 0's between consecutive 1's $\leq$ k (ISI)
  — d controls intersymbol interference
  — k controls clock update information
  MAXIMUM # 0's BETWEEN 1's (CLOCK RECOVERY & GAIN CONTROL)
- Example (d,k) = (1,3) string:

  $1\ \underbrace{0}_{d}\ 1\ \underbrace{0\ 0\ 0}_{k}\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ \ldots$
- State diagram representation of (1,3) constraints



- Code rate = (user bits/code bits)

(d,k)	Maximum Rate,C	Practical Rate
(0,1)	0.6942	1/2, 2/3
(1,3)	0.5515	1/2
(2,7)	0.5174	1/2
(1,7)	0.6793	2/3

CONSTRAINT CAPACITY OF CONSTRAINT

- Rate determines detection window

DOES NOT COMPLETELY CHARACTERIZE THE CODE. MANY DIFFERENT CODES SATISFY THE (1,7) CONSTRAINT

Code Constraints and Capacities

Unconstrained input		Constrained output

$$\begin{array}{ccc} |\leftarrow m \rightarrow| & \qquad\qquad & |\leftarrow n \rightarrow| \end{array}$$

$$2^m \left\{ \begin{array}{ccc} 000000 & \longleftrightarrow & 001001001 \\ 000001 & \longleftrightarrow & 010001001 \\ 000010 & \longleftrightarrow & 010010010 \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ 111111 & \longleftrightarrow & 100000001 \end{array} \right\} 2^{Cn}$$

- Shannon capacity $= C \leq 1$
- Necessary condition for block code:

$$2^m \leq 2^{Cn}$$

$$\frac{m}{n} \leq C$$

- Sufficient condition for block code if n <u>large</u>:

$$\frac{m}{n} < C$$

(2,7)

- Example: (d,k) constraints

  $C = \log_2 \lambda$, where λ is largest real root of $p(x) = x^{k+1} - x^{k-d} - \ldots - x - 1$

Examples of RLL Codes

Magnetic Disk	Magnetic Tape	Optical Disk
1/2 (0,1) FM, Double Frequency, Manchester, Biphase	1/2 (0,1) Phase Encoding 4/5 (0,2) Group Code Recording (GCR) 8/9 (0,3)	
1/2 (1,3) Modified FM (MFM) Delay Modulation, Miller 2/3 (1,7) (Jacoby)	$1/2^-$ (1,3) Zero Modulation (ZM) 1/2 (1,5) Miller^2	1/2 (1,3) Delay Modulation 2/3 (1,7) (Horiguchi-Morita)
1/2 (2,7) (Franaszek) 1/2 (2,11) 3PM		8/17 (2,10) EFM

Comparison of RLL (d,k) Constraints

CLOSEST SPACING OF 1's — SIZE OF DETECTION WINDOW

Data: . 1 . 1 . 0 . 1 .

FM 1/2(0,1) . 1 . 1 . 1 . 1 . 0 . 1 . 1 . 1 . — PAD AN ADDITIONAL 1 AFTER EACH DATA BIT

(D C D C D C D C)

+100% DENSITY

MFM 1/2(1,3) . 1 . 0 . 1 . 0 . 0 . 0 . 1 . 0 . — PUT A ZERO IN BETWEEN CONSECUTIVE ONES

+50% DENSITY

1/2(2,7): . 1 . 0 . 0 . 1 . 0 . 0 . 0 . 0 .

ISI INCREASED BY 11% — 33% WIDER DETECTION WINDOW

2/3(1,7): . 1 . 0 . 1 . 0 . 0 . 0 .

965 - IBM - 04

(0,1) → (1,3) → (2,7) { Less pulse crowding

(0,1) → (1,3) → (1,7) { Same/larger window

(2,7) vs (1,7) Involves tradeoffs

1,7 ISI INCREASE HF PATTERN STORES 12 DATA BITS IN 9 FLUX TRANSITIONS

2,7 " " 8 FLUX TRANSITIONS

1,7 HAS 1/9 = 11% MORE ISI

(2,7) vs MFM (1,3)

Fixed User Bit Density

MFM . 1 . 0 . 1 . $T_{min} = 2$

(2,7) . 1 . 0 . 0 . 1 . $T_{min} = 3$

(2,7) reduces intersymbol interference

Fixed Minimum Transition Spacing T_{min}

MFM . 1 . 0 . 1 .

2 BITS

Density Ratio
$2 \times 1/2 = 1$ bit/T_{min}

3 BITS

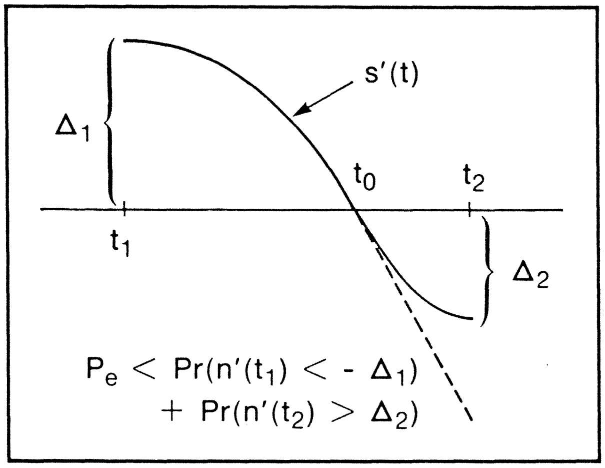
(2,7) . 1 . 0 . 0 . 1 .

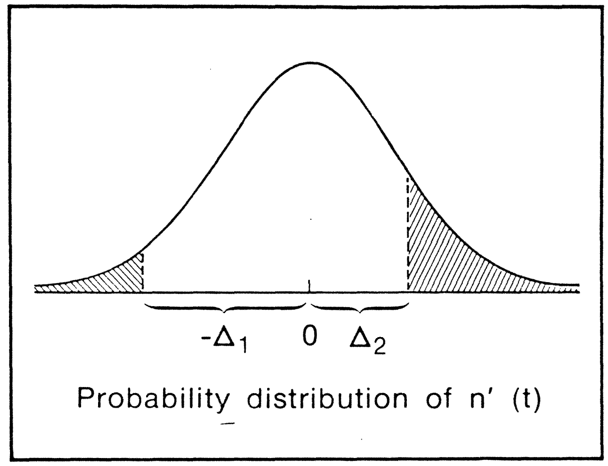
Density Ratio
$3 \times 1/2 = 1.5$ bit/T_{min}

(2,7) increases density ratio by 50%
with decreased detection window

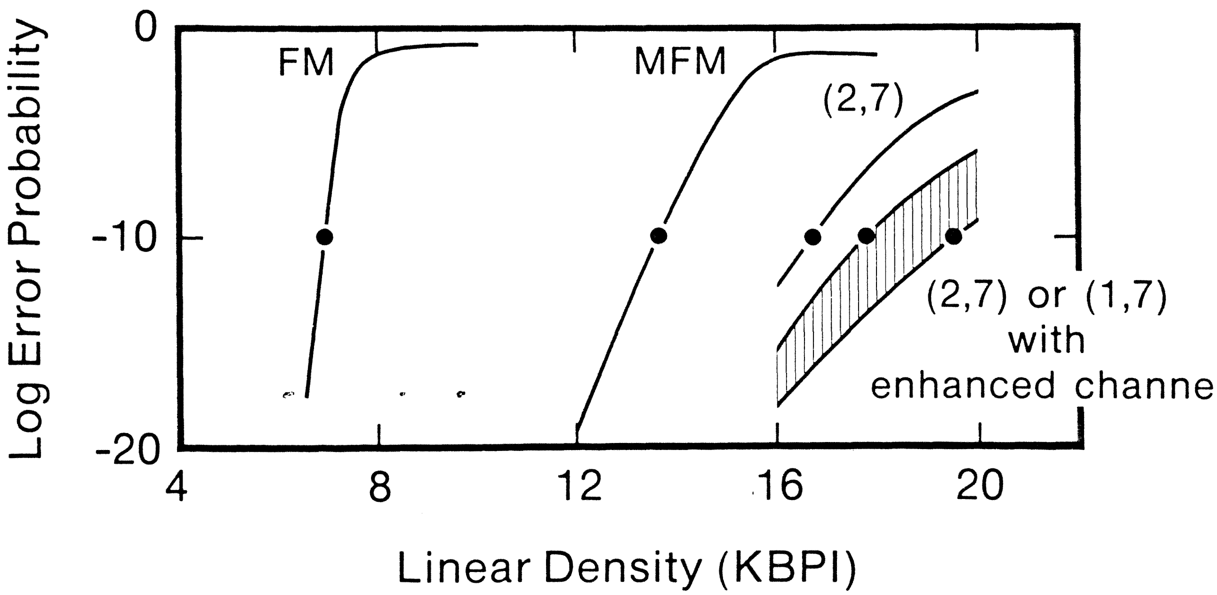
Peak Detection Channel Model

- Peak shift error rate calculation



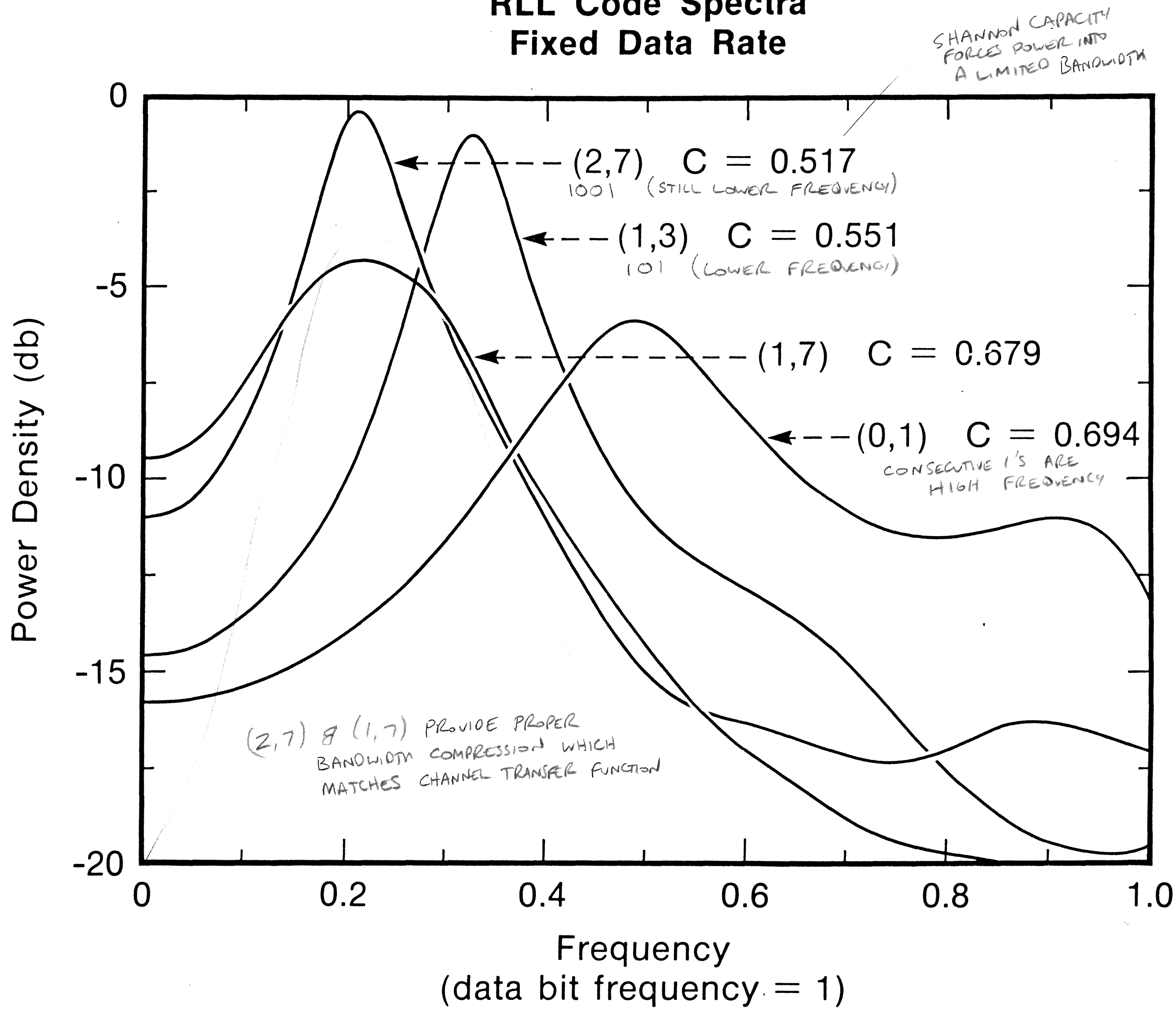


- RLL code performance



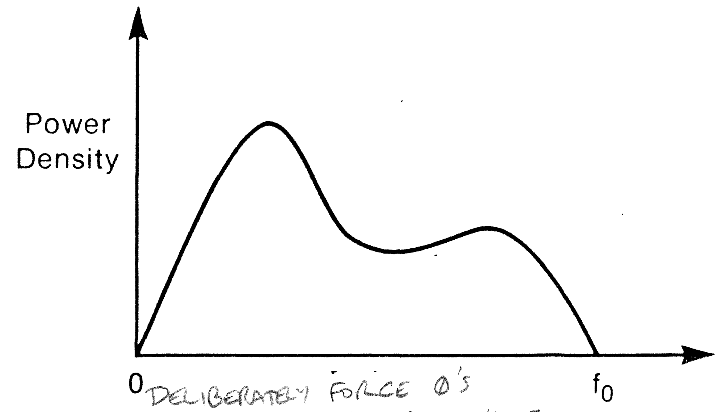
- Factor of 2.5 in linear density attributable to RLL code progress

RLL Code Spectra
Fixed Data Rate



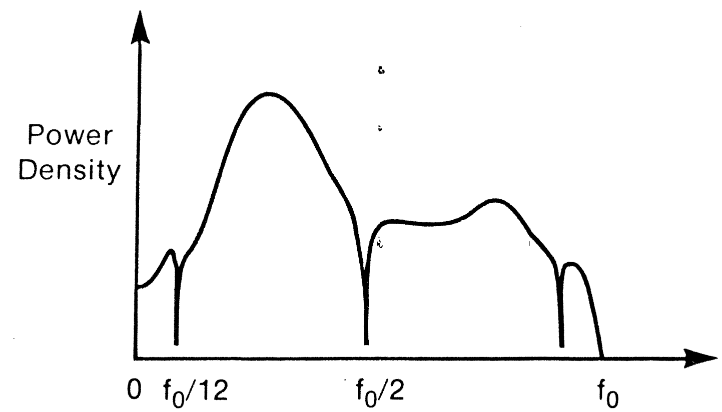
Spectral Null Codes

- Null at DC (f=0) DC FREE
  - AC-coupled read/write electronics, rotary head (HELICAL SCAN)
  - Low frequency noise suppression
  - Tracking and focus servo on pregrooved disk



DELIBERATELY FORCE 0's AT PARTICULAR FREQUENCIES

- Nulls at f≠0
  - Embedded tracking servo (magnetic disk)
  - Embedded timing servo (optical disk)

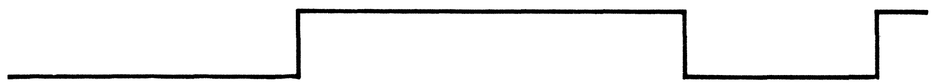


DC Null Codes

- Running Digital Sum (RDS): "accumulated charge"

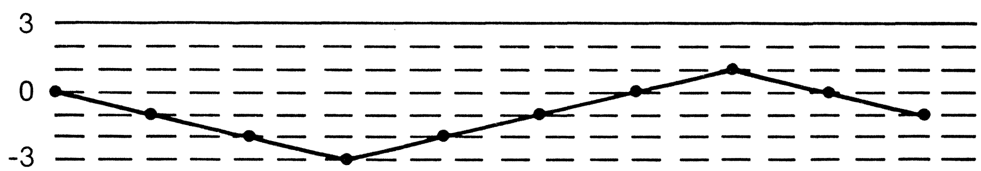
Bits . 0 . 0 . 1 . 0 . 0 . 0 . 1 . 0 . 1 .

Write Signal



Levels $\{a_i\}$. -1 . -1 . -1 . +1 . +1 . +1 . +1 . -1 . -1 .

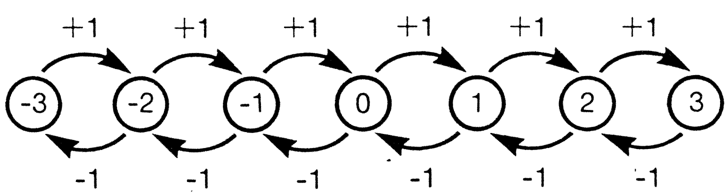
RDS $\sum_{i=1}^{N} a_i$



- Bounded RDS $\Longrightarrow$ DC null

$$\left| \sum_{i=1}^{N} a_i \right| \leq c, \text{ for all } N \geq 1, \text{ all } \{a_i\}$$

- State diagram for bounded RDS signals



Maximum rate: $\log_2 2 \cos \dfrac{\pi}{2c + 2}$

- RLL combined with RDS: (d,k;c) constraint

Codes with Null at f≠0

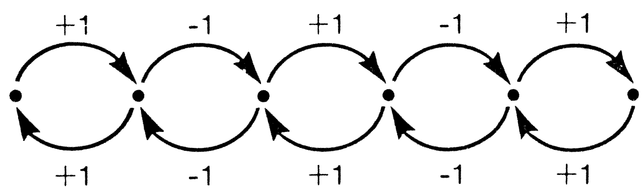
- Generalized RDS at $f=kf_0/n$

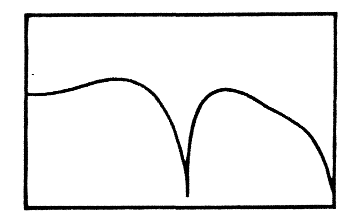
$$RDS_f(N) = \sum_{i=0}^{N} a_i e^{-j2\pi ki/n} = DFT\ (a_0,\ldots,a_N)$$

Spectral null at $f \Longleftrightarrow |RDS_f(N)| \leq c$, for $N \geq 0$

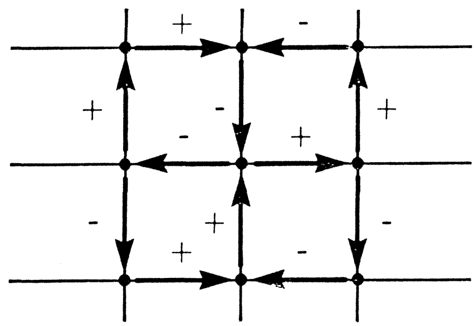
- State diagrams: examples

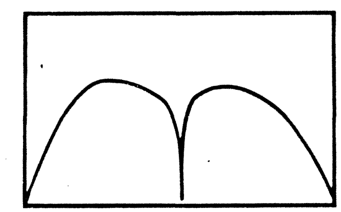
$f=f_0/2$



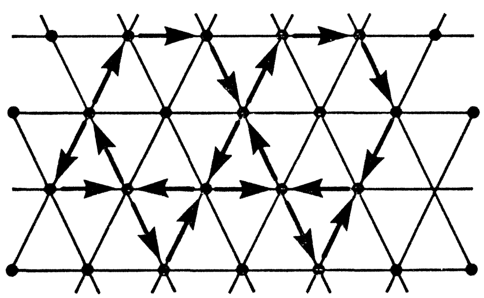


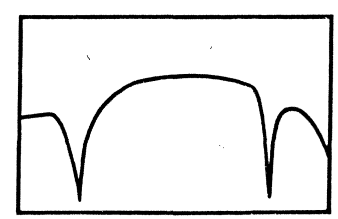
f=DC
and
$f=f_0/2$
(Interleaved DC)





$f=f_0/6$



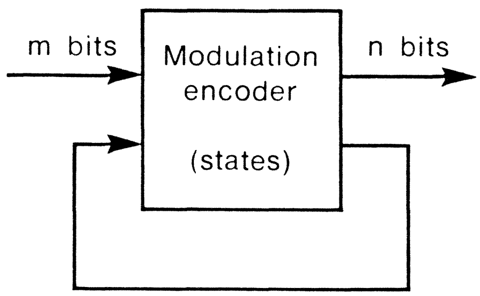


Code Construction

Techniques	Examples	
Block codes	4/5 (0,2)	GCR
	8/9 (0,3)	
	1/2 (0,1;1)	FM
Sequence state codes (fixed and variable length)	1/2 (1,3)	MFM
	1/2 (2,7)	(Franaszek)
	2/3 (1,7)	(Horiguchi-Morita)
	1/2 (1,5;3)	Miller^2
Look ahead codes	1/2 (2,11)	3PM
	2/3 (1,7)	(Jacoby)
	1/2 (1,3;3)	ZM
	8/17 (2,10;c)	EFM
Sliding block codes	1/2 (2,7)	(Adler-Coppersmith-Hassner)
	2/3 (1,7)	

Practical Code Implementation

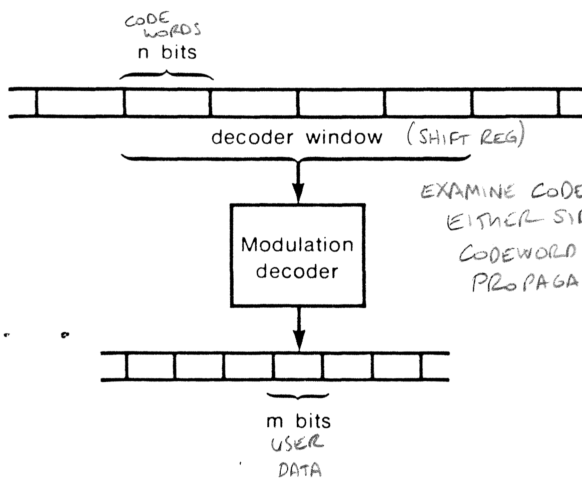
Encoder Finite State Machine



Features

- High rate m/n
- Low complexity

Decoder Sliding-Block Decoder



EXAMINE CODEWORDS ON EITHER SIDE OF CURRENT CODEWORD, LIMITS ERROR PROPAGATION

Features

- Limited error propagation
- Low complexity

Block Coding

FM (Frequency Modulation)
1/2 (0,1;1)

Encoder: Insert redundant code bit "1" between consecutive data bits (clock synchronization and dc-balancing)

Data	Code
0	10
1	11

Decoder: Drop redundant code bits

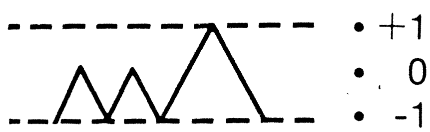
Example:

Data: . 1. 1. 0. 0.

FM Coded: .11.11.10.10.

Signal

RDS



Sequence State Coding
(fixed length)

MFM (Modified Frequency Modulation)

1/2(1,3)

Encoder:

A. After data bit = "0":

$0 \rightarrow \underline{1}0$
$1 \rightarrow \underline{0}1$

B. After data bit = "1":

$0 \rightarrow \underline{0}0$
$1 \rightarrow \underline{0}1$

State / Data	A	B
0	$\underline{1}0$/A	$\underline{0}0$/A
1	$\underline{0}1$/B	$\underline{0}1$/B

Decoder: Drop redundant bits

<u>Example:</u>

Data: . 1. 1. 0. 0.

MFM Coded: $.\underline{0}1.\underline{0}1.\underline{0}0.\underline{1}0$

MFM
(Modified Frequency Modulation)

1/2(1,3)

Encoder:

A. After data bit = "0":

0 → 10
1 → 01

B. After data bit = "1":

0 → 00
1 → 01

Data \ State	A	B
0	10/A	00/A
1	01/B	01/B

Decoder: Drop redundant bits

Example:

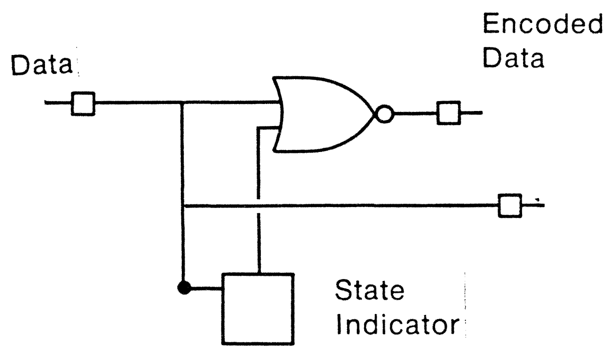
Data: . 1. 1. 0. 0.

MFM Coded: .01.01.00.10

MFM 1/2(1,3)

- 91% efficient   (0.5/0.5515 ≈ 0.91)
- Finite-state fixed-length encoder (1 bit → 2 bits)

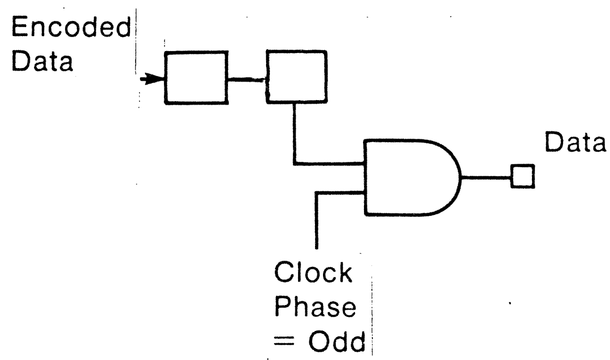
Data \ State	A	B
0	10/A	00/A
1	01/B	01/B



State A: Previous input = "0"
State B: Previous input = "1"

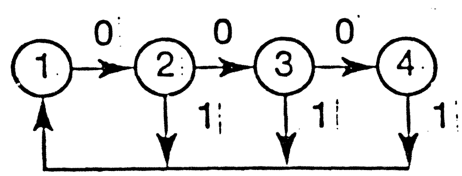
- Sliding block decoder

Code	Data
10	0
00	0
01	1



- Error propagation ≤ 1 data bit

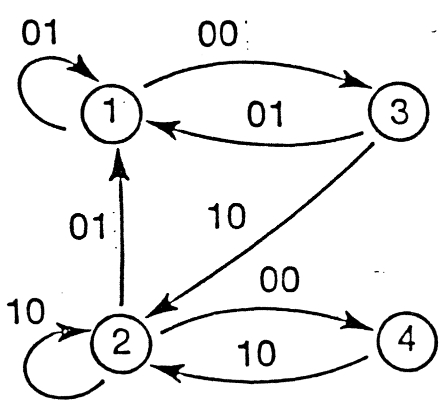
Construction of 1/2 (1,3) Code



State diagram G for (1,3)

$$T = \begin{matrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{matrix}$$

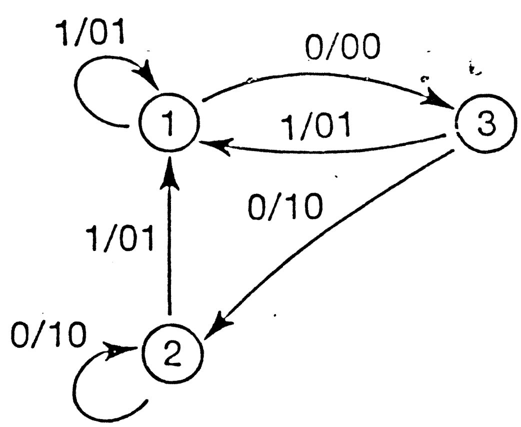
State-transition matrix T



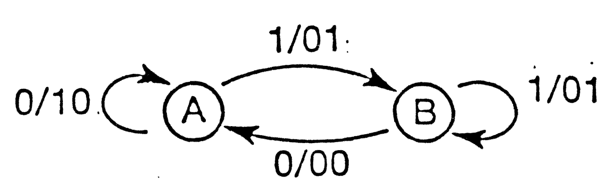
G 2

$$T^2 = \begin{matrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{matrix}$$

T 2



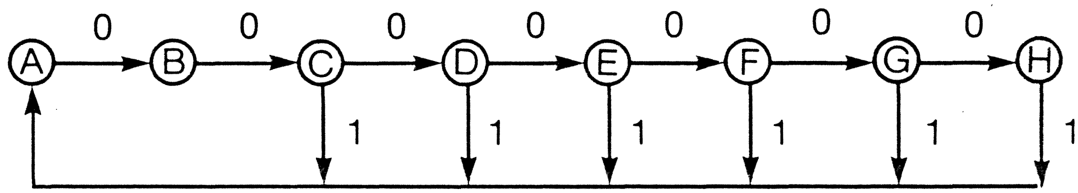
Encoder graph



Simplified (MFM)

Sequence State Coding
(variable length)

- 1/2(2,7) (Franaszek)
- Graph representation of (2,7) strings



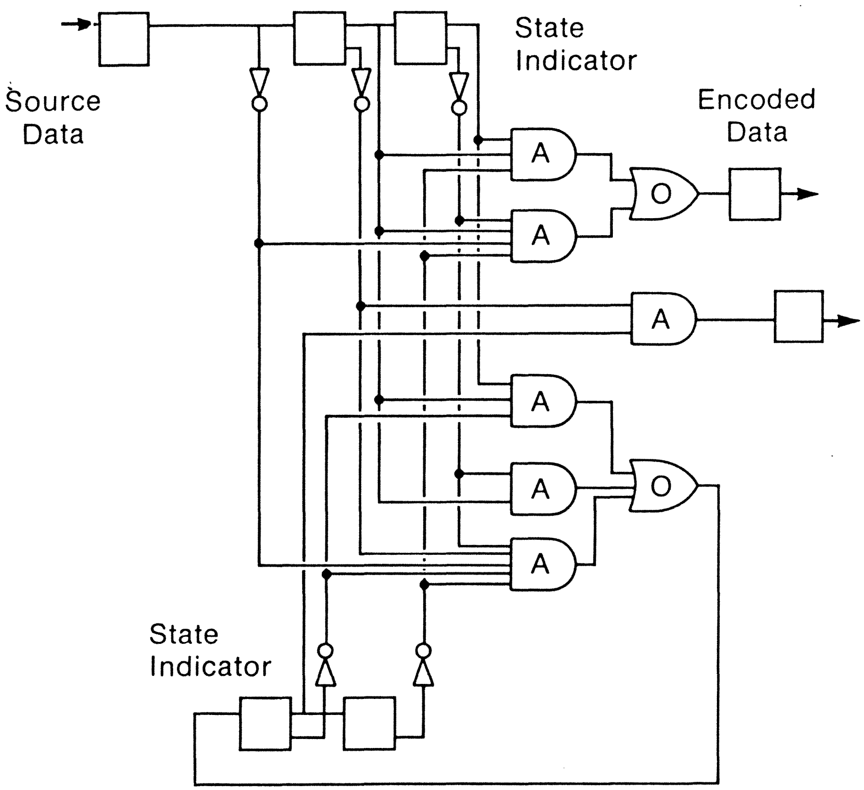
- Finite-state encoder based on graph states
  — Fixed length code impractical (34 bit codewords)
  — Variable length block code

Data \ State	C,D
10	0100 /C
11	1000 /D
000	000100 /C
010	100100 /C
011	001000 /D
0010	00100100/C
0011	00001000/D

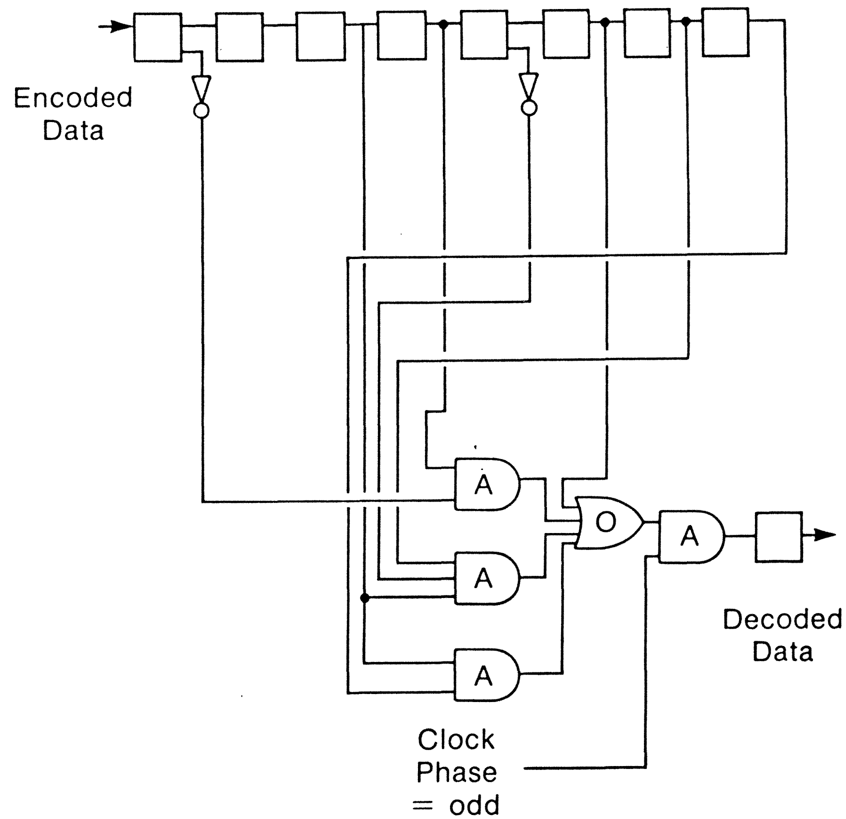
- Practical fixed length encoder obtained by introducing new states
- Sliding block decoder with error propagation $\leq$ 4 data bits

(2,7) Code Implementation

- Encoder logic circuit



- Decoder logic circuit



Look Ahead Coding

- 2/3 (1,7) (Jacoby)
  - — Basic encoding table

Data	Code
0 0	1 0 1
0 1	1 0 0
1 0	0 0 1
1 1	0 1 0

- Potential (1,7) violations: 010101

  00.00 → 101.101

  - — Violation substitution table

Data	Code
0 0•0 0	1 0 1•0 0 0
0 0•0 1	1 0 0•0 0 0
1 0•0 0	0 0 1•0 0 0
1 0•0 1	0 1 0•0 0 0

DATA 00 01 10

CODE 1 $\underbrace{00 \quad 000 \quad 00}_{k=7}$ 1

DATA 11 00 11 00

CODE 010 101 010 101

d = 1

(1,7) Code Implementation

- 98% efficient
- Finite-state fixed-length encoder (2 bits → 3 bits)

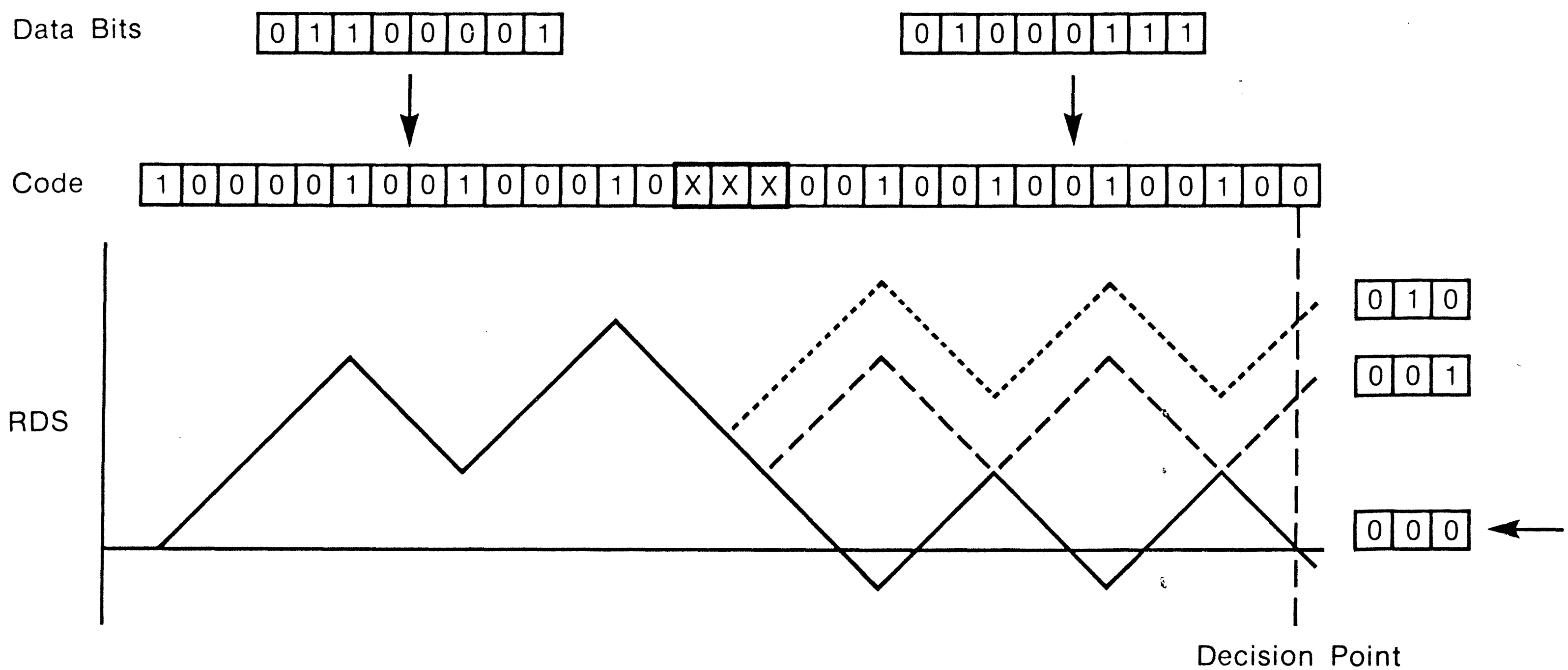
Data \ State	A	B	C	D	V
00	101/V	100/A	001/V	010/A	000/A
01	100/V	100/B	010/V	010/B	000/B
10	101/C	100/C	001/C	010/C	000/C
11	101/D	100/D	001/D	010/D	000/D

State A: Previous input = "00" (no violation)
State B: Previous input = "01" (no violation)
State C: Previous input = "10"
State D: Previous input = "11"
State V: Previous input caused "Violation" pattern

- Sliding block decoder with
  error propagation ≤ 5 data bits

EFM

- Eight to Fourteen Modulation
- 8/17 (2,10;c)
- Look-ahead encoder



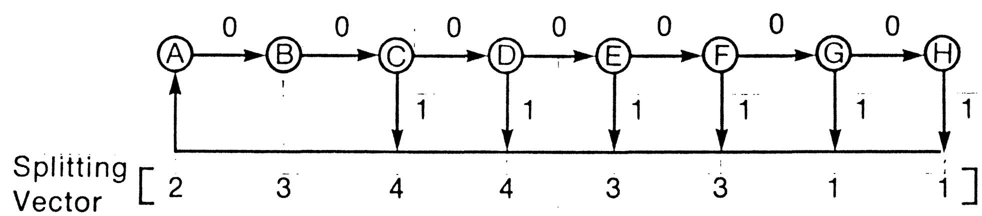
- Block decoder (after discarding merge bits)
- Error propagation ≤ 1 byte

Sliding Block Code Algorithm
(Adler, Coppersmith, Hassner, Marcus)

- General code construction procedure for finite memory channels, e.g. (d,k)
- Produces code at any rate $m/n \leq C$
    - Finite-state encoder
    - Sliding block decoder
    - Limited error propagation
- Based on results in symbolic dynamics
- "Automatic" code construction possible

Sliding Block Code Algorithm

- Generates new graph representation of (D,K) constraints
- Finite-state encoder based on new graph states
- Example: 1/2(2,7)



State-Splitting

21 states
Two 2-bit codewords/state

State-Amalgamation

7 states
Two 2 bit codewords/state

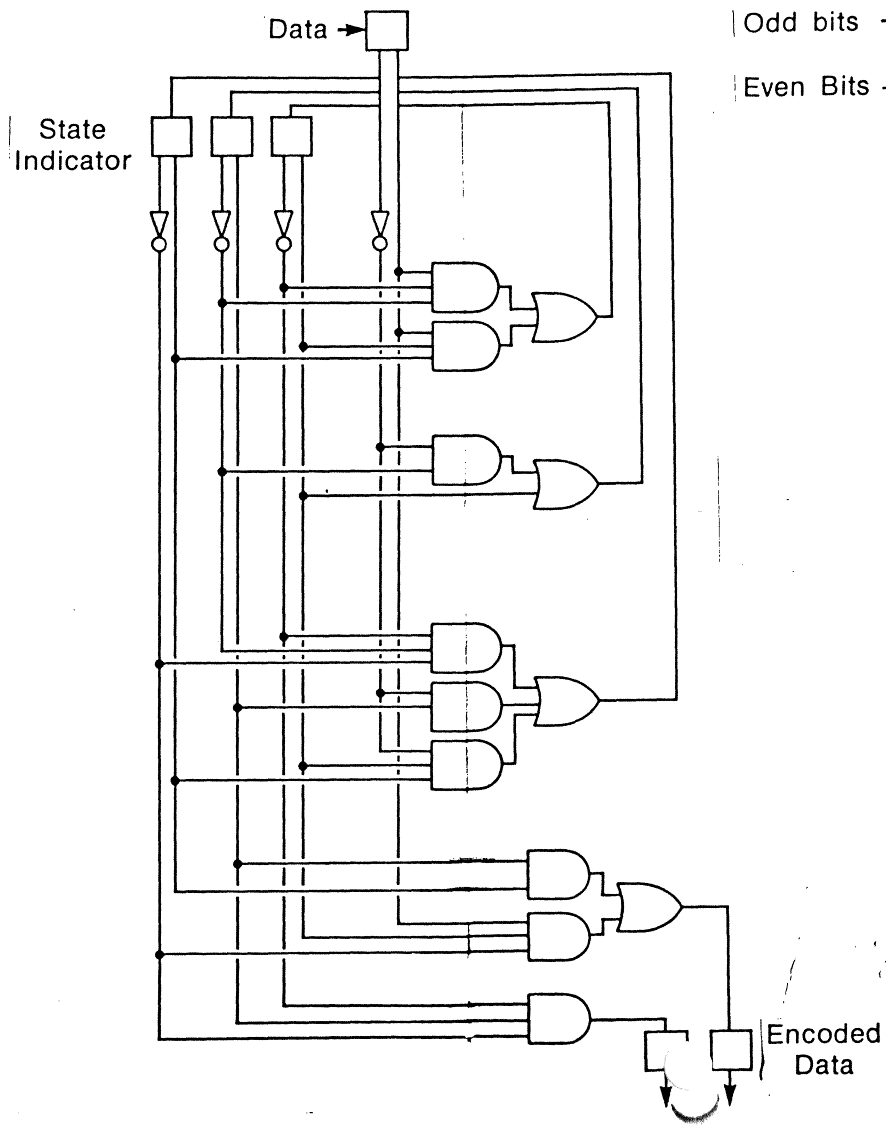
Data Assignment

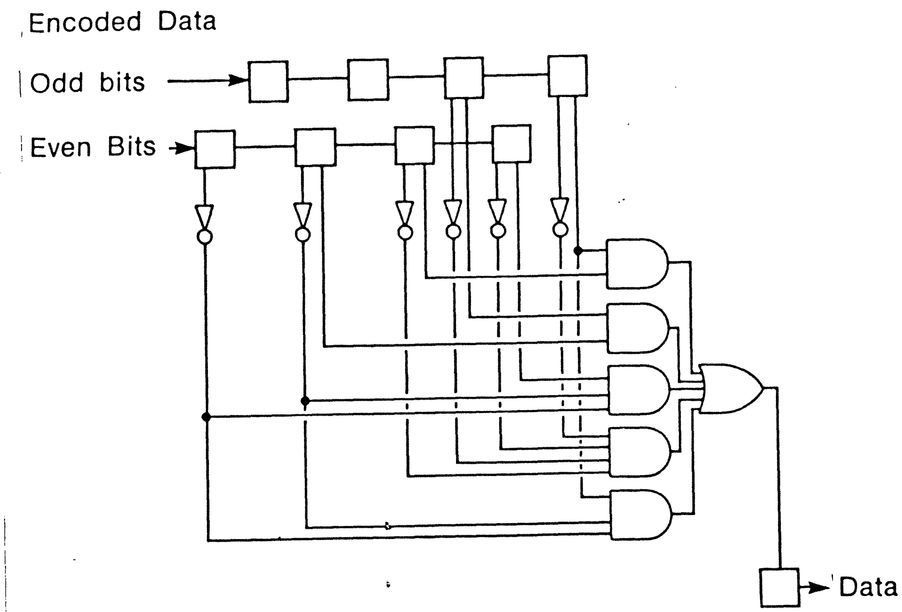
Encoder/Decoder Logic
for (2,7) code

State / Data	A^1	A^2	B	C^1	C^2	C^3	E
0	$00/C^1$	00/B	$01/A^2$	$10/A^2$	00/B	$00/C^1$	$00/C^1$
1	$00/C^3$	$00/C^2$	$01/A^1$	$10/A^1$	10/B	00/E	10/B

Encoder logic circuit

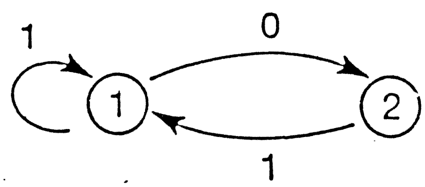
Decoder logic circuit



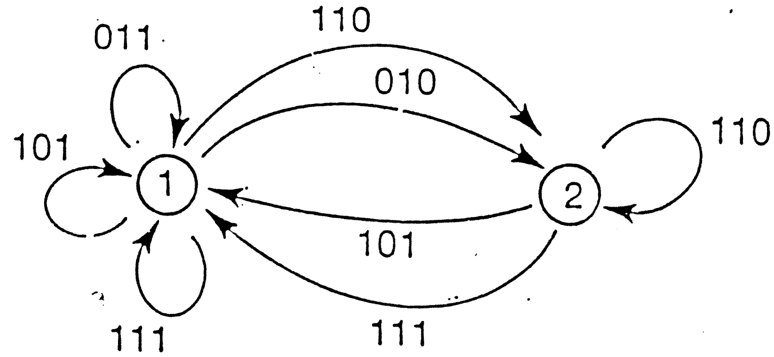


965 - IBM - 04

Construction of 2/3 (0,1) Code



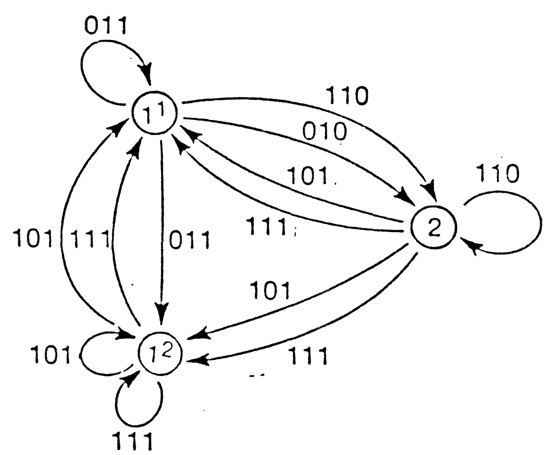
State diagram G for (0,1)



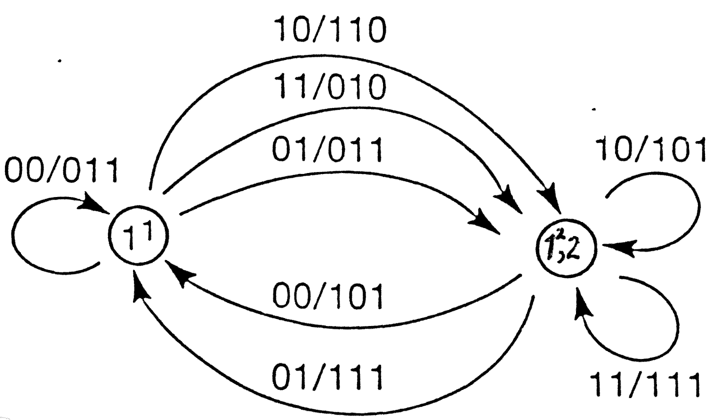
G^3

$$T^3 v = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \geq \begin{bmatrix} 8 \\ 4 \end{bmatrix} = 2^2 v$$

Eigenvector inequality



Split graph



Encoder

	{101,111}	{011,110,010}
011	01	00
110	10	--
010	11	--
101	10	00
111	11	01

Decoder

Constrained Codes for Partial Response Channels

(0,G/I) constraints for $(1 - D^2)$

- 0 = minimum run of 0's
- G = maximum run of 0's in channel output (Global)
- I = maximum run of 0's in even/odd substrings (Interleaved) k constraint on even and odd substrings

Why?

- 0 = no restriction on intersymbol interference
- G = timing/gain control information
- I = limit length of minimum distance error events

Applications

- 8/9 (0,4/4) and (0,3/6) block codes
- 8/9 (0,3/5) sequence-state code
- 8/9 (0,3/3) sliding-block code

MATCHED SPECTRAL NULL CODES

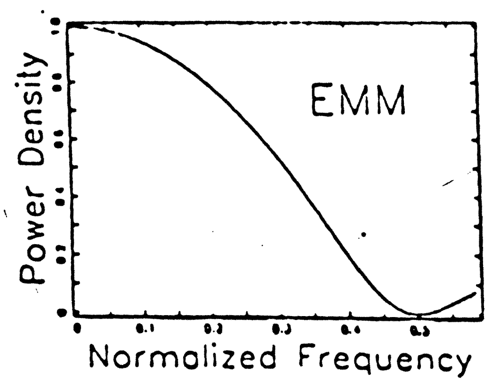
- Improved codes for low SNR optical/magnetic PARTIAL RESPONSE channels
- Code design principle based on new general theorem

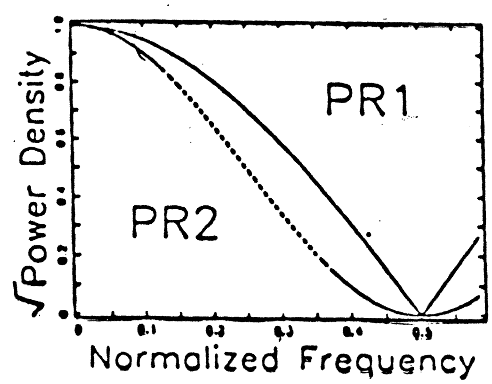
Significant coding gain results if:

$$\left\{\begin{array}{l}\text{null frequencies in}\\ \underline{\text{code}}\ \underline{\text{power}}\ \underline{\text{spectrum}}\end{array}\right\} = \left\{\begin{array}{l}\text{null frequencies in}\\ \underline{\text{channel}}\ \underline{\text{frequency}}\ \underline{\text{response}}\end{array}\right\}$$

EMM
(rate 2/3)

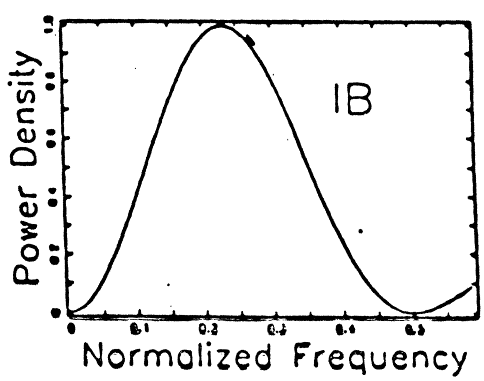
PR1: 3 dB
PR2: 4 dB

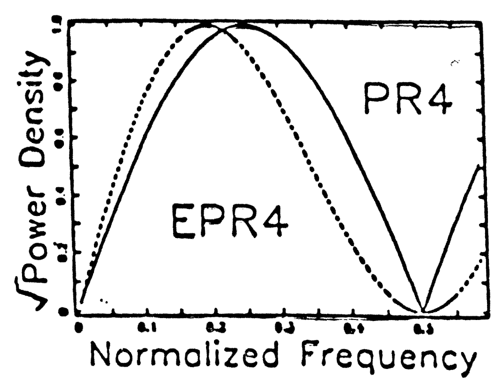




Interleaved
Biphase
(rate 1/2)

PR4: 4.8 dB
EPR4: 4.8 dB





Outline

- Introduction and Overview
- Information Theory
  - State diagrams
  - Shannon capacity
  - Statistical properties
  - Power spectrum calculations
- Design of Run-Length-Limited Codes
  - Code construction techniques
  - Applications
- Design of Spectral Null Codes
  - Characterization of spectral null constraints
  - Applications
- Trellis Codes for Partial Response Channels

LECTURE # 6

TOPICS:

1. Sequence Detection

2. Viterbi algorithm.

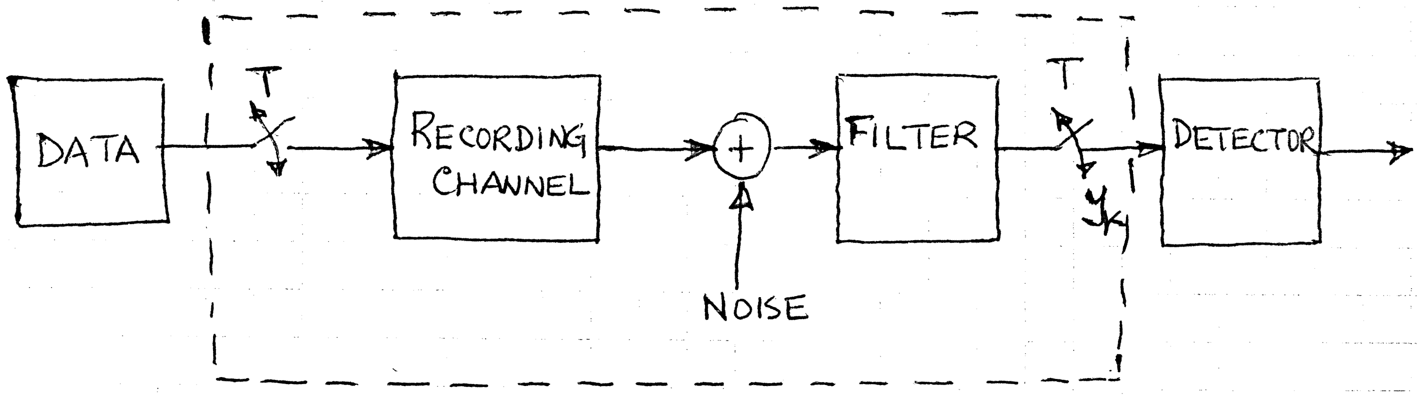
3. Performance measure for sequence detection
   -3dB OF S/N RATIO, SAME BER

4. Overview of decision-feedback equalization.

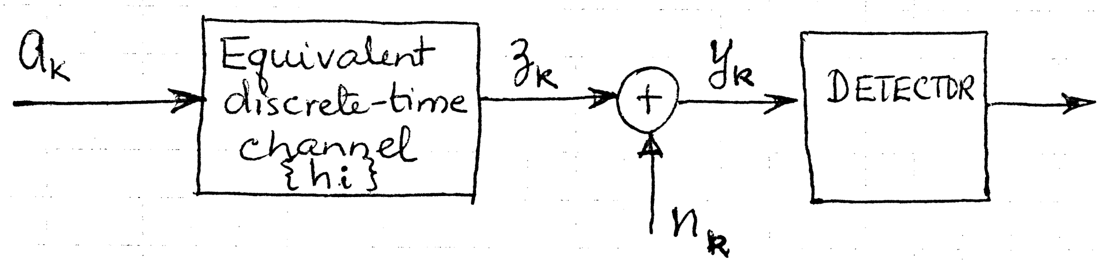
I. SEQUENCE DETECTION:

- applicable to bandlimited channels with controlled intersymbol interference (isi)

- provides improved performance relative to bit-by-bit detection methods (such as peak detection, threshold detection).



$\Downarrow$



$$z_k = \sum_i h_i a_{k-i} \quad \text{(Finite-impulse response model)}$$

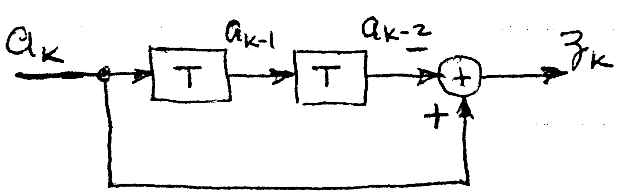
FINITE # OF SAMPLE VALUES

That is, the noise-free sample is a linear combination of the current and previous inputs.

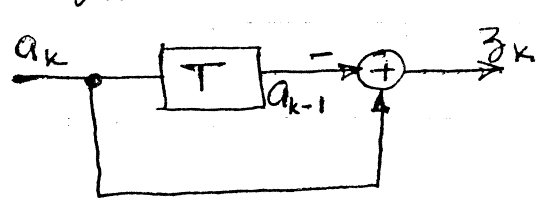
THRESHOLD DETECTION IGNORES DATA DEPENDENCY/ ANY BIT EFFECTS FUTURE & PRESENT ENCODING.

e.g. modified duobinary : $z_k = a_k - a_{k-2}$

(1-D) channel : $z_k = a_k - a_{k-1}$



modified duobinary



(1 − D)

The observed sample value y_k is corrupted by noise:

$$y_k = z_k + n_k$$

The detection problem is to faithfully estimate a_k from the noisy observations.

- Let the recorded data sequence be of length N; that is, $a_1\ a_2 \ldots a_N$ is recorded.

$\therefore$ Number of possible sequences to choose from is $M \triangleq 2^N$.

Define $\underline{a}^i = [\,a_1^i\ a_2^i\ \ldots\ a_N^i\,]$

$$i = 1, 2, \ldots, M.$$

Now each $\underline{a}^i$ produces a corresponding unique $\underline{z}^i$ (the noise-free sequence).

That is,

$$\underline{z}^i = [\,z_1^i\ z_2^i\ \ldots\ z_N^i\,]$$

$$i = 1, 2, \ldots, M.$$

and

$$\underline{z}^i \Longleftrightarrow \underline{a}^i$$

If the random variable n_k (the noise) is an uncorrelated sequence with Gaussian distribution, then the optimum detector selects the sequence $\underline{z}^k$ if

$$(\underset{\substack{\uparrow \\ \text{OBSERVED}}}{\underline{y}} - \underline{z}^k)^2 < (\underline{y} - \underline{z}^i)^2 \quad \text{for } i \neq k \text{ and } i = 1, 2, \ldots, M$$

CHOOSE SEQUENCE CLOSEST TO OBSERVED SEQUENCE

Equivalently, the optimum detector computes the cost ("distance") associated with each possible sequence $\underline{z}^i$, and selects the one that corresponds to the minimum.

Let $J(\underline{z}^i) \triangleq$ cost ("distance") for sequence $\underline{z}^i$

$$= (\underline{y} - \underline{z}^i)^2$$

Since

$$\underline{y} = [y_1 \; y_2 \; \ldots \; y_N]$$
$$\underline{z}^i = [z_1^i \; z_2^i \; \ldots \; z_N^i]$$

Then

$$J(\underline{z}^i) = (\underline{y} - \underline{z}^i)^2$$

$$= \sum_{l=1}^{N} (y_l - z_l^i)^2 = (y_1 - z_1)^2 + (y_2 - z_2)^2 + (y_3 - z_3)^2 + \ldots$$

DISTANCE SQUARED TO CORRECT (VALID) VECTOR

Optimum detector: $\min\limits_{i} J(\underline{z}^i)$

CLOSEST SEQUENCES ARE 1 BIT AWAY FROM NEIGHBORING VECTOR

Note that the possible values of z_ℓ^i depend on the equivalent discrete-time channel.

For example, for modified duobinary z_ℓ^i can take on 3 possible values: $0, \pm 2$.

Thus, for each possible $\underline{a}^i$, one can pre-compute the corresponding $\underline{z}^i$ and then solve the above minimization problem:

- If

  $N = 5 \Rightarrow$ 32 costs to be computed and compared.

  $N = 10 \Rightarrow$ 1024 costs to be computed and compared.

  $N = 16 \Rightarrow$ 64k costs to be computed and compared

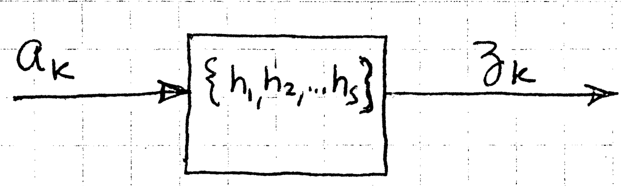
  ⋮

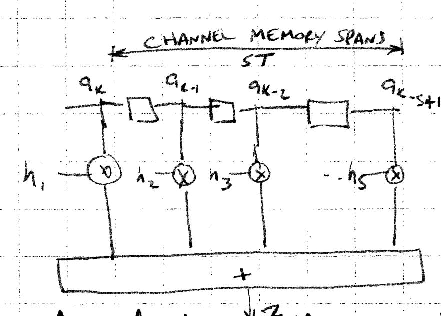
Exponential growth with N in the number of costs/comparisons — IMPRACTICAL FOR EVEN SMALL VALUES OF N !!

- Viterbi algorithm can be used to perform the same task, but without the exponential growth in the number of required computations. It performs maximum likelihood detection with specific (fixed) computational requirements per bit interval.

II. VITERBI ALGORITHM:

- Finite-state description of the channel:





span of the equivalent discrete-time channel $= sT$.

z_k = STATE OF CHANNEL MEMORY + INPUT

Let $[a_{k-1}\ a_{k-2} \cdots a_{k-s+1}] \triangleq$ state of the channel.

$\therefore$ Number of possible states $= 2^{s-1}$

Given the state of the channel, the values of $\{h_i\}$, and the input a_k, one can determine the noise-free output z_k.

EXAMPLES:

a) (1 − D) channel:

$$z_k = a_k - a_{k-1}$$

state is defined by the value of a_{k-1}. Since a_{k-1} can take on 2 values, the channel can be in two possible states.

SINGLE INFORMATION BITS

b) $(1-D^2)$ channel (modified duobinary):

$$z_k = a_k - a_{k-2} \, (+ \, 0 \cdot a_{k-1})$$

state is defined by the value of $a_{k-1} \, a_{k-2}$. Since the input is binary, there are four possible states.

- Trellis diagram for a channel:

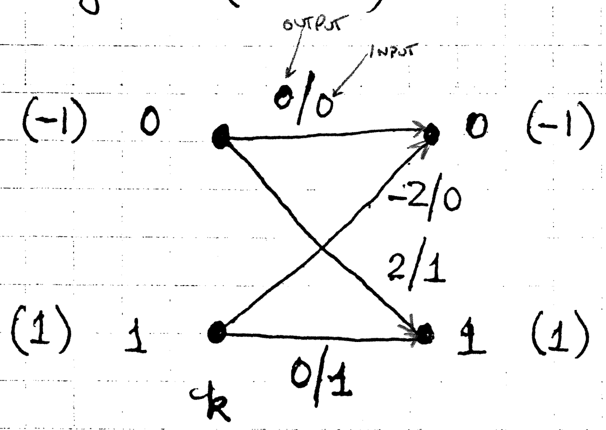
Trellis diagram is a graph with nodes and branches.

Nodes represent the possible states of the channel

Branches represent the $\boxed{\text{allowed}}$ inter-state transitions from time k to k+1.

e.g. (1 − D) channel:

CONSTRUCT FOR DUOBINARY



branch labels:

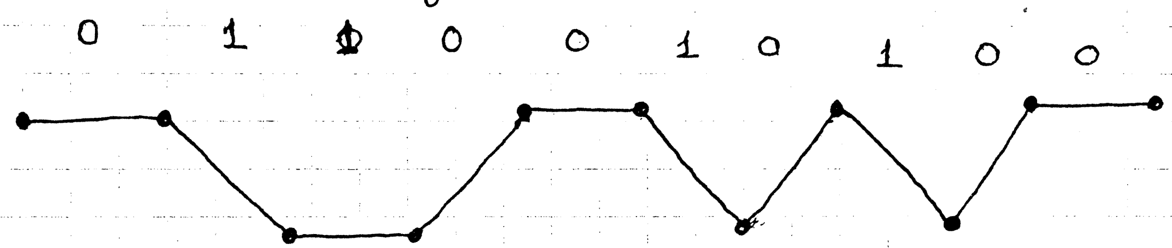
z_k / a_k ← input data.

↑ output sample value

$a_k = \{0, 1\}$

Consider the sequence 0 1 1 0 0 1 0 1 0 0.

The corresponding path in the trellis is



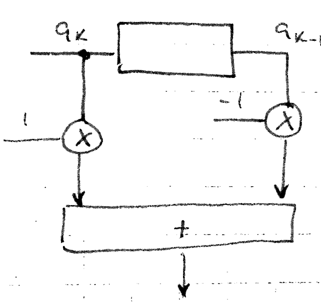
and the corresponding output (noise-free) sample sequence is

0 2 0 −2 0 2 −2 2 −2 0

The function of the Viterbi algorithm is to trace the most likely path based on <u>noisy</u> observations y_k.

MEMORY OF CHANNEL (# OF STATES) DETERMINES COMPLEXITY OF VETERBI DECODER.

$$z_k = a_k - a_{k-1}$$
$$= -1 - (-1) = 0$$
$$= 1 - (-1) = 2$$



- Viterbi algorithm:

  - **Branch metric** is the cost associated with a transition. For the AWGN channel, it is equal to

$$\left(y_k - z^b\right)^2$$

y_k: observed sample value

z^b: label associated with a specific branch in the trellis.

e.g. for the (1-D) channel, there are only 4 branch metrics per bit interval. These are

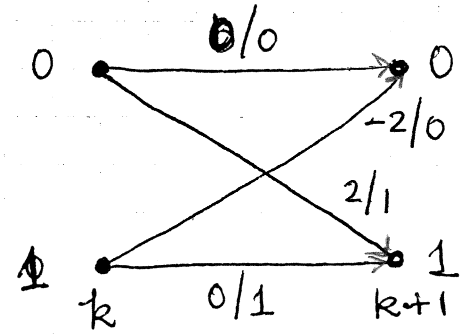
$$(y_k - 0)^2 = y_k^2$$
$$(y_k - 2)^2 = y_k^2 - 4y_k + 4$$
$$(y_k + 2)^2 = y_k^2 + 4y_k + 4$$

  - **Augmented path metric** is the cost of moving along a specific path from time k to (k+1). It equals the sum of the path metric at time k and a specific branch metric at time k.

- Path metric is the minimum augmented path metric into a specific state.

- Survivor path (sequence) is the path associated with the path metric.

EXAMPLE: (1−D) channel



Branch metrics:
$$b_k^0 = (y_k - 0)^2$$
$$b_k^2 = (y_k - 2)^2$$
$$b_k^{-2} = (y_k + 2)^2$$

Augmented Path metrics:

$$P_{k+1}^{00} = P_k^0 + b_k^0 \quad \text{(from state } 0 \to 0\text{)}$$
$$P_{k+1}^{10} = P_k^1 + b_k^{-2} \quad \text{(from state } 1 \to 0\text{)}$$
$$P_{k+1}^{01} = P_k^0 + b_k^2$$
$$P_{k+1}^{11} = P_k^1 + b_k^0$$

DETERMINED BY MEMORY OF CHANNEL

PATH METRICS = # STATES
SURVIVOR SEQUENCES = # STATES

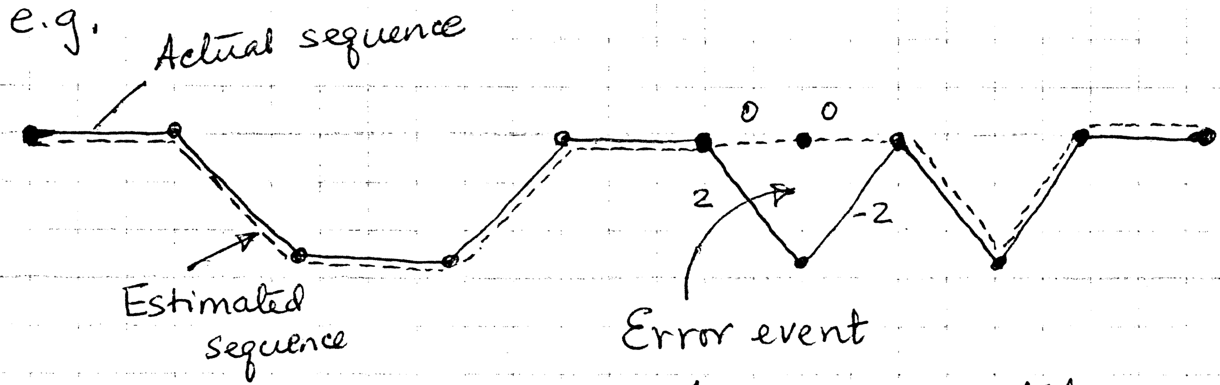
Path metrics:
$$P_{k+1}^0 = \min\left[P_{k+1}^{00}, P_{k+1}^{10}\right]$$
$$P_{k+1}^1 = \min\left[P_{k+1}^{01}, P_{k+1}^{11}\right]$$

COMPLEXITY?

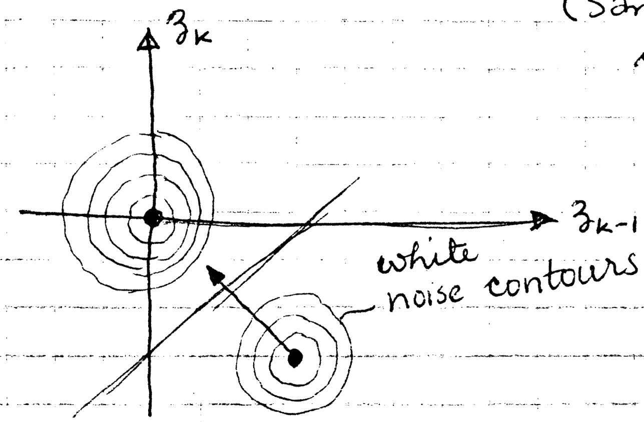
III. PERFORMANCE OF THE VITERBI ALGORITHM

- ERROR EVENT : actual sequence is different from estimated sequence.

e.g.



(same starting state; same ending state)

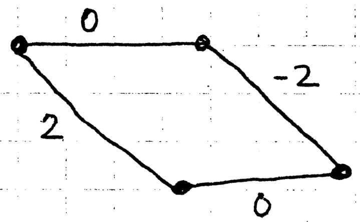


The performance with sequence detection depends on the distance between allowed sequences.

$$d^2(\underline{z}^\alpha, \underline{z}^\beta) = \text{distance between sequences } \underline{z}^\alpha \text{ and } \underline{z}^\beta$$

$$\triangleq \sum_{\ell} (z_\ell^\alpha - z_\ell^\beta)^2 \qquad \text{(for AWGN channel)}$$

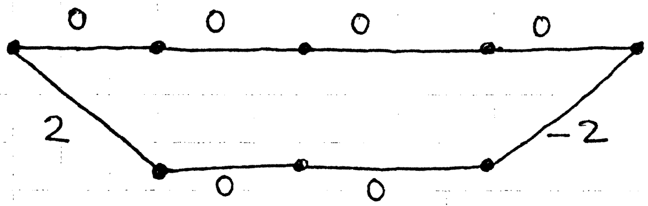
e.g., (1-D) channel



$\underline{z}^{\alpha} = (0, -2)$

$\underline{z}^{\beta} = (2, 0)$

$$d^2(\underline{z}^{\alpha}, \underline{z}^{\beta}) = (0-2)^2 + (-2-0)^2$$
$$= 8$$



$\underline{z}^{\alpha} = (0, 0, 0, 0) \qquad \underline{z}^{\beta} = (2, 0, 0, -2)$

$$d^2(\underline{z}^{\alpha}, \underline{z}^{\beta}) = (0-2)^2 + (0-0)^2 + (0-0)^2 + (0-(-2))^2$$
$$= 8.$$

- Minimum distance = minimum of all the distances between allowed sequences of samples.

let $d_{min} \triangleq$ minimum distance.

Then, probability of error with Viterbi algorithm

$$P_e = K\, Q\left(\frac{d_{min}}{2\sigma}\right)$$

noise standard deviation

a constant (determined empirically).

EXAMPLE : for modified duobinary

$$d_{min} = \sqrt{8} = 2\sqrt{2} .$$

$$P_e \approx Q\left(\frac{\cancel{2}\sqrt{2}}{\cancel{2}\sigma}\right) = Q\left(\frac{\sqrt{2}}{\sigma}\right)$$ (with Viterbi detection)

Recall,

$$P_e \approx Q\left(\frac{1}{\sigma}\right)$$ (with threshold detection)

at low error rate (high SNR), the performance gain due to Viterbi detection relative to threshold detection is

$$20 \log_{10}\left(\frac{\text{argument of the Q function for Viterbi detection}}{\text{argument of the Q function for threshold detection}}\right).$$

e.g. for modified duobinary,

gain due to Viterbi detection is

$$20 \log_{10}\left(\frac{\sqrt{2}/\sigma}{1/\sigma}\right) = 20 \log_{10}\sqrt{2} = 3\text{dB}.$$

With Viterbi detector, we need approx. 14.3dB SNR to get an error rate of 10^{-7}.

5/17/89

SHANNON & WEAVER
MATH. THEORY OF COMMUNICATION

FSTD

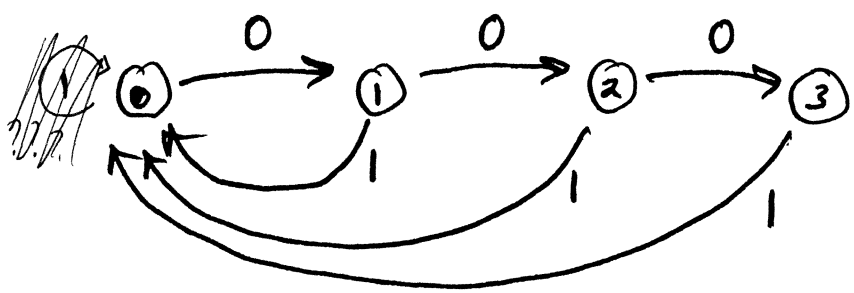
Lecture 8: Information Theory

Representation of Constraints

Defn: A discrete noiseless channel (DNC) is a set of sequences obtained by walks along a labelled, directed finite graph

Terminology: Finite directed graph with edge labels is called a Finite State Transition Diagram (FSTD)

Example: Run-Length-Limited (RLL) (1, 3) constraint

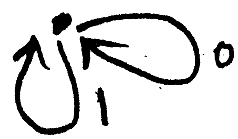


Remark 1: Other names for DNC

- Sofic system (symbolic dynamics)
- Regular language
- Constrained channel

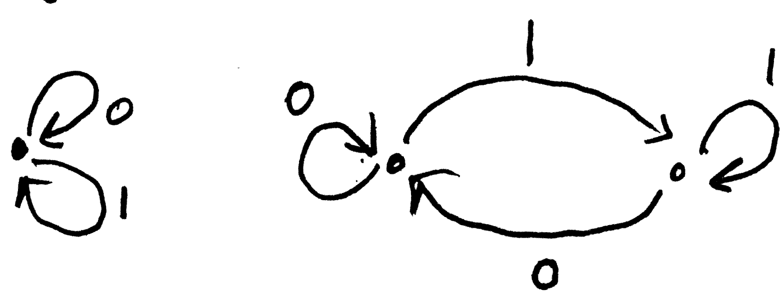
Remark 2: "Multiple" edges allowed between states; "Loops" allowed

RLL $(0,\infty)$



Remark 3: Same constraint can be represented by different graphs

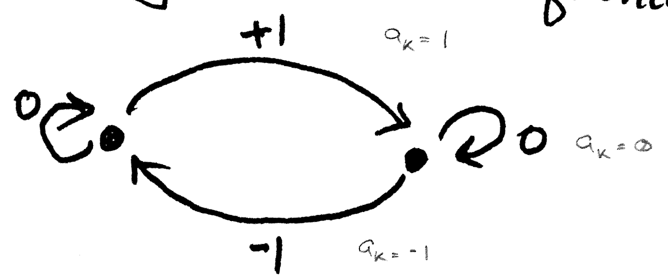
RLL $(0,\infty)$



Remark 4: Non-binary symbol alphabets and variable length symbols allowed

(USED WITH PARTIAL RESPONSE OR MEMORY CHANNEL)

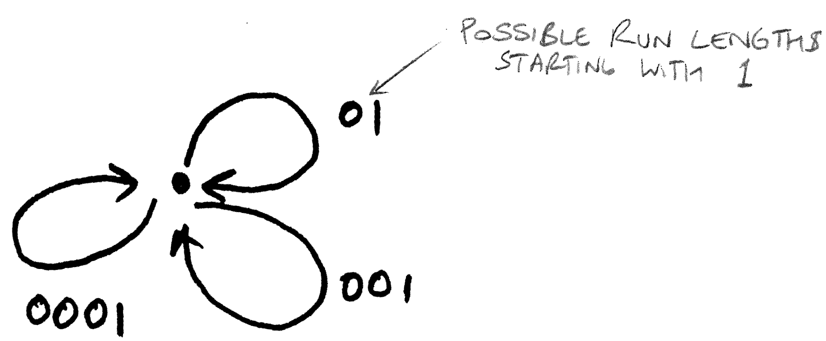
Ex. Duobinary or dicode sequences - ternary



$(1 - D)$

$+1\ [0 \ldots 0]\ -1\ [0 \ldots 0]$

Ex. Variable length representation of (1, 3)



1
101
1001
10001
3

Important example: Unconstrained system on N-ary alphabet $\mathcal{U}_N$ $(0 \ldots N-1)$
(Full N-shift)

SHIFTING SEQUENCES YIELDS A SEQUENCE STILL IN THE SET

e.g. $\mathcal{U}_2 = RLL(0, \infty)$

CODING PROBLEM: Convert arbitrary (unconstrained) data sequences efficiently into sequences constrained by DNC.

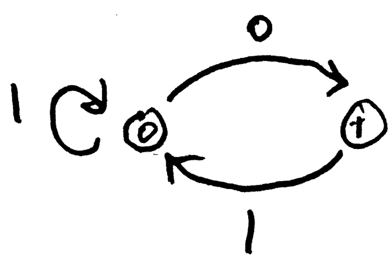
$$\mathcal{U}_2 \xrightarrow[m]{} \boxed{} \xrightarrow[n]{} S$$

Defn: The m^{th} power, S^m, of a DNC S is the system obtained by blocking sequences of S into (non-overlapping) groups of m symbols

$$\underbrace{x_{-m} \cdots x_{-1}}_{X_{-1}}, \underbrace{x_0\, x_1 \cdots x_{m-1}}_{X_0}, \underbrace{x_m \cdots x_{2m-1}}_{X_1}$$

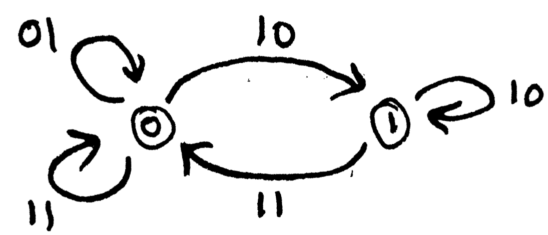
Remark: If S is described by FSTD G, then S^m is described by G^m, which has the same states as G, and an edge from state i to state j for every path of length m from i to j in G, with corresponding m-block label.

Example: $S = RLL(0,1)$ - find S^2 and S^3



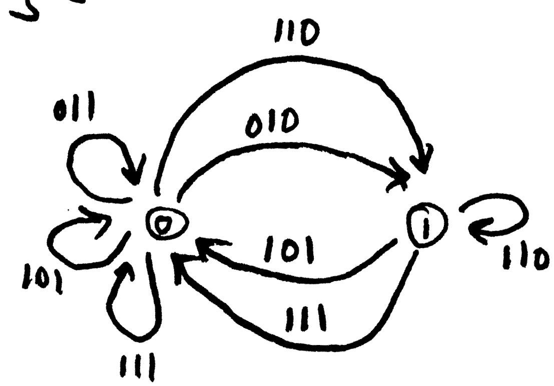
FSTD G for S

VALID 1 BIT SEQUENCES



FSTD G^2 for S^2

VALID 2 BIT SEQUENCES



FSTD G^3 for S^3

VALID 3 BIT SEQUENCES

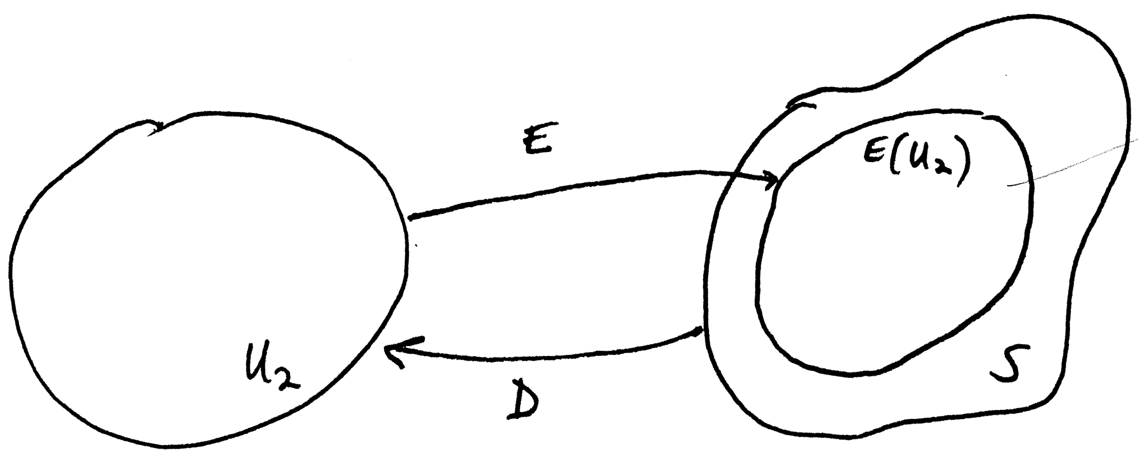
Defn: A *rate m:n code* from $\mathcal{U}_2$ to S is a pair of maps.

Encoder: $E: \mathcal{U}_2^m \longrightarrow S^n$ $\qquad \xrightarrow{x} \boxed{E} \xrightarrow{y}$

Decoder: $D: S^n \longrightarrow \mathcal{U}_2^m$ $\qquad \xrightarrow{y} \boxed{D} \xrightarrow{x}$

where $D \circ E = I$, the identity function

i.e. $D(E(x)) = x$. DECODER PROPERLY DECODES ENCODER



MAY NOT MAP INTO EVERY CODE SEQUENCE

Finite-State-Machine Encoder

Defn: A finite-state-machine (FSM) is a synchronous system consisting of:

1) a finite input alphabet $\mathcal{X} = \{\alpha_1, \ldots, \alpha_p\}$
2) a finite output alphabet $\mathcal{Y} = \{\beta_1, \ldots, \beta_q\}$
3) a finite state set $\mathcal{Z} = \{\sigma_1, \ldots, \sigma_m\}$
4) a pair of characterizing functions

f_y : output function

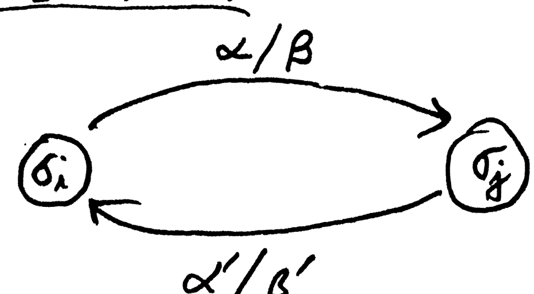
and f_z : state function

given by:

OUTPUT $$y_t = f_y(x_t, z_t)$$

NEXT STATE $$z_{t+1} = f_z(x_t, z_t) \quad \text{for time } t = 1, 2, \ldots$$

Graphical representation:



Labels : input/output

e.g. $x_1 \cdots x_m / y_1 \cdots y_m$

7

Sliding Block Decoder

Want decoder function to be state-independent (BLIND TO STATE)

Defn: $D: S^m \to U_2^m$ is a (finite) sliding block mapping

if $X_j = D\left(\left(Y_k\right)_{k=-\infty,\dots,\infty}\right)_j$

is given by: (FINITE)

$$X_j = \tilde{D}\left(Y_{j-N}, \dots, Y_j, \dots, Y_{j+N}\right)$$

WINDOW OF N SYMBOLS TO LEFT & RIGHT OF j

where

$$\tilde{D}: \mathcal{Y}^{2N+1} \longrightarrow \mathcal{X}$$

groups of 2N+1 output — input blocks

WINDOW IS 2N+1 SYMBOLS

N=0 - BLOCK MAPPING, 1:1 MAPPING

In pictures:

$$Y_0\ Y_1 \cdots \underbrace{Y_{j-N} \cdots Y_j \cdots Y_{j+N}} \cdots$$

$$\downarrow \tilde{D}$$

$$X_j$$

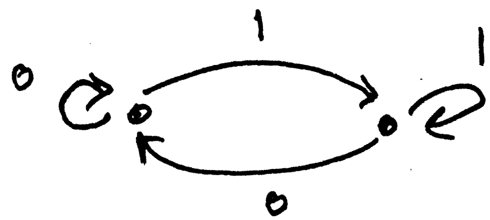
DEPENDS ON SOME COMBINATIONS OF SYMBOLS

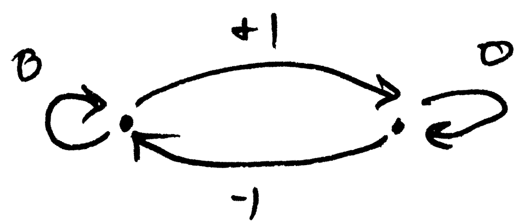
$N=0 \Rightarrow$ block mapping

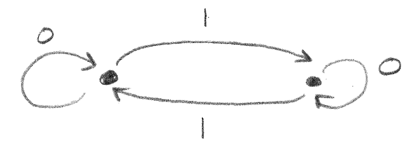
CODE CONSTRUCTION IDEA: Exploit similar structure of FSTD and FSM

Example: Rate 1:1 code from U_2 (ARB BINARY DATA) to duobinary

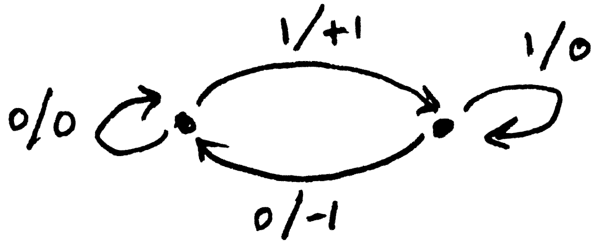
FSTD



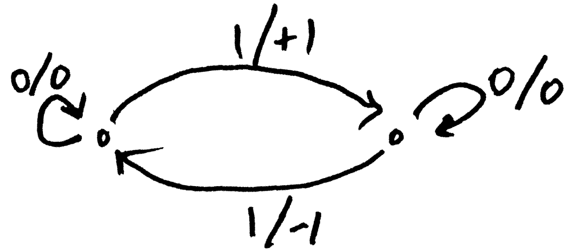




Encoder FSM



Better representation of U_2



Decoder

Not sliding block! NO FINITE WINDOW

+1 00 ·····0 → 1 1 1 1 1 ···1

-1 00 ····0 → 0 00 ····0

INFINITE WINDOW REQUIRED ∴ NOT SLIDING BLOCK

Sliding block decoder

$N = 1$

0 → 0

+1 → 1

-1 → 1

MORE SPECIFIC TECHNIQUE: Rate $m:n$ code from U_2 to S

Find graph H with following properties:

1) Each vertex has 2^m outgoing edges

2) DATA LABELS: Each m-block appears as a label of one edge from each vertex

3) CODE LABELS: Each edge has an n-block from S. The FSTD given by H with these labels represents S^n, or a subsystem of S^n

GENERATES VALID CODE SEQUENCES

4) DECODABILITY (STATE-FREE): For k large enough, all paths $e_{-k} \dots e_{-1} e_0 e_1 \dots e_k$ which generate the same code sequence have the same data label on edge e_0

ONLY ONE DATA SYMBOL

ENCODING: Pick initial state. Follow edges according to data inputs, reading off code labels

DECODING: Sliding block, with look-ahead and look-back of k symbols in S^n

<u>Defn</u>: The state-transition matrix T corresponding to G is given by:

$$t_{ij} = \begin{cases} k & \text{if there are } k \text{ edges from state } i \text{ to state } j \text{ in } G \\ 0 & \text{if no edges } i \to j \end{cases}$$

ADJACENCY MATRIX

<u>Remark 1</u>: The <u>row-sums</u> of T indicate the number of outgoing edges from the states of G.

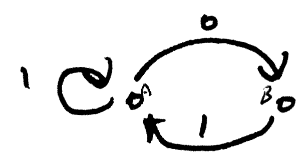
2^M EDGES

<u>Remark 2</u>: The state-transition matrix of G^m is T^m (Exercise)

(USE MATRIX MULTIPLICATION)
IT COUNTS PROPERLY THE EDGES COMING OUT OF GRAPH

<u>Example</u>: RLL(0,1)

1 BIT → 1 SYMBOL



$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

$$T = \begin{array}{c|c} & \begin{array}{cc} A & B \end{array} \\ \begin{array}{c} A \\ B \end{array} & \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \end{array} \quad \begin{array}{c} 2 \\ 1 \end{array}$$

NOT A GOOD GRAPH

row sums

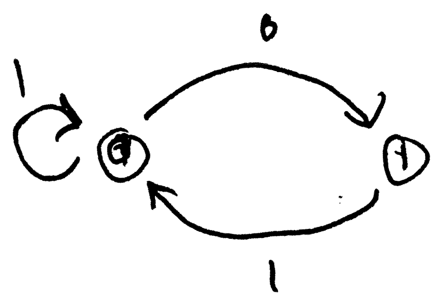
$$T^2 = \begin{array}{c|c} & \begin{array}{cc} A & B \end{array} \\ \begin{array}{c} A \\ B \end{array} & \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \end{array} \quad \begin{array}{c} 3 \\ 2 \end{array}$$

ENOUGH EDGES

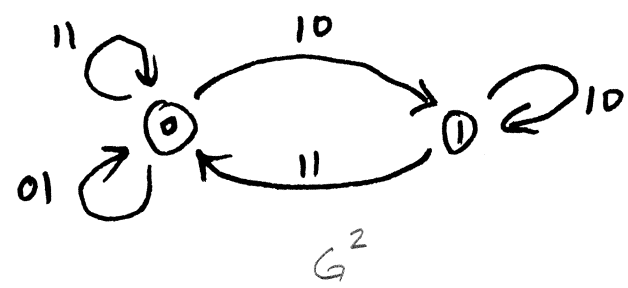
row sums

$\Rightarrow$ 1:2 code possible from U_2 to RLL(0,1)

CODE CONSTRUCTION EXAMPLE: Rate 1:2 code from U_2 to RLL(0,1) (Frequency Modulation, FM)

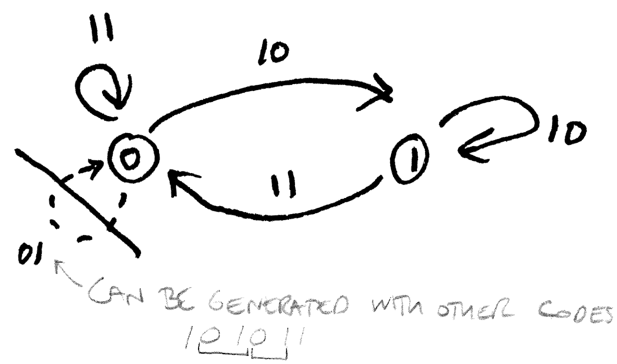


G



G^2

- Discard extra edge at state 0
- Assign data labels 1/11, 0/10



- "Merge" states 0 and 1 to get state-free encoder (block code)

$1 \rightarrow 11$
$0 \rightarrow 10$

- Decoder is obviously a block decoder

REMARK: Code design involves choices: discarding edges, assigning data labels

CODE RATES AND SHANNON CAPACITY

Defn. For a DNC, the Shannon Capacity C is defined by the limit:

$$C = \lim_{n \to \infty} \frac{1}{n} \log_2 s_n$$

where

$$s_n = \#\text{ of sequences of length } n \text{ in } S$$

2^{cn} CODE SEQUENCES OF LENGTH n

$2^M < 2^{cn}$

$\uparrow$ # DATA SEQUENCES

$\frac{M}{n} \leq C$

RATE CAPACITY

Remark 1: Units are bits/symbol. $\frac{\text{CODE BITS}}{\text{CODE SYMBOL}}$

Also use the notation $Cap(S)$

Remark 2: Capacity is also called topological entropy (symbolic dynamics)

Coding Theorems

Theorem 1 [Shannon] If there exists a code at rate $m:n$ from $\mathcal{U}_2$ to S, then

$$m/n \leq \mathrm{Cap}(S) = C$$

CAPACITY IS ULTIMATE LIMIT

Theorem 2 [Shannon] For rates $m/n < \mathrm{Cap}(S)$, there exists a code from $\mathcal{U}_2$ to S with a (finite) sliding block decoder with rate $mk : nk$, for some $k \geq 1$

CONVERSE

CODE MAY NEED TO GROUP SYMBOLS

CALCULATING CAPACITY

FACT 1: C represents the growth rate of the number of sequences of length n in S.

FACT 2: IF G is irreducible (every state i reaches every state j)
[Shannon]
and Shannon (no sequence is generated by more than one path from a given state)

then

$$C = \log_2 \lambda$$

where

λ = largest eigenvalue of T MATRIX

QA 184
251

Review of Linear Algebra

Eigenvector equation:

$$Tv = \lambda v, \quad v \neq 0$$

$$(T - \lambda I)v = 0$$

Requires: $\underbrace{\det(T - \lambda I)}_{\text{characteristic polynomial}} = 0$

$-\lambda I$

$$\begin{pmatrix} -\lambda & & & & \\ & -\lambda & & & \\ & & -\lambda & & \\ & & & -\lambda & \\ & & & & \ddots \\ & & & & & -\lambda \end{pmatrix}$$

$$\text{DET} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = a_{11}\,\text{DET}\begin{pmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{pmatrix} - a_{21}\,\text{DET}\begin{pmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{pmatrix} + a_{31}\,\text{DET}\begin{pmatrix} a_{12} & a_{13} \\ a_{22} & a_{23} \end{pmatrix}$$

PERRON-FROBENIUS THEOREM

$$\text{DET}\begin{pmatrix} -\lambda & 1 & 0 \\ 1 & -\lambda & 1 \\ 1 & 0 & -\lambda \end{pmatrix}$$

Example: RLL $(0,1)$ $\quad T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$

$$\det(T - \lambda I) = \det\begin{bmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{bmatrix} = (1-\lambda)(-\lambda) - 1$$

$$= \lambda^2 - \lambda - 1 = \emptyset$$

So, by quadratic formula:

$$\lambda_1 = \frac{1+\sqrt{5}}{2}, \quad \lambda_2 = \frac{1-\sqrt{5}}{2} \Rightarrow C = \log_2 \frac{1+\sqrt{5}}{2} \approx .692$$

CAPACITY OF $(0,1)$ CONSTRAINT

CAPACITY OF VARIABLE LENGTH GRAPHS

FACT 3: [Shannon] If S has variable length symbols, with lengths $\ell_{ij}^{(k)}$, then

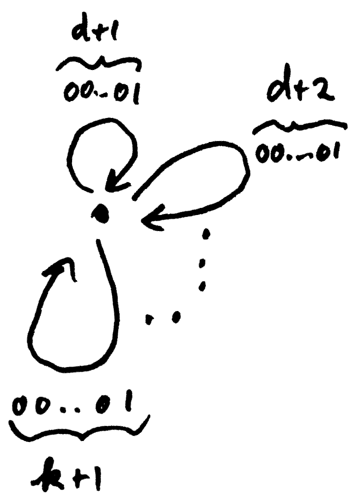
LENGTH OF k^{th} SYMBOL GOING FROM i TO j

$$\mathrm{Cap}(S) = \log_2 \lambda,$$

where λ = largest real root of

$$\det\left[\left(\sum_k \lambda^{-\ell_{ij}^{(k)}}\right)_{ij} - I\right] = 0$$

Example: RLL(d,k)



SCALAR

$$\det\left[\left(\sum_{\ell=d+1}^{k+1} \lambda^{-\ell}\right) - 1\right] = 0$$

or

$$\left(\sum_{\ell=d+1}^{k+1} \lambda^{-\ell}\right) - 1 = 0 \qquad -\lambda^{k+1}\left[\sum_{\ell=d+1}^{k+1} \lambda^{-\ell}\right] = 0$$

$$\Rightarrow \boxed{\lambda^{k+1} - \lambda^{k-d} - \dots - \lambda - 1 = 0}$$

$$-1\left[\lambda^{k-d} + \lambda^{k-d-1} + \dots + \lambda + 1 - \lambda^{k+1}\right]$$

TABLE I
CHANNEL CAPACITIES OF CCRLL CODES

		c													
d	k	1	2	3	4	5	6	7	8	9	10	11	12	13	∞
0	1	.5000	.6358	.6662	.6778	.6834	.6866	.6885	.6898	.6907	.6914	.6919	.6922	.6925	.6942
0	2	--	.7664	.8244	.8468	.8578	.8640	.8678	.8704	.8722	.8734	.8744	.8751	.8757	.8792
0	3	--	.7925	.8704	.9012	.9165	.9252	.9306	.9342	.9367	.9386	.9399	.9410	.9418	.9468
0	4	--	--	.8832	.9120	.9380	.9486	.9552	.9596	.9627	.9650	.9667	.9680	.9690	.9752
0	5	--	--	.8858	.9256	.9460	.9578	.9652	.9702	.9738	.9763	.9783	.9798	.9810	.9881
0	6	--	--	--	.9273	.9488	.9614	.9694	.9747	.9786	.9811	.9834	.9851	.9864	.9942
0	7	--	--	--	.9276	.9497	.9627	.9710	.9766	.9806	.9836	.9858	.9875	.9888	.9971
0	8	--	--	--	--	.9499	.9632	.9717	.9774	.9815	.9845	.9868	.9886	.9900	.9986
0	9	--	--	--	--	.9500	.9633	.9719	.9777	.9819	.9849	.9873	.9891	.9905	.9993
1	2	--	.3471	.3822	.3931	.3978	.4003	.4018	.4027	.4034	.4038	.4041	.4044	.4046	.4057 ←
1	3	--	.4248	.5000	.5237	.5341	.5396	.5428	.5449	.5463	.5473	.5480	.5486	.5490	.5515
1	4	--	--	.5391	.5746	.5905	.5989	.6039	.6702	.6093	.6109	.6121	.6129	.6136	.6175
1	5	--	--	.5497	.5947	.6153	.6263	.6328	.6371	.6400	.6421	.6436	.6448	.6457	.6509
1	6	--	--	--	.6020	.6260	.6391	.6470	.6522	.6557	.6582	.6604	.6615	.6626	.6690
1	7	--	--	--	.6039	.6305	.6451	.6540	.6599	.6639	.6668	.6689	.6706	.6718	.6793
1	8	--	--	--	--	.6321	.6477	.6574	.6638	.6682	.6713	.6737	.6755	.6769	.6853
1	9	--	--	--	--	.6325	.6488	.6590	.6657	.6704	.6738	.6763	.6783	.6798	.6888
1	10	--	--	--	--	--	.6492	.6597	.66666	.6715	.6751	.6777	.6798	.6814	.6909
1	11	--	--	--	--	--	.6493	.6600	.6671	.6721	.6758	.6785	.6806	.6823	.6922
1	12	--	--	--	--	--	--	.6601	.6673	.6724	.6761	.6789	.6811	.6828	.6930
2	3	--	.2028	.2625	.2757	.2807	.2832	.2845	.2853	.2859	.2863	.2866	.2868	.2869	.2878
2	4	--	--	.3471	.3777	.3893	.3950	.3981	.4001	.4013	.4022	.4029	.4034	.4038	.4057 ←
2	5	--	--	.3723	.4199	.4384	.4475	.4526	.4557	.4578	.4593	.4603	.4611	.4617	.4650
2	6	--	--	--	.4366	.4614	.4737	.4807	.4851	.4879	.4899	.4914	.4925	.4933	.4979
2	7	--	--	--	.4418	.4718	.4870	.4956	.5011	.5047	.5072	.5091	.5105	.5115	.5174
2	8	--	--	--	--	.4761	.4935	.5036	.5099	.5142	.5172	.5194	.5210	.5223	.5293
2	9	--	--	--	--	.4774	.4965	.5077	.5148	.5196	.5230	.5255	.5274	.5288	.5369
2	10	--	--	--	--	--	.4977	.5097	.5174	.5227	.5264	.5291	.5312	.5328	.5418
2	11	--	--	--	--	--	~~.4980~~	.5107	~~.5188~~	.5244	.5283	.5313	.5335	.5352	.5450
2	12	--	--	--	--	--	--	.5110	.5195	.5253	.5295	.5325	.5352	.5369	.5471
2	13	--	--	--	--	--	--	.5111	.5198	.5258	.5301	.5333	.5357	.5376	.5485
3	4	--	--	.1903	.2101	.2162	.2188	.2202	.2210	.2215	.2219	.2221	.2223	.2225	.2232
3	5	--	--	.2434	.2902	.3049	.3112	.3146	.3166	.3178	.3187	.3193	.3197	.3200	.3218
3	6	--	--	--	.3224	.3464	.3570	.3625	.3658	.3679	.3694	.3704	.3711	.3716	.3746
3	7	--	--	--	.33329	.3660	.3807	.3885	.3932	.3962	.3982	.3996	.4007	.4015	.4057
3	8	--	--	--	--	.3746	.3929	.4029	.4088	.4127	.4153	.4172	.4185	.4196	.4251
3	9	--	--	--	--	.3774	.3990	.4107	.4179	.4224	.4257	.4279	.4296	.4309	.4376
3	10	--	--	--	--	--	.4017	.4149	.4230	.4283	.4320	.4346	.4366	.4381	.4460
3	11	--	--	--	--	--	.4025	.4170	.4259	.4318	.4359	.4388	.4410	.4427	.4516
3	12	--	--	--	--	--	--	.4179	.4274	.4338	.4382	.4414	.4438	.4456	.4556
3	13	--	--	--	--	--	--	.4182	.4282	.4349	.4396	.4430	.4455	.4475	.4583
4	5	--	--	.1278	.1662	.1747	.1779	.1794	.1803	.1808	.1812	.1814	.1816	.1817	.1823
4	6	--	--	--	.2271	.2480	.2559	.2597	.2618	.2631	.2639	.2644	.2647	.2649	.2669
4	7	--	--	--	.2478	.2822	.2955	.3019	.3055	.3078	.3092	.3103	.3110	.3115	.3142
4	8	--	--	--	--	.2975	.3162	.3254	.3306	.3338	.3360	.3374	.3385	.3393	.3432
4	9	--	--	--	--	.3030	.3267	.3386	.3453	.3496	.3523	.3543	.3557	.3568	.3620
4	10	--	--	--	--	--	.3316	.3458	.3540	.3592	.3626	.3650	.3667	.3681	.3746
4	11	--	--	--	--	--	.33336	.3496	.3591	.3650	.3690	.3719	.3739	.3755	.3833
4	12	--	--	--	--	--	--	.3514	.3619	.3686	.3731	.3763	.3786	.3804	.3894
4	13	--	--	--	--	--	--	.3520	.3633	.3706	.3756	.3791	.3817	.3837	.3937
5	6	--	--	--	.1313	.1451	.1493	.1511	.1521	.1527	.1530	.1533	.1534	.1536	.1542
5	7	--	--	--	.1713	.2054	.2160	.2206	.2230	.2244	.2252	.2257	.2260	.2262	.2281
5	8	--	--	--	--	.2318	.2499	.2578	.2620	.2644	.2660	.2670	.2676	.2680	.2709
5	9	--	--	--	--	.2415	.2672	.2786	.2847	.2883	.2906	.2922	.2933	.2941	.2979
5	10	--	--	--	--	--	.2755	.2903	.2983	.3030	.3061	.3081	.3096	.3107	.3158
5	11	--	--	--	--	--	.2786	.2965	.3063	.3121	.3159	.3185	.3204	.3217	.3282
5	12	--	--	--	--	--	--	.2996	.3109	.3178	.3222	.3253	.3275	.3292	.3369
5	13	--	--	--	--	--	--	.3007	.3134	.3212	.3263	.3298	.3323	.3342	.3432
6	7	--	--	--	.0934	.1217	.1279	.1303	.1314	.1320	.1324	.1327	.1328	.1329	.1335
6	8	--	--	--	--	.1691	.1850	.1910	.1939	.1954	.1962	.1967	.1970	.1972	.1993
6	9	--	--	--	--	.1863	.2133	.2237	.2287	.2315	.2332	.2342	.2349	.2353	.2382
6	10	--	--	--	--	--	.2268	.2418	.2492	.2533	.2559	.2576	.2587	.2594	.2633
6	11	--	--	--	--	--	.2320	.2516	.2614	.2669	.2703	.2726	.2741	.2750	.2804

* Ordinary RLL values (no charge constraint) are in right column.

ACKNOWLEDGMENT

The authors wish to thank Pat Kocsis for help in preparation the manuscript.

REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, July 1948.

[2] A. M. Patel, "Zero-modulation encoding in magnetic recording," *IBM J. Res. Dev.*, vol. 19, pp. 366–378, July 1975.

[3] A. M. Patel, "Charge-constrained byte-oriented (0, 3) code," *IBM Tech. Disclosure Bull.*, vol. 19, pp. 2715–2724, Dec. 1976.

[4] J. Hong and D. L. Ostapko, "Codes for self-clocking, ac-coupled transmission: aspects of synthesis and analysis," *IBM J. Res. Dev.*, vol. 19, pp. 358–365, July 1975.

[5] T. M. Chien, "Upper bound on the efficiency of dc-constrained codes," *Bell Syst. Tech. J.*, vol. 49, pp. 2267–2287, Nov. 1970.

[6] P. A. Franaszek, "Sequence-state methods for run-length-limited coding," *IBM J. Res. Dev.*, vol. 14, pp. 376–383, July 1970.

Comment on units:

$$C = \log_b \lambda$$

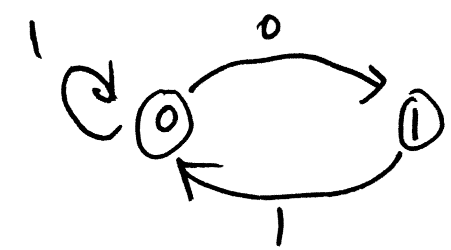
If $b=2$, units are binary digits/symbol
or bits/symbol

If $b=3$, units are ternary digits/symbol

etc. ⋮

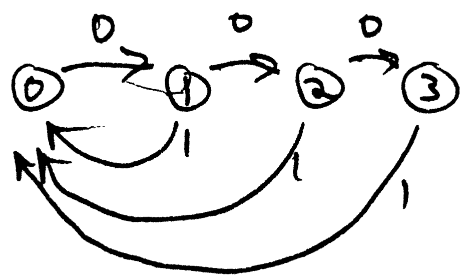
Statistical Properties of DNC

Questions we might ask:



1) In RLL (0,1), how frequently does 0 occur?

2) How often does 0 1 1 0 occur?



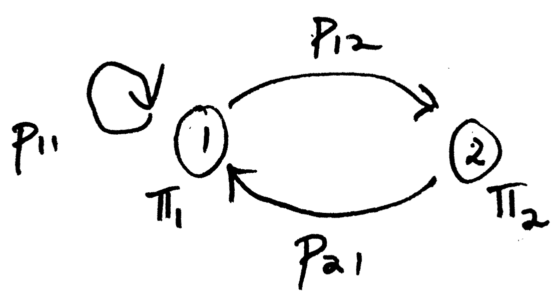
3) In RLL (1, 3), what percentage of runs have length 1 ?
length 2 ?
length 3 ?

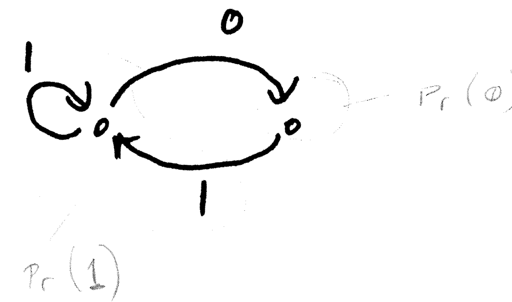
4) In general, how often does a specific string $x_1 \cdots x_N$ of N bits occur ?

Motivation: Timing/gain control, worst case error-rate patterns, synchronization patterns

FSTD's AND MARKOV SOURCES

Assign probabilities π_i to states σ_i and transition probabilities $p_{ij}^{(k)}$ to state transitions $\sigma_i \to \sigma_j$





<u>Objective</u>: Find $\{\pi_i\}$ and $\{p_{ij}\}$ corresponding to results obtained by frequency analysis of all ~~st~~ sequences produced by G

<u>Then</u>: for RLL (0,1)

$Pr(1 \text{ occurs}) = \pi_1$ IF A 1 OCCURS, YOU MUST END UP IN STATE 1

$Pr(0 \text{ occurs}) = \pi_2$ IF A 0 OCCURS, YOU MUST END UP IN STATE 2

$\pi_1 \mid \pi_2 \; \pi_1 \; \pi_1 \; \pi_2 \mid$

$Pr(0110 \text{ occurs}) = \pi_1 \, p_{12} \, p_{21} \, p_{11} \, p_{12}$

<u>Remark 1</u>: a) $P = (p_{ij})$ is a stochastic matrix

$$\sum_j p_{ij} = 1$$

b) $\pi = (\pi_i)$ is the <u>stationary</u> <u>distribution</u>

$$\pi_j = \sum_i \pi_i \, p_{ij}$$

or $\pi P = \pi$

(Left eigenvector of P, with eigenvalue $\lambda = 1$)

<u>Remark 2</u>: For any Markov source where $T^n > 0$, some n, and given $P = (p_{ij})$, then π always exists

SHANNON PROBABILITIES

Let T be the transition matrix for the irreducible DNC S.

Let λ = largest eigenvalue of T (and T^t)

v = corresponding right eigenvector of T

u = corresponding left eigenvector of T

Then:

$$p_{ij} = \frac{v_j}{v_i \lambda} \quad \text{when } t_{ij} \neq 0$$

HANDLES NULL EDGES

$$= 0 \quad \text{otherwise}$$

and

$$\pi_i = \frac{u_i v_i}{\sum_j u_j v_j}$$

Proof: Shannon-Weaver, chapter 1

EXAMPLE: RLL (0,1)

$$T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} ; \quad \lambda^2 - \lambda - 1 = 0$$

$$\lambda = \frac{1+\sqrt{5}}{2} \approx 1.618$$

$\lambda V_2 + V_2 = \lambda^2 V_2$

$V_2(\lambda+1) = \lambda^2 V_2$

Solve: $Tv = \lambda v$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \lambda v_1 \\ \lambda v_2 \end{bmatrix}$$

2.618
2.618

$1 + \lambda = \lambda^2$

$1 + V_1 = \lambda V_1$

Assume $v_2 = 1$, then $\left.\begin{aligned} v_1 + v_2 &= \lambda v_1 \\ v_1 &= \lambda v_2 \end{aligned}\right\} \Rightarrow v_1 = \lambda$

so $v = [\cancel{\lambda \;\; 1}]$ $\begin{bmatrix} \lambda \\ 1 \end{bmatrix}$

Similarly: $uT = \lambda u$ yields a solution

$u = [\lambda \;\; 1]$

$\sum_j v_j u_j = v_1 u_1 + v_2 u_2 = \lambda \cdot \lambda + 1 \cdot 1 = 1 + \lambda^2$

So:

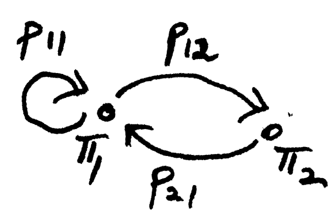
$P_{11} = \frac{v_1}{v_1 \lambda} = \lambda^{-1} \approx .618$

$P_{12} = v_2 / v_1 \lambda = \lambda^{-2} \approx .382$

$P_{21} = \frac{v_1}{v_2 \lambda} = 1$

$\pi_1 = \frac{\lambda^2}{1+\lambda^2} \approx .724$

$\pi_2 = \frac{1}{1+\lambda^2} \approx .276$



POWER SPECTRUM CALCULATION

Constrained sequences: $a_0\, a_1 \cdots a_m \cdots \in S$

Average power density spectrum $\Phi_s(f)$

$$\Phi_s(f) = \lim_{M\to\infty} E_s \left[\frac{\left| \sum_{m=0}^{M-1} a_m D^m \right|^2}{M} \right]$$

where $D = e^{-i2\pi f}$

Wiener - Khinchin Theorem

$$\Phi_s(f) = \sum_{j=-\infty}^{\infty} R_s(j) D^j, \quad \text{where } D = e^{-i2\pi f}$$

and $R_s(j) = j^{th}$ autocorrelation

$= E_s[a_0 a_j]$

Lecture 9: Information Theory (cont.)

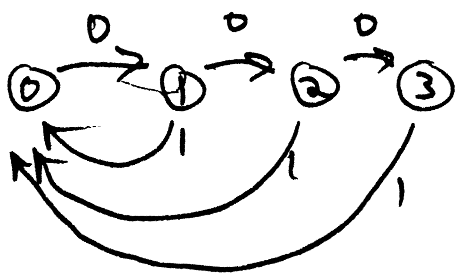
Code Construction

STATISTICAL PROPERTIES OF DNC

Questions we might ask:

1) In RLL (0,1), how frequently does 0 occur?

2) How often does 0 1 1 0 occur?



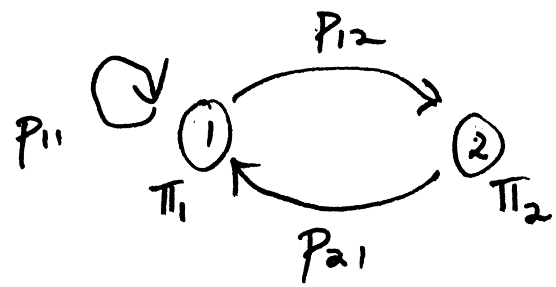
3) In RLL (1, 3), what percentage of runs have length 1 ?
length 2 ?
length 3 ?

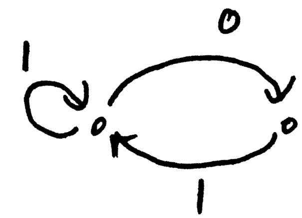
4) In general, how often does a specific string $x_1 \ldots x_N$ of N bits occur?

MOTIVATION: Timing/gain control, worst case error-rate patterns, synchronization patterns

FSTD's and Markov Sources

Assign probabilities π_i to states σ_i and transition probabilities $p_{ij}^{(k)}$ to state transitions $\sigma_i \to \sigma_j$





<u>Objective</u>: Find $\{\pi_i\}$ and $\{p_{ij}\}$ corresponding to results obtained by frequency analysis of all ~~st~~ sequences produced by G

<u>Then</u>: for RLL (0, 1)

$$Pr(1 \text{ occurs}) = \pi_1$$

$$Pr(0 \text{ occurs}) = \pi_2$$

$$Pr(0110 \text{ occurs}) = \pi_1\, p_{12}\, p_{21}\, p_{11}\, p_{12}$$

<u>Remark 1</u> : a) $P = (p_{ij})$ is a stochastic matrix

$$\sum_j p_{ij} = 1$$

b) $\pi = (\pi_i)$ is the <u>stationary</u> <u>distribution</u>

$$\pi_j = \sum_i \pi_i p_{ij}$$

or $\pi P = \pi$

(Left eigenvector of P, with eigenvalue $\lambda = 1$)

<u>Remark 2</u> : For any Markov source where $T^m > 0$, some m, and given $P = (p_{ij})$, then π always exists

SHANNON PROBABILITIES

Let T be the transition matrix for the irreducible DNC S.

Let λ = largest eigenvalue of T (and T^t)

v = corresponding right eigenvector of T

u = corresponding left eigenvector of T

Then:

$$p_{ij} = \frac{v_j}{v_i \lambda} \quad \text{when } t_{ij} \neq 0$$

$$= 0 \quad \text{otherwise}$$

and

$$\pi_i = \frac{u_i v_i}{\sum_j u_j v_j} \leftarrow \text{NORMALIZING FACTOR}$$

Proof: Shannon-Weaver, chapter 1

EXAMPLE: RLL (0,1)

$$T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \; ; \quad \lambda^2 - \lambda - 1 = 0$$

$$\lambda = \frac{1+\sqrt{5}}{2} \approx 1.618$$

GOLDEN RATIO

Solve: $T v = \lambda v$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \lambda v_1 \\ \lambda v_2 \end{bmatrix}$$

Assume $v_2 = 1$ (SCALING FACTOR), then $\left.\begin{aligned} v_1 + v_2 &= \lambda v_1 \\ v_1 &= \lambda v_2 \end{aligned}\right\} \Rightarrow v_1 = \lambda$

$v_2 = 1$

so $v = [\lambda \;\; 1]^T$ RIGHT EIGENVECTOR

Similarly: $uT = \lambda u$ yields a solution

$u = [\lambda \;\; 1]$ LEFT ~~RIGHT~~ EIGENVECTOR

$$[u_1 \;\; u_2] \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \lambda u_1 \\ \lambda u_2 \end{bmatrix}$$

$u_1 + u_2 = \lambda u_1$

$u_1 = \lambda u_2$

$u_2 = 1$

$u_1 = \lambda$

So:

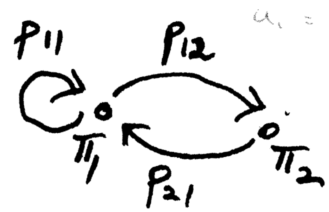
$$p_{11} = \frac{v_1}{v_1 \lambda} = \lambda^{-1} \approx .618$$

$$p_{12} = v_2 / v_1 \lambda = \lambda^{-2} \approx .382$$

$$p_{21} = \cancel{\frac{v_1}{v_1 \lambda}} = 1$$

$$\pi_1 = \frac{\lambda^2}{1+\lambda^2} \approx .724$$

$$\pi_2 = \frac{1}{1+\lambda^2} \approx .276$$



Power Spectrum Calculation

Constrained sequences: $a_0 a_1 \cdots a_m \cdots \in S$

Average power density spectrum $\Phi_s(f)$

POWER

$$\Phi_s(f) = \lim_{M \to \infty} E_s \left[\frac{\left| \sum_{m=0}^{M-1} a_m D^m \right|^2}{M} \right]$$

EXPECTED VALUE

LENGTH OF SEQUENCE

where $D = e^{-i2\pi f}$

Wiener - Khinchin Theorem

$$\Phi_s(f) = \sum_{j=-\infty}^{\infty} R_s(j) D^j, \quad \text{where } D = e^{-i2\pi f}$$

and $R_s(j) = j^{th}$ autocorrelation

$= E_s[a_0 a_j]$

Autocorrelation for dicode sequences

$$R(j) = \pi_2 P_{22}^{j} + (-1)\pi_2 P_{23}^{j} + \pi_3 P_{33}^{j} + (-1)\pi_3 P_{32}^{j}$$

$$P^2 = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \qquad \text{(check!)}$$

$$P^n = \frac{1}{2^n}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad n>1$$

<u>and</u>

$$P^n = P^2 \qquad \text{for } n \geq 2$$

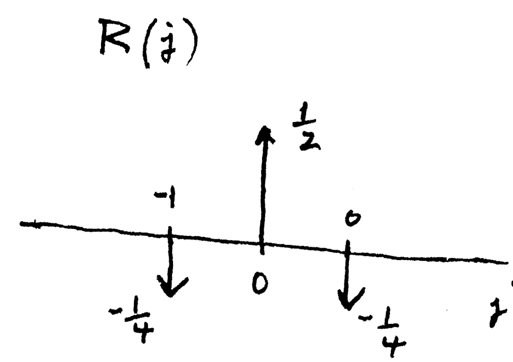
So,

$$R(0) = \frac{1}{4}(1) + \frac{1}{4}(1) = \frac{1}{2}$$

$$R(1) = (-1)\frac{1}{4}\left(\frac{1}{2}\right) + (-1)\frac{1}{4}\left(\frac{1}{2}\right) = -1/4$$

$$R(n) = 0, \quad n \geq 2$$

$$R(-j) = R(j), \quad j \geq 0$$



$$\underline{R(j) = \frac{1}{2}\left[\delta(j) - \frac{1}{2}\left[\delta(j-1) + \delta(j+1)\right]\right]}$$

Fourier Transform of R(j) for dicode

$$R(j) = \frac{1}{2}\left[\delta(j) - \frac{1}{2}\left(\delta(j-1) + \delta(j+1)\right)\right]$$

EINER–KICHAN

Recall

$$\mathfrak{F}\left(\delta(j-j_0)\right) = e^{-i2\pi f j_0}$$

DELAYED IMPULSES

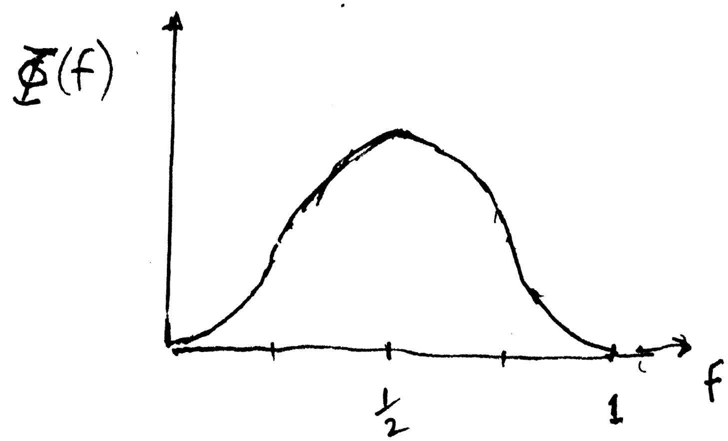
So

$$\mathfrak{F}(R(j)) = \frac{1}{2} - \frac{1}{4}\left[\underbrace{e^{-i2\pi f} + e^{i2\pi f}}_{2\cos 2\pi f}\right]$$

$$= \frac{1}{2}\left[1 - \cos 2\pi f\right]$$

$\cos 2\pi f = \cos^2 \pi f - \sin^2 \pi f$

$$\boxed{\Phi(f) = \sin^2 \pi f}$$



Spectral null at $f = 0$!

DOESN'T TRANSMIT CONSTANT VALUES
NO OUTPUT WITH CONSTANTS

(TRANSFER FUNCTION)

Remark 1: Recall frequency response of 1-D channel

$$\frac{1}{2}(1-D)\Big]_{D=e^{-i2\pi f}} = \frac{1}{2}(1-e^{-i2\pi f})$$

$$= \frac{1}{2}e^{-i\pi f}\left[e^{i\pi f} - e^{-i\pi f}\right]$$

or

$$H(f) = \frac{1}{2}e^{-i\pi f}\left[2i\sin\pi f\right] = i e^{-i\pi f}\sin\pi f$$

MODULUS 1 (CONTRIBUTES PHASE ONLY)

TRANSFER FUNCTION

By the convolution theorem:

SPECTRUM

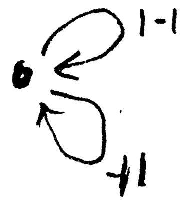
$$\Phi_{out}(f) = \Phi_{in}(f)\cdot|H(f)|^2$$

$\Phi_{in}(f) = 1$ for random data ± 1

$\sin^2\pi f$

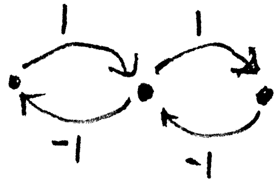
$$\boxed{\Phi_{out}(f) = \sin^2\pi f}$$

Remark 2: Power spectrum of biphase signals is similar



$$\boxed{\Phi_{\text{biphase}}(f) = 2\sin^2 \pi f}$$

Matched spectral null at $f=0$ (DC) !!



$+1 \quad -1$

$-1 \quad +1$

$\Phi_{\text{BIPHASE}}(0) = 0$

$\Phi_{\text{BICODE}}(0) = 0$

Exercise: Compute $\Phi_{\text{biphase}}(f)$ from $R_{\text{biphase}}(j)$

General formula for (d,k) constraints (see [G-H-S]):

Let $\lambda = 2^{\text{cap}(d,k)}$ (largest eigenvalue of $T_{d,k}$).

Then:

$$\Phi_{d,k}(f) = p(1)\left[1 - G(D)\,G(D^{-1})\right] \Big/ \left(1+G(D)\right)\left(1+G(D^{-1})\right)$$

where $G(D) = \sum_{j=d+1}^{k+1} \lambda^{-j} D^{j}$

and $p(1) = 1 \Big/ \sum_{j=d+1}^{k+1} j\,\lambda^{-j}$

Block Codes and Sequence-State Codes

References: Freiman and Wyner, Tang and Bahl } block codes; Franaszek, Horiguchi-Morita } sequence-state codes

Also: Lectures 4-5 from EE458, Winter 1986

Block coding objective: For RLL(d,k) constraint, and codeword length n, find the largest list L_n of freely concatenable codewords.

Note: If $|L_n|$, the number of codewords in L_n, satisfies

$$|L_n| \geq 2^m$$

then a rate $m:n$ block code is possible

Block Codes · for Magnetic Tape

1. Phase Encoding (PE) $\frac{1}{2}(0,1)$

   For RLL(0,1), $|L_2| = 2 \geq 2^0$ $\longrightarrow$

2. GCR (Group Code Recording) $\frac{4}{5}(0,2)$

   For RLL(0,2), $|L_5| = 17 \geq 2^0$ $\longrightarrow$ IBM 3420 (9-track standard)

3. $\frac{8}{9}(0,3)$ code

   For RLL(0,3), $|L_9| = 293 \geq 2^0$ $\longrightarrow$ IBM 3480 (18-track standard)

Comparison of Tape Codes

PE $\frac{1}{2}(0,1)$: 1 0 1 1 1 0 1 0 1 0 1 0 1 1 1 1

GCR $\frac{4}{5}(0,2)$: 1 1 1 0 1 1 0 0 1 1

$\frac{8}{9}(0,3)$: 0 1 0 0 1 0 0 1 1

Data: 0 1 0 0 0 0 1 1

The *detection window* T_d for the code is

$$T_d = \frac{m}{n} T \text{ in seconds}$$

and the *clock rate* is (m/n) (1/T).

Table 9-4 lists several group and block codes. They represent a complete re-coding of the original code, and some of the rules for the re-coding are rather complex. The coding is in essence done by breaking the original code up in groups of n bits and mapping them into m bits.

The pattern of the m bits can be *predetermined* and stored in a library, with a pattern for each combination of the incoming n bits. Such an arrangement is made for the popular *4/5 code* or *GCR code,* and the conversion table is shown in Fig. 9-30. This code is one of many *IBM* originated codes that paved the way for today's high packing densities of 6,250 bits per inch, and higher.

Basically, GCR uses the NRZI format for "1"s and "0"s, but a restriction is added: there can be no more than two "0"s in sequence (k=2, d=0). This quarantees that flux changes occur at least once in every three bit cells, and the variable-frequency clock need only be able to lock onto three pulses, corresponding to a succession of "1"s, alternate "1"s and "0"s, and a "1" followed by two "0"s.

The GCR-code is also advantageous when error detection and correction is considered. The reader is referred to *Ringkjφb's* paper for details.

Fig. 9-30. 4/5 GCR group code library.

DATA 1234/5678		STORAGE/RECORD* 12345/678910
0000		11001
0001		11011
0010		10010
0011		10011
0100		11101
0101		10101
0111	<0110	10110
1000		10111
1001		11010
1010		01001
1011		01010
1100		01011
1101		11110
1110		01101
1111		01110
		01111

*SUBGROUP BIT POSITIONS

11100
10100

Block Codes - Pros and Cons

Advantages: - Table look-up for encoder and decoder

- ROM implementation
- Simple and inexpensive (in some cases)

Disadvantages: - For rate m/n, close to capacity C, codeword length might have to be large ($mk : nk$, k large)

- Large codeword length n increases error propagation
- Look-up table size might be impractical

<u>Example</u>: Rate $\frac{1}{2}$ (1,3) block code requires codeword length <u>at least</u> 20 bits long!
Code will be 10:20.

SEQUENCE-STATE CODES

Objective: Reduce code complexity (e.g. codeword length) for a given code rate $\frac{m}{n}$ by using a FSM encoder derived from FSTD for constraint.

Example: $\frac{1}{2}$ RLL(1,3)

a) Simplest block code is no better than 10:20
Error propagation can be as much as 10 bits
(depending on data assignment)

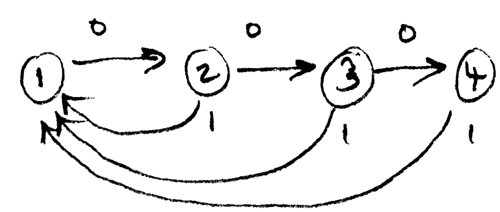
b) MFM code (sequence-state) is 1:2
Error propagation is ≤ 1 bit
FSM encoder requires only 2 states.

CONSTRUCTION MFM
$\frac{1}{2}$ (1, 3)

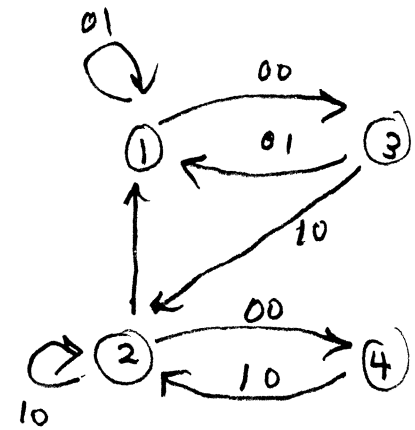
Cap (1,3) = .5515

Target rate = $\frac{1}{2}$; try for 1:2

S



S^2



$$T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

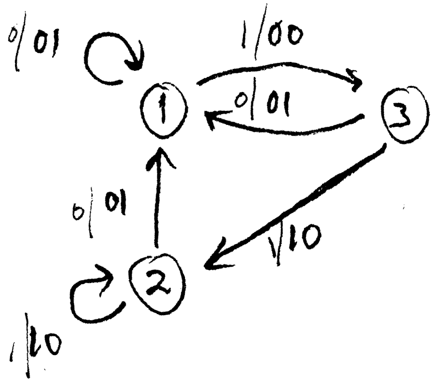
$$T^2 = \begin{array}{c} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{l} 2 \\ \cancel{3}\ 2 \\ 2 \\ 1 \ \leftarrow \end{array}$$

Eliminate state ④ in S^2 (row 4 and column 4 of T^2)

Resulting subgraph has matrix with all rowsums exactly 2

$\Rightarrow$ rate 1:2 is possible !

MFM (cont.)



Encoder table

State \ Next	1	2	3
1	1/01	–	0/00
2	1/01	0/10	–
3	1/01	0/10	–

Note: a) state-independent assignment of data to codewords

$\Rightarrow$ block decoder

01 $\rightarrow$ 1		0
00 $\rightarrow$ 0		1
10 $\rightarrow$ 0		1

b) States 2 and 3 are identical in terms of outgoing edges

$\Rightarrow$ "merge" into single combined state 2^*

(outgoing edges same as 2 and 3
incoming edges same as edges coming into 2 or 3

	1	2^*
1	1/01	0/00
2^*	1/01	0/10

Construction of Sequence-State Codes

1. For DNC S, compute capacity $Cap(S)$
2. Choose rate $\frac{m}{n} \leq C$
3. Set $k=1$ (for $mk:nk$ code)
4. Compute S^{kn} and T^{kn}
5. Find subset of states (principal states) such that the rowsums of the corresponding submatrix of T^{kn} exceed 2^{km}. If none exists, set $k \leftarrow k+1$ and return to step 4, otherwise proceed to step 6
6. Assign 2^{km} data words (km-tuples) to 2^{km} codewords outgoing from each state in $\mathcal{S}$ to get a rate $km:kn$ code

Note: state-independent assignment is always possible for (d,k) sequence-state codes [Franaszek]

76

Remark 1: Block codes are obtained by finding the common intersection of the codeword lists corresponding to the principal states

Remark 2: The row sum condition will be satisfied for T^{km} when the following inequality holds:

$$T^{km} v \geq 2^{km} v$$

where

$$v_i = \begin{cases} 1 & \text{if state } i \in \mathcal{P}\text{, the set of principal states} \\ 0 & \text{if not} \end{cases}$$

VARIABLE LENGTH CODES

Motivation RLL(2,7), $C = .517$

The simplest rate $\frac{1}{2}$ sequence-state code (fixed-length) has a 17:34 structure (i.e. the smallest n for which a set of principal states exists is $n = 34$)

Idea Use variable length codewords to reduce maximum codeword length as well as the number of codewords

EXACTLY 1 WAY TO PARSE DATA FROM THIS LIST

Example: IBM (2,7) code

Data	Codeword
11	0100
10	1000
011	000100
010	001000
000	100100
0011	00100100
0010	00001000

Full prefix-free list

SOLVES MIS-FRAMING PROBLEM

codeword boundaries marked by patterns: - ENDING PATTERN

.01.00.

.10.00

VARIABLE LENGTH CODE CONSTRUCTION

Analogous to fixed-length algorithm.

Key ingredient: Kraft-McMillan theorem

If codewords starting from state σ in S^n have lengths $\{n_1, n_2, \ldots, n_\ell\}$ satisfying

$$\sum_{i=1}^{\ell} 2^{-n_i/n} \geq 1,$$

then a full prefix-free list of data words can be assigned to some subset of the codewords in a one-to-one manner.

Reference: EE 458 course notes, Winter 1986, Lectures 4 & 5.

Sliding Block Code Construction

Problem: Sequence-state methods (fixed and variable length) may require use of codewords from S^{kn}, for large k

Objective: Find a code construction method which, for any rate $\frac{m}{n} < C$, produces a rate $m:n$ code with finite-state encoder and sliding block decoder.

Solution: [Adler-Coppersmith-Hassner 1983]

If S has finite memory, then such a code can be systematically found for any rate $\frac{m}{n} \leq C$.

equality

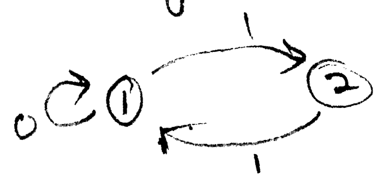
Defn: S has finite memory if it can be represented by a FSTD such that any code sequence

$$x_{-M} \ldots x_{-1}\, x_0\, x_1 \ldots x_A$$

uniquely determines the edge which produced the symbol x_0
(M = memory, A = anticipation).

Example 1: RLL (d,k) has finite memory, with $M = k-1$, $A = 0$
(Exercise)

Example 2: "Even" system has infinite memory



The sequence $\underbrace{111 \cdots 111}_{L}$ is generated

by 2 state sequences which disagree everywhere.

$1212 \ldots 12 \ldots$

or $2121 \ldots$

41

Lecture 10: Variable length codes

Sliding block codes.

PREFIX FREE LISTS

Defn: A finite set of (variable-length) strings over an alphabet $\mathcal{A}$ is prefix free if no string in the list is a prefix of another string in the list

Example: $\mathcal{A} = \{0, 1\}$

$L_1 = \{0, 10\}$ is prefix free

$L_2 = \{0, 01, 001\}$ is not prefix free

Notation: Let L^n denote the strings obtained from concatenation of n strings in L

Example: $L_1^2 = \{00, 010, 100, 1010\}$

$\{0, 10\} \quad \{0, 10\}$

UNIQUE DECODABILITY

Defn: The set L is uniquely decodable (U.D.) if every string in L^n arises in only one way as a concatenation of words in L. TRUE FOR ALL n

$001 \to 001$
$001 \to 0|01$

Example: L_1 is U.D.; L_2 is not U.D. $[001 = 0.01]$
PREFIX FREE

Lemma: A prefix-free list is U.D.
(but not vice versa!)

Defn: A prefix-free list L over the alphabet A is full if every string over A can be decomposed as a concatenation of words in L (uniquely)

Examples: $A = \{0,1\}$.
A^n is full, pre free for all $n \geq 1$.

PARSE 0

0 1 □

WAIT FOR NEXT SYMBOL TO PARSE

$2^{-1}+2^{-2}+2^{-2}=1$

$2^{-1}+2^{-2}\neq 1$

Examples: $L=\{0,10,11\}$ is full, prefix free

$L=\{0,10\}$ is prefix free, but **not** full.

11 CANNOT BE PARSED

$\{1 \;\; 01 \;\; 00\}$ IS FULL, PREFIX FREE

01 00 1

Lemma: Let L be a prefix free, binary list, $L=\{c_1,\dots,c_k\}$, with $|c_1|=m_1,\dots |c_k|=m_k$. The L is full if and only if:

$$\sum_{i=1}^{k} 2^{-m_i}=1$$

FULL BINARY TREE

Remark: For any U.D. list L, the word lengths $m_1,\dots,m_k$ must satisfy:

$$\sum_{i=1}^{k} 2^{-m_i}\le 1 \qquad (*)$$

BINARY PREFIX FREE LIST

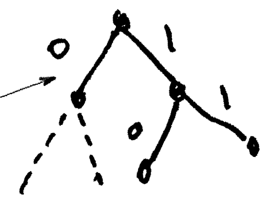
Theorem: Let $m_1,\dots,m_k$ be a set of integers satisfying $(*)$. Then there exits a binary, prefix free list $L=\{c_1,\dots,c_k\}$ such that $|c_i|=m_i$, for $i=1,\dots,k$.

Proofs: See handout

Example: $\{\overset{0}{n_1}, \overset{10}{n_2}, \overset{11}{n_3}\} = \{1, 2, 2\}$

$\sum_{i=1}^{3} 2^{-n_i} = \frac{1}{2} + \frac{1}{4} + \frac{1}{4} = 1$

FULL & PREFIX FREE



$L = \{0, 10, 11\}$

Example: $\{\overset{0}{n_1}, n_2, n_3\} = \{1, 1, 2\}$

$\sum_{i=1}^{3} 2^{-n_i} = 1.25$

> 1

$\frac{1}{2} + \frac{1}{2} + \frac{1}{4}$

$\Rightarrow$ no prefix free list with these word lengths exists.

In fact, no U.D. list exists.

VARIABLE LENGTH SEQUENCE STATE CODES
(CONSTRUCTION METHOD)

For rate m/n binary code, find a set of principal states $\mathcal{J} = \{\sigma_1, \sigma_2, \ldots, \sigma_t\}$, and for each state $\sigma_i \in \mathcal{J}$, find a list of codewords $L(\sigma_i) = \{c_{i1}, c_{i2}, \ldots, c_{ik_i}\}$ with codeword lengths

$$n(\sigma_i) = \{n_{i1}, n_{i2}, \ldots, n_{ik_i}\},$$

where $n_{ij} = l_{ij}\, n$, such that:

HOW MANY BASIC CODEWORDS

1) Each codeword in $L(\sigma_i)$ terminates at some state in $\mathcal{J}$

2) $$\sum_{j=1}^{k_i} 2^{-l_{ij} m} = 1$$ FULL PREFIX FREE

3) If a codeword path reaches a state $\sigma \in \mathcal{J}$ at time ln, then the codeword must terminate there

NO PROBLEM FRAMING CODEWORDS

Remarks:

a) Conditions 1) and 2) guarantee that encoding is possible.

(Use a FULL prefix free binary list with word lengths $\{\ell_{ij}m\}_{j=1}^{k_i}$

b) Condition 3) ensures that each $L(\sigma_i)$ is prefix free, and if the constraint S has finite memory, decoder error propagation is limited (i.e. no "mis-framing" can occur).

c) If no such set of principal states exists, then the "basic word length" m should be increased to a multiple of m, until the procedure succeeds

GO TO NEXT HIGHER POWER 3, 6, 9 / 6, 12, 18 MAKING WORDS LONGER

EXAMPLES

1. Reference : Franaszek, in Information and Control

$S = RLL(\overset{d}{2}, \overset{k}{\infty})$

$$T = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$g(\lambda) = \lambda^3 - \lambda^2 - 1$

$\lambda_{max} \approx 1.4656$

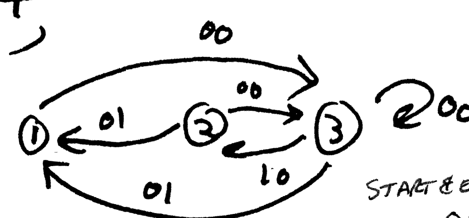
$C = \log_2 \lambda_{max} \approx .5515$ (same as RLL(1,3))

GIVEN: <u>Rate 1/2 Code</u>

Fixed length : minimum code word length required = 14, giving rate 7 : 14 code.

Variable length : Set $\mathcal{J} = \{\sigma_3\}$ (PRINCIPAL STATE SET) in S^2

T^{14}



$L(\sigma_3) = \{00, 0100, 1000\} \longleftarrow$ Prefix free !

START & END AT 3

STATE 3 00 3 2

0100 3→2→3

1000

$\underline{m} = \{2, 4, 4\}$

so $\underline{\ell} = \{1, 2, 2\}$

~~010100~~

10 01 00

(71)

EXAMPLES

1. (cont.)

Reduces to block code

Data	Code
0	00
10	0100
11	1000

ENOUGH TO ASSIGN PREFIX FREE LIST

DON'T NEED 10 01 00

USE SUMMATION = 1 TO SELECT CODES

USE 00 TO DELIMIT CODE WORDS

Note: "00" marks word endings

Sample encoding/decoding

Data: 0 . 0 . 1 . 0 . 1 . 1 . 1 . 0

Framed: $\underbrace{0}$. $\underbrace{0}$. $\underbrace{1\ 0}$. $\underbrace{1\ 1}$. $\underbrace{1\ 0}$ . . .

LOOKUP TABLE

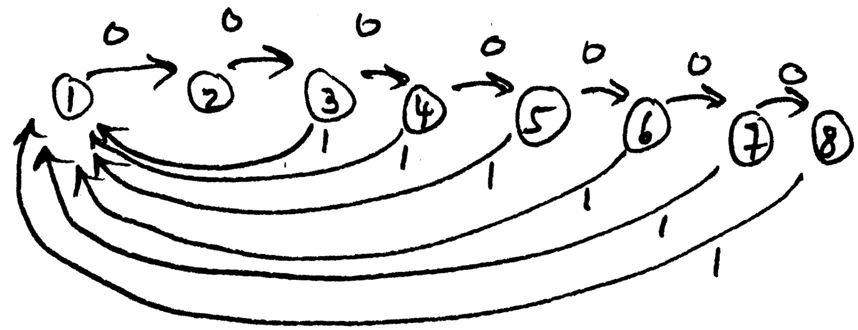
Encoded: $\underbrace{00}$. $\underbrace{00}$. $\underbrace{0100}$. $\underbrace{1000}$ . . . .

Decoded: 0 . 0 . 10 . 11 . . . .

EXAMPLES

2. Reference: Franaszek patent.

$S = RLL(2,7)$



$C \cong .517$

<u>Rate ½ code</u> MULTIPLE OF 2 IN LENGTH

RATE 1/3 – MULTIPLES OF 2 IN LENGTH

Fixed length: minimum codeword length = 34, giving 17:34 code

③ ④ LOOK BUSY

1 STATE DOES NOT GIVE ENOUGH CODEWORDS SO THAT $\Sigma = 1$

TRY 2 STATES

Variable length: $\mathcal{T} = \{\sigma_3, \sigma_4\}$

$L(\sigma_3) \cap L(\sigma_4) =$

100/3 ← REMOVED TO MAKE PREFIX FREE

0100/3

1000/4

000100/3

001000/4

100100/3

00100100/3

00001000/4

$\Sigma < 1$ USE MORE PRINCIPAL STATES

$\Sigma > 1$ REMOVE CODEWORDS

$\underline{m} = \{4, 4, 6, 6, 6, 8, 8\}$

$\underline{l} = \{2, 2, 3, 3, 3, 4, 4\}$

$\sum 2^{-l_i} = 1$

2. (cont.)

Reduces to block code!

Data	Code
11	0100
10	1000
011	000100
010	001000
000	100100
0011	00100100
0010	00001000

	END SEQUENCES
00	0100
10	~~0100~~ 1000

Implementation issues: see handout

SLIDING BLOCK CODE CONSTRUCTION

(STATE SPLITTING)

ADLER, COPPERSMITH, HASSNER

Problem: Sequence-state methods (fixed and variable length) may require use of codewords from S^{kn}, for large k

Objective: Find a code construction method which, for any rate $\frac{m}{n} < C$, produces a rate $m:n$ code with finite-state encoder and sliding block decoder.

Solution: [Adler-Coppersmith-Hassner 1983]

If S has finite memory, then such a code can be systematically found for any rate $\frac{m}{n} \leq C$.

equality

POSSIBLE TO DESIGN 100% EFFICIENT CODES

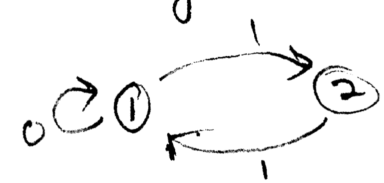
Defn: S has <u>finite memory</u> if it can be represented by a FSTD such that any code sequence

$$\overbrace{x_{-M} \cdots x_{-1}}^{\text{PREVIOUS BITS}} \; x_0 \; \overbrace{x_1 \cdots x_A}^{\text{FUTURE BITS}}$$

uniquely determines the edge which produced the symbol x_0
(M = memory, A = anticipation).

Example 1: RLL (d,k) has finite memory, with $M = k-1$, $A = 0$
(Exercise)

Example 2: "Even" system has <u>infinite</u> memory



The sequence $\underbrace{111\cdots111}_{L}$ is generated

by 2 state sequences which disagree everywhere.

1212 . . . 12 . . .
or 2121 . . . 21 . . .

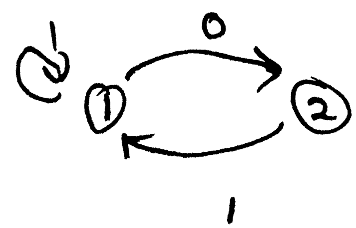
} DOES NOT UNIQUELY DETERMINE THE EDGE

Approximate Eigenvectors

Defn: Let S be a constrained system represented by FSTD G, with state-transition matrix T. Suppose $m/n \le Cap(S)$. An approximate eigenvector of S for rate m/n is an integer, non-negative vector v satisfying

$$T^n v \ge 2^m v.$$

Example: $S = RLL(0,1)$



$$T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \qquad v = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

$2/3 < Cap(S) = .694\ldots$

$m = 2$

$n = 3$

$$T^3 \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \overset{T^3}{\begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}} \overset{v}{\begin{bmatrix} 2 \\ 1 \end{bmatrix}}$$

$$= \underset{T^3 v}{\begin{bmatrix} 8 \\ 5 \end{bmatrix}} \underset{\ge}{\ge} \overset{2^m}{4} \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

$T^3 v \ge 2^m v$

So $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$ is an approximate eigenvector for rate $2/3$

SEQUENCE - STATE
- NEED TO GO TO HIGHER POWER

STATE - SPLITTING

MOTIVATION: If $v = [1, 1, \ldots, 1]$ is an approximate eigenvector for rate m/n, and S has finite memory, then row sums are $\geq 2^m$, implying that a rate $m:n$ code is possible.

IDEA: If $v = [1 1 \cdots 1]$ is not an approximate eigenvector, then modify the FSTD description of S^n until $v = [1 1 \cdots 1]$ is an approximate eigenvector of the modified state transition matrix for S^n.

How? "Split" states whose corresponding eigenvector component > 1, starting with state having largest component, and reducing eigenvector components in the process. Continue until $v = [1 1 \cdots 1]$ works ☺

STATE-SPLITTING ALGORITHM

1. Find an approximate eigenvector for rate m/n:

$$T^n v \geq 2^m v$$

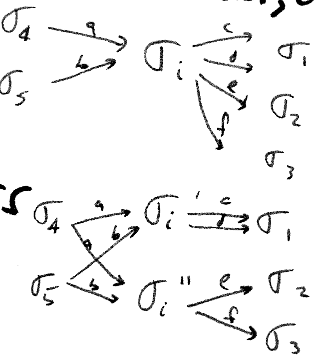
(see handout for details on this step)

2. IF $v = [1 1 \cdots 1]$ works, construct a sequence state code as usual (eliminating states with component 0), since rowsums $\geq 2^m$.

3. If not, split a state having the largest eigenvector component, σ_i, as follows

[see Marcus and Adler, et al]

   a) Partition the outgoing edges from σ_i into disjoint subsets E_j, $j=1,\ldots,k$. with total edge weights $w_j 2^m$, where $w_j \geq 1$ and $\sum_{j=1}^{k} w_j = v_i$. (Put excess edges into E_k).



   b) Split σ_i into "offspring" states σ_i^j, $j=1,\ldots,k$ with outgoing edges E_j and same incoming edges as σ_i, and components w_j.

4. Let $\hat{T}$ be the state-transition matrix for the new FSTD and go to step 2.

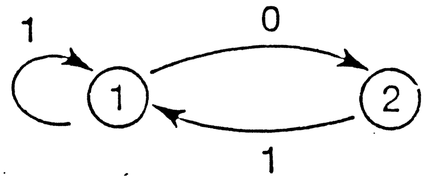
Remark: The state-splitting procedure must terminate, since eigenvector components are being reduced at each pass. Eventually, the components are all reduced to 1.

Remark: The final number of states in the FSTD will be

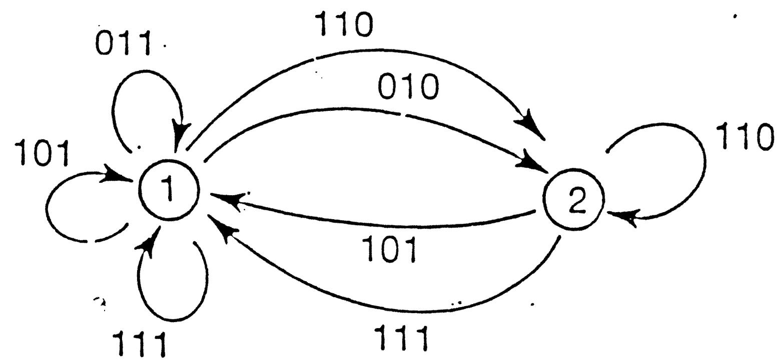
$$\sum_{i=1}^{\#\text{states}} v_i ,$$

but often states can be merged at the end to simplify the encoder.

Construction of 2/3 (0,1) Code



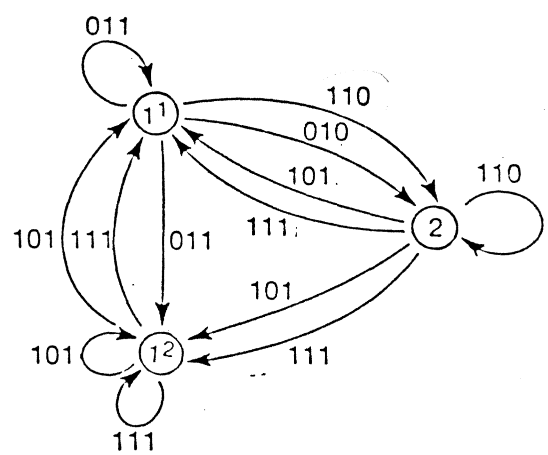
State diagram G for (0,1)



G^3

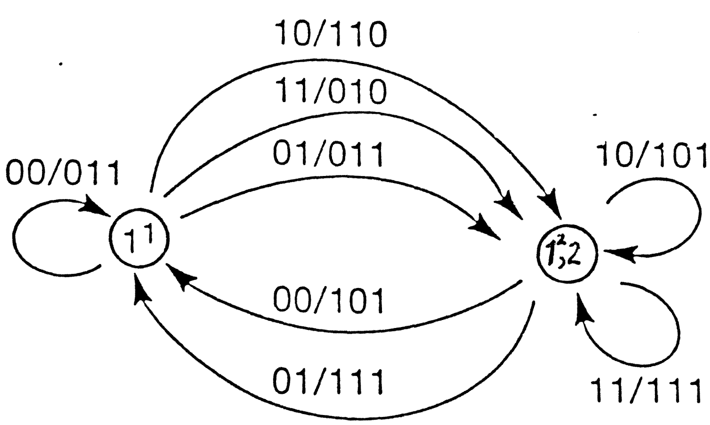
$$T^3 v = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \geq \begin{bmatrix} 8 \\ 4 \end{bmatrix} = 2^2 v$$

Eigenvector inequality



Split graph

LOOK AHEAD 1 EDGE TO MAKE $1^1, 1^2$ DETERMINISTIC



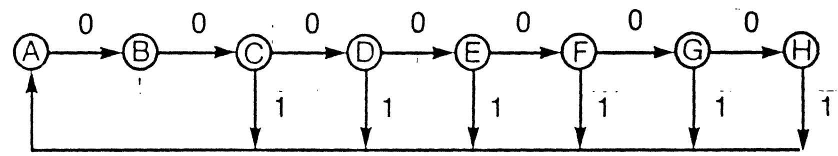
Encoder

	{101,111}	{011,110,010}
011	01	00
110	10	--
010	11	--
101	10	00
111	11	01

Decoder

Sliding Block Code Algorithm

- Generates new graph representation of (D,K) constraints
- Finite-state encoder based on new graph states
- Example: 1/2(2,7)



Splitting Vector [2 3 4 4 3 3 1 1]

$T^2 v \geq 2v$

State-Splitting

21 states
Two 2-bit codewords/state

State-Amalgamation

7 states
Two 2 bit codewords/state

Data Assignment

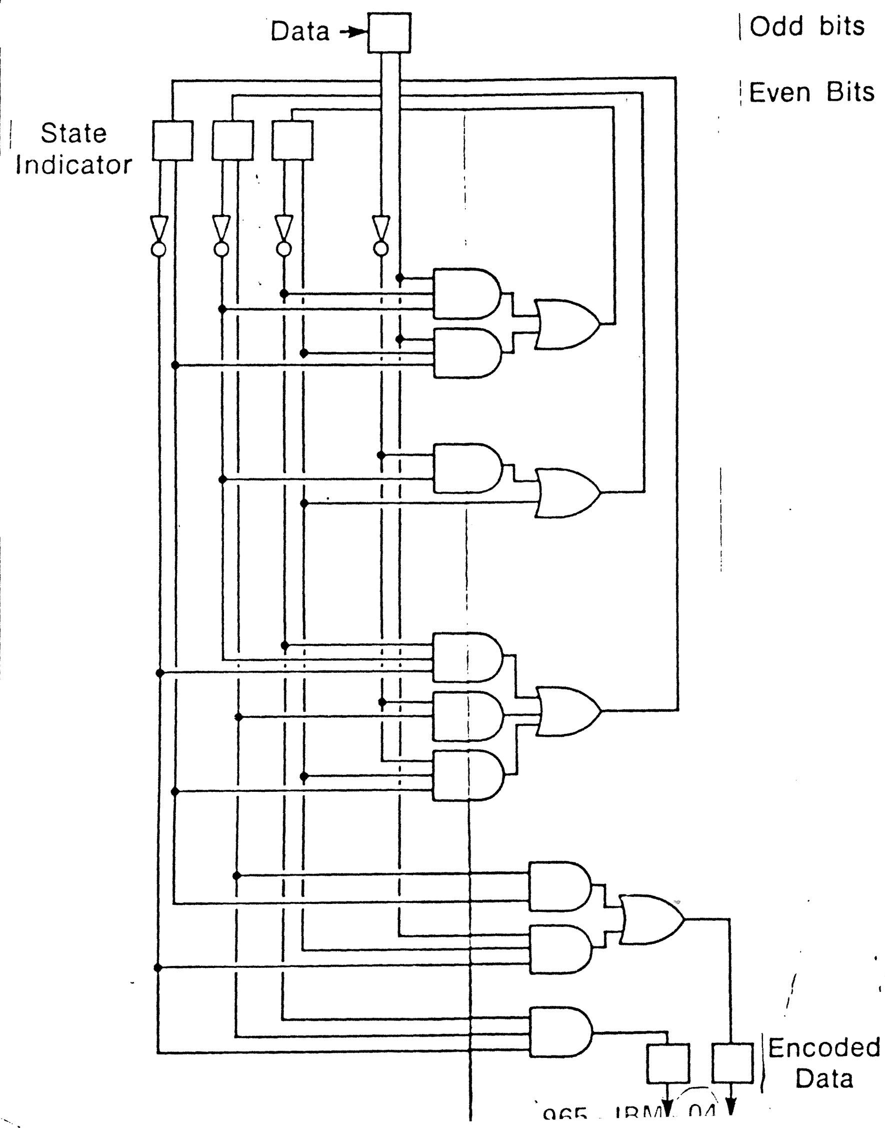
Encoder/Decoder Logic
for (2,7) code

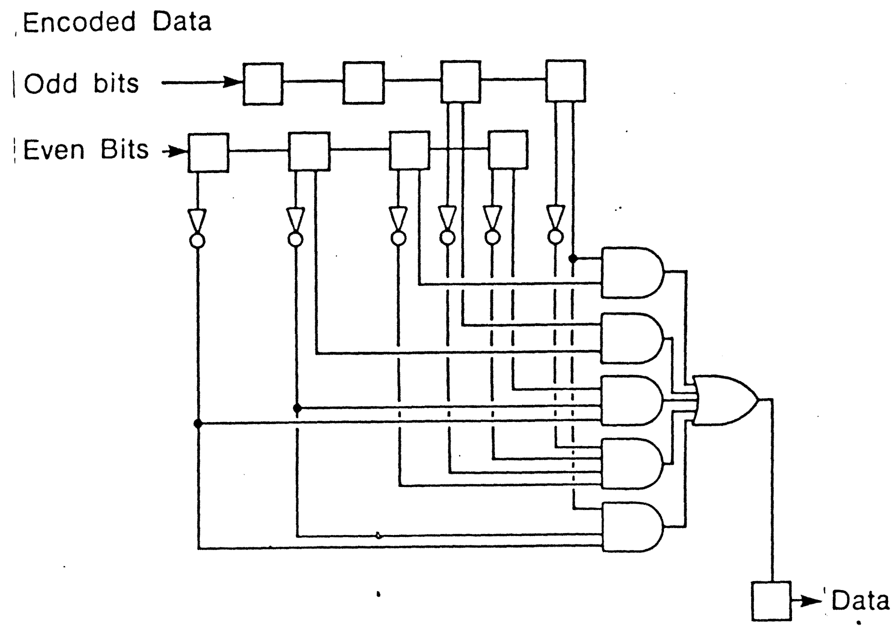
(57)

State / Data	A^1	A^2	B	C^1	C^2	C^3	E
0	00/C^1	00/B	01/A^2	10/A^2	00/B	00/C^1	00/C^1
1	00/C^3	00/C^2	01/A^1	10/A^1	10/B	00/E	10/B

Encoder logic circuit

Decoder logic circuit





WINDOW = 8 BITS

4 USER BITS OF ERROR PROP.

EXAMPLE

Rate 2/3, RLL(1,7) [Adler-Hassner-Moussouris]

Represent S^3 as follows:

$$
\begin{array}{c|ccccc}
 & 000 & 001 & 010 & 100 & 101 \\
\hline
000 & 0 & 1 & 1 & 1 & 1 \\
001 & 1 & 1 & 1 & 0 & 0 \\
010 & 1 & 1 & 1 & 1 & 1 \\
100 & 1 & 1 & 1 & 1 & 1 \\
101 & 1 & 1 & 1 & 0 & 0
\end{array}
\qquad T
$$

Set $v = [4\ 3\ 5\ 5\ 3]$.

Then: $Tv = 2^2 v = 4v$ (check!)

After 2 rounds of "splitting", and some "merging", the IBM (1,7) code results.

Remark: The IBM (1,7) code is essentially equivalent* to the Jacoby (1,7) code, which is usually described as follows.

- Basic encoding table

Data	Code
00	101
01	100
10	001
11	010

Note: $00.00 \rightarrow \underbrace{101.101}_{\text{violates } (1,7)}$

- Violation substitution table

Data	Code
00.00	101.000
00.01	100.000
10.00	001.000
10.01	010.000

* The two codes produce the same code sequences. Only the data-to-codeword assignment is different

(1,7) Code Implementation

- 98% efficient
- Finite-state fixed-length encoder (2 bits → 3 bits)

Data \ State	A	B	C	D	V
00	101/V	100/A	001/V	010/A	000/A
01	100/V	100/B	010/V	010/B	000/B
10	101/C	100/C	001/C	010/C	000/C
11	101/D	100/D	001/D	010/D	000/D

State A: Previous input = "00" (no violation)
State B: Previous input = "01" (no violation)
State C: Previous input = "10"
State D: Previous input = "11"
State V: Previous input caused "Violation" pattern

- Sliding block decoder with
  error propagation ≦ 5 data bits