

# Digital Communications

Prof. Dr.-Ing. Gerhard Bauch

Institut für Nachrichtentechnik

# Unser Rat



- Machen Sie sich Notizen während der Vorlesung !
- Schreiben Sie Herleitungen an der Tafel mit !
- Arbeiten Sie die Vorlesung nach!
- Gehen Sie zur Übung !
- Rechnen Sie rechtzeitig die Aufgaben selbstständig !
- Prüfen Sie sich immer, ob Sie es wirklich **verstanden** haben !
- Rechnen alter Klausuren genügt **nicht** zum Bestehen der Prüfung.

# Institut für Nachrichtentechnik - Lehrveranstaltungen

Semester

3. Modern Wireless Systems  
Grünheid, WP, 2V, 1Ü



2. Mobile Communications  
Grünheid, WP, 2V, 1Ü



1. Digitale Nachrichtenübertragung  
Bauch, P, 2V, 1Ü, Laborpraktikum



- Informations- und  
Codierungstheorie  
Bauch, WP, 3V, 1Ü



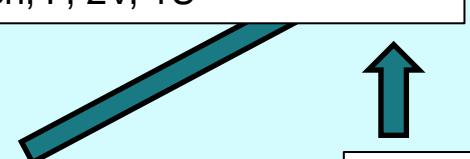
- Digital Signal Processing  
and Digital Filters  
Bauch, P, 3V, 1Ü



5. Grundlagen der Nachrichtentechnik  
Bauch, P, 2V, 1Ü

Bachelor

4. Systemtheorie  
Bauch, P, 3V, 1Ü



- Stochastische Prozesse  
Bauch, P, 2V, 1Ü

- Laborpraktikum  
Grundlagen der  
Elektrotechnik

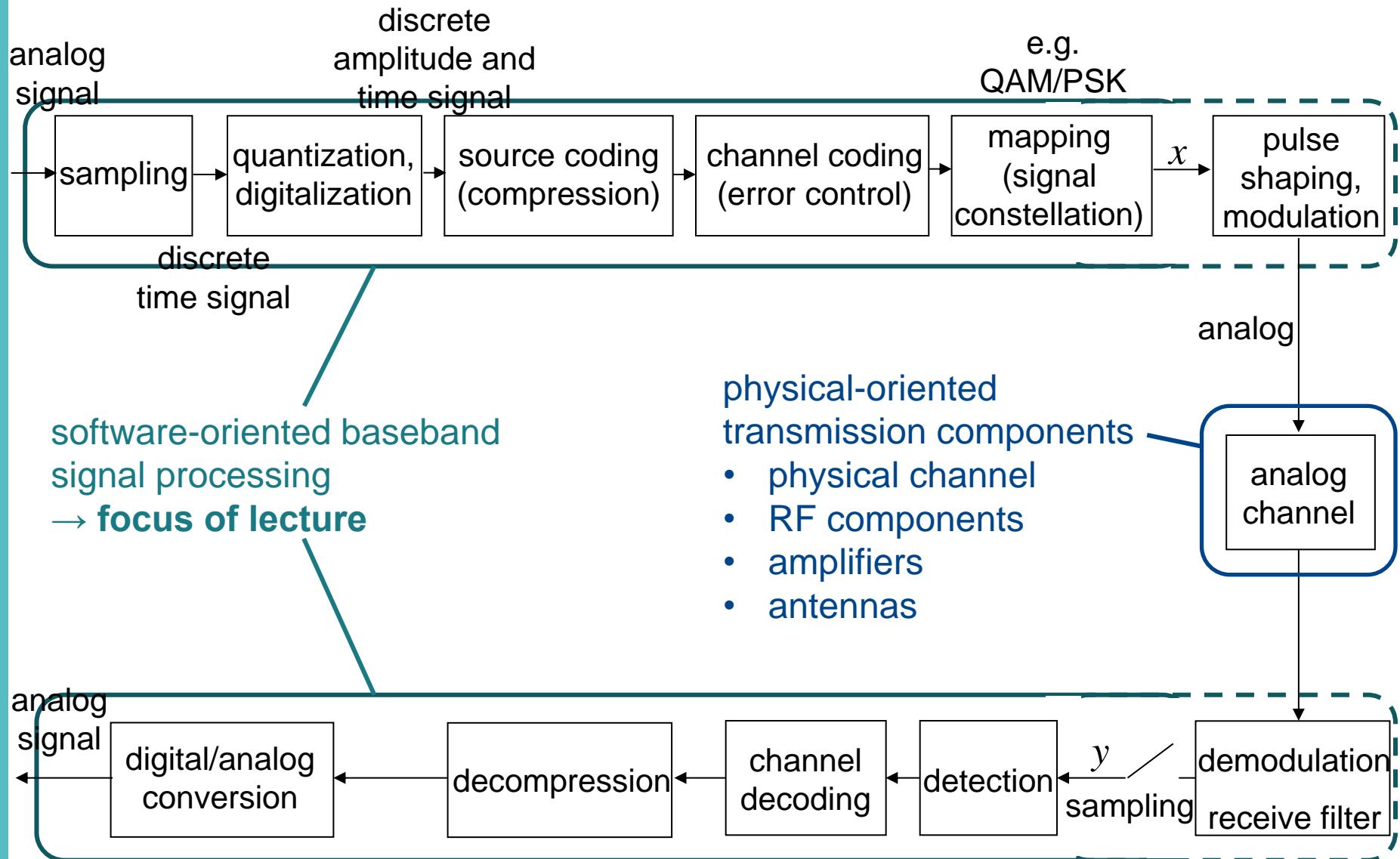
Seminar Informationstechnik

# Literature

**Recommended literature for this course:**

- **K.-D. Kammeyer: Nachrichtenübertragung.** Teubner.
- **P.A. Hoeher: Grundlagen der digitalen Informationsübertragung.** Teubner.
  
- J.G. Proakis, M. Salehi: Digital Communications. McGraw-Hill.
- S. Haykin: Communication Systems. Wiley.

# Digital Communications System



# Digital Communications

Most modern communications systems are based on digital technology. An analog source signal is converted into a digital (i.e. discrete in time and amplitude) signal, usually into a binary signal (i.e. a sequence of binary digits). The binary signal is compressed and encoded by the encoder of a forward error control (FEC) code in order to protect the sequence against transmission errors. Some side information such as an address or training data for synchronisation or channel estimation at the receiver may be added.

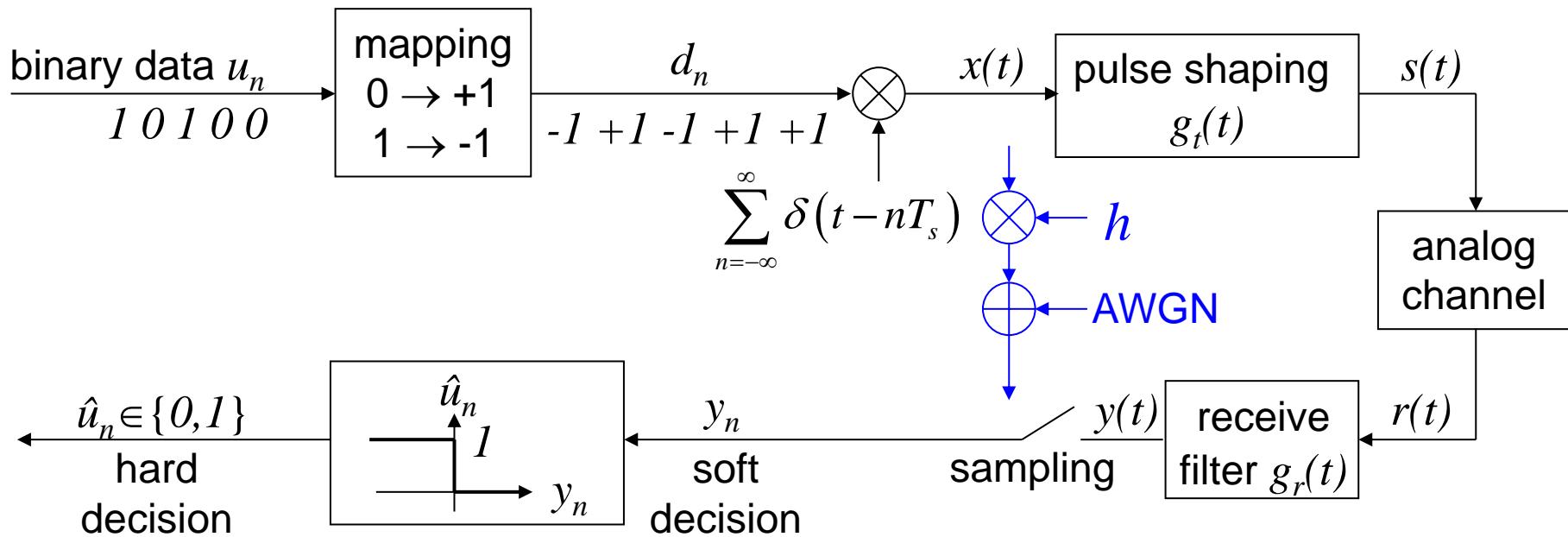
At the transmitter, the message is described with a certain precision by a set of symbols. This set of symbols is known to the receiver.

For transmission over the physical channel, which is analog by nature, the digital sequence has to be converted into an analog transmit signal, which is adapted to the physical transmission medium (e.g. electrical waveforms for wireline transmission, an optical signal for transmission over optical fiber or electromagnetic waves for wireless transmission).

The task of the receiver is the reproduction of the message with a definable reproduction degradation which is due to imperfections in the system.

At the receiver, the distorted analog signal is converted into a digital signal. The quantized signal is estimated by means of digital signal processing and decoding. Finally, the source signal is reconstructed by digital to analog conversion.

# Baseband Transmission



$$x(t) = \sum_{n=-\infty}^{\infty} d_n \delta(t - nT_s); \quad \begin{cases} d_n \in \{-1, +1\}: \text{antipodal signalling} \\ d_n \in \{0, 1\}: \text{on-off signalling} \end{cases}$$

Analog transmit signal:

$$s(t) = x(t) * g_t(t) = \left( \sum_{n=-\infty}^{\infty} d_n \delta(t - nT_s) \right) * g_t(t)$$

$$= \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s)$$

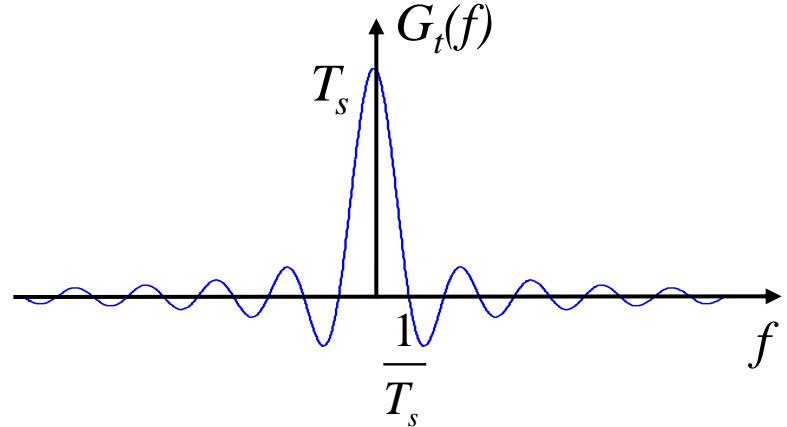
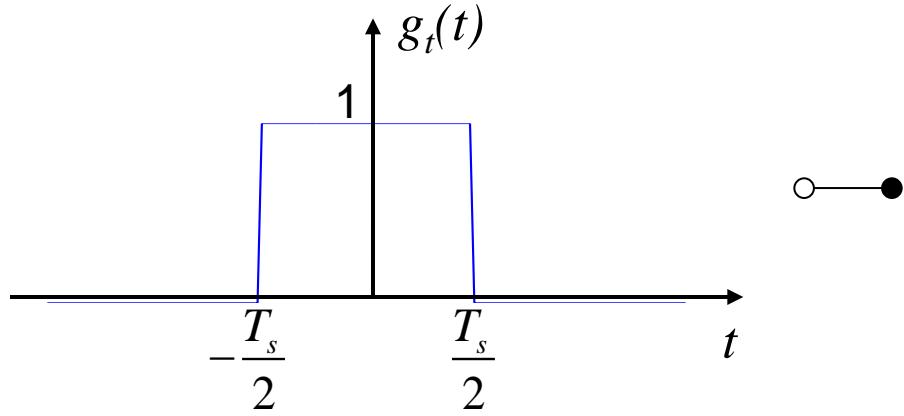
# Pulse Shaping: Time-Limited Pulses

## Non-Return to Zero (NRZ) Pulse (Rectangular Pulse)

$$g_t(t) = \text{rect}\left(\frac{t}{T_s}\right) = \begin{cases} 1 & \text{for } |t| < \frac{T_s}{2} \\ \frac{1}{2} & \text{for } |t| = \frac{T_s}{2} \\ 0 & \text{otherwise} \end{cases}$$

○ —●

$$G_t(f) = T_s \frac{\sin(\pi f T_s)}{\pi f T_s} = T_s \text{ si}(\pi f T_s)$$



# Pulse Shaping: Band-Limited Pulses

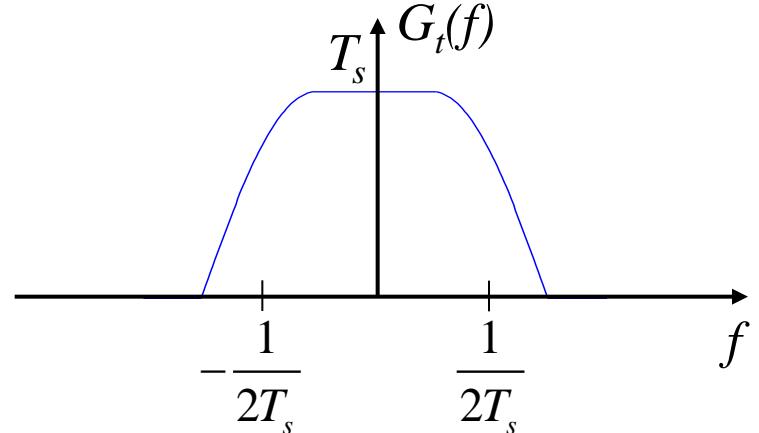
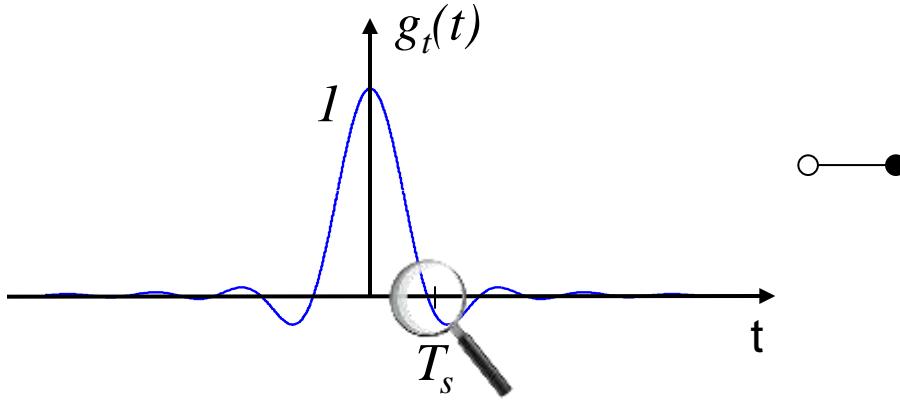
## Square-Root-Raised-Cosine Pulse

$$g_t(t) = \frac{\left(4\alpha \frac{t}{T_s}\right) \cos\left(\pi(1+\alpha)\frac{t}{T_s}\right) + \sin\left(\pi(1-\alpha)\frac{t}{T_s}\right)}{\pi \frac{t}{T_s} \left(1 - \left(\frac{4\alpha t}{T_s}\right)^2\right)}$$

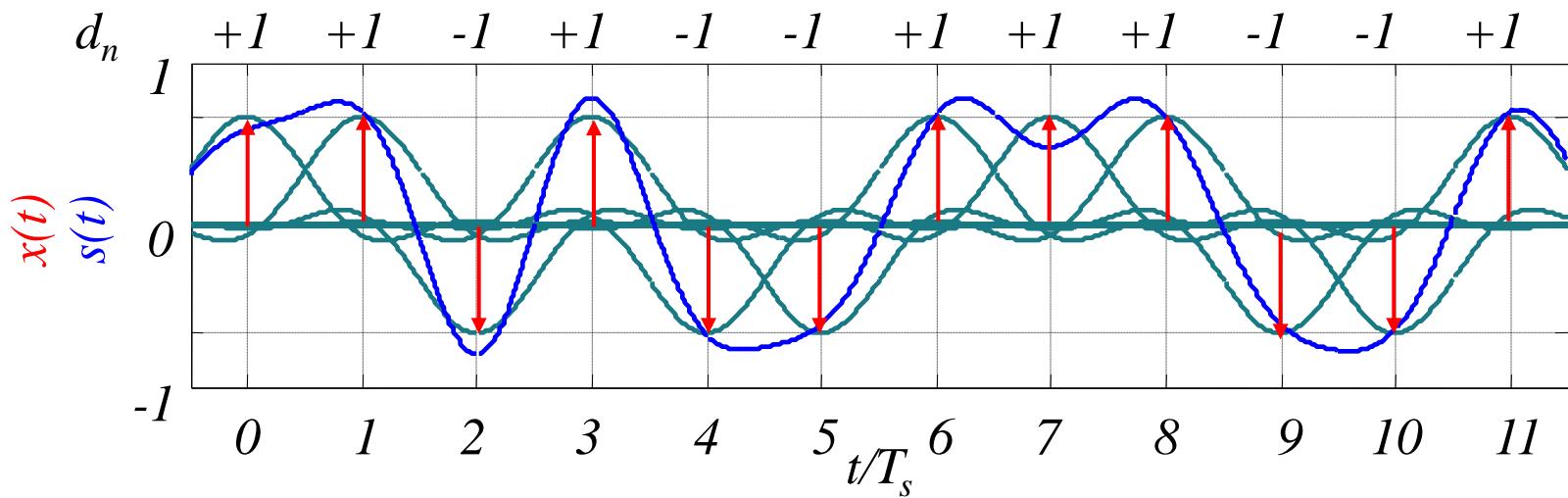
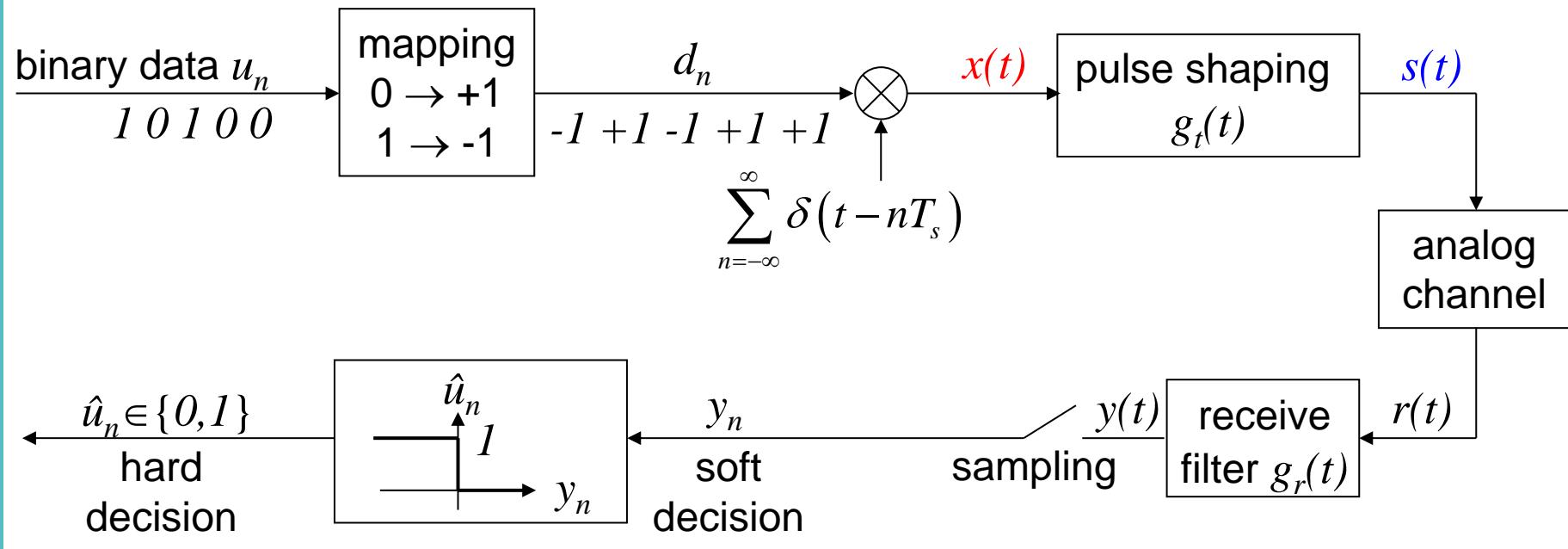
○ —●  $G_t(f) = \begin{cases} T_s & \text{for } |f| \leq (1-\alpha)\frac{1}{2T_s} \\ T_s \cos\left(\frac{\pi}{4\alpha}(2T_s f - 1 + \alpha)\right) & \text{for } (1-\alpha)\frac{1}{2T_s} < |f| \leq (1+\alpha)\frac{1}{2T_s} \\ 0 & \text{otherwise} \end{cases}$

roll-off factor  $\alpha$ :  $0 \leq \alpha \leq 1$

Example:  $\alpha=0.5$ :



# Baseband Transmission with Square-Root Raised Cosine Pulses



# Power Spectral Density of Baseband Signals

Assumption: Uncorrelated data with zero mean, e.g.  $d_n \in \{-1, +1\}$

Transmit signal:  $s(t) = \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s)$

Pulse shaping filter:  $g_t(t)$  

**Average autocorrelation function  
for cyclostationary process**

$$\bar{r}_{ss}(\tau) = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} r_{ss}(\tau) dt$$

$$= \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} E\{S^*(t)S(t+\tau)\} dt$$

**Average power spectral density**

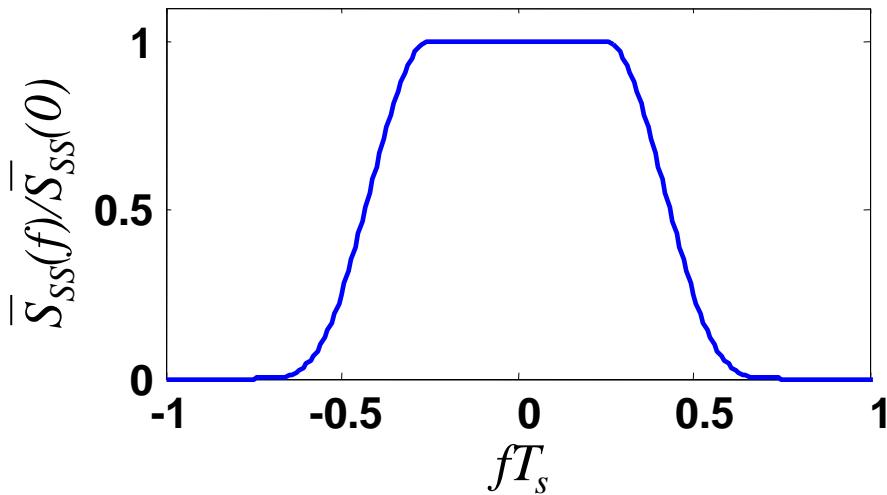
$$\bar{S}_{ss}(f) = \frac{1}{T_s} E\{|d_n|^2\} |G_t(f)|^2$$

The average power spectral density is determined by the pulse shaping filter  $G_t(f)$ . Particularly, the required bandwidth for transmission of the baseband signal is controlled by the pulse shaping filter.

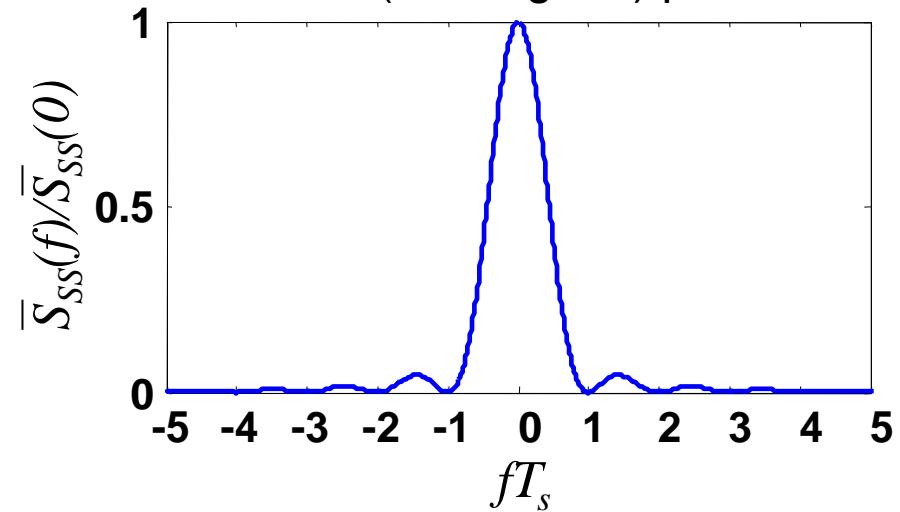
# Power Spectral Density - Examples

$$\bar{S}_{ss}(f) = \frac{1}{T_s} E\left\{ |d_n|^2 \right\} |G_t(f)|^2$$

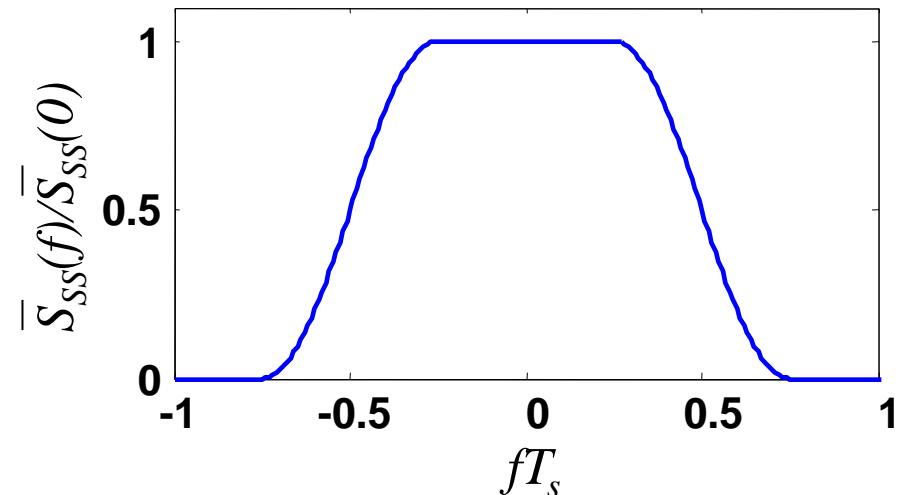
Raised-cosine pulse  
with roll-off factor  $\alpha=0.5$



NRZ (rectangular) pulse



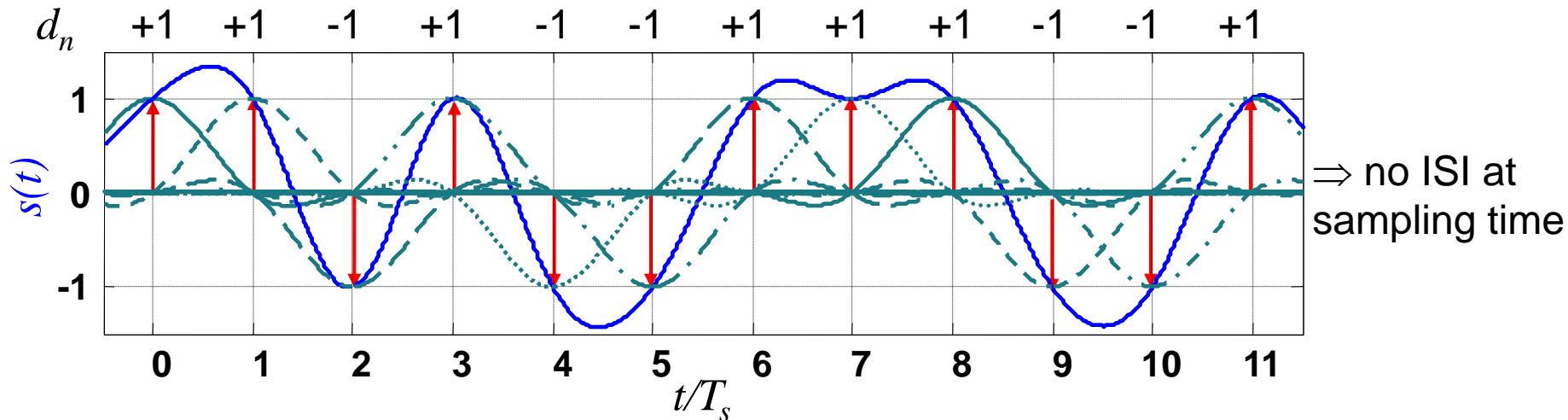
Square-root raised-cosine pulse  
with roll-off factor  $\alpha=0.5$



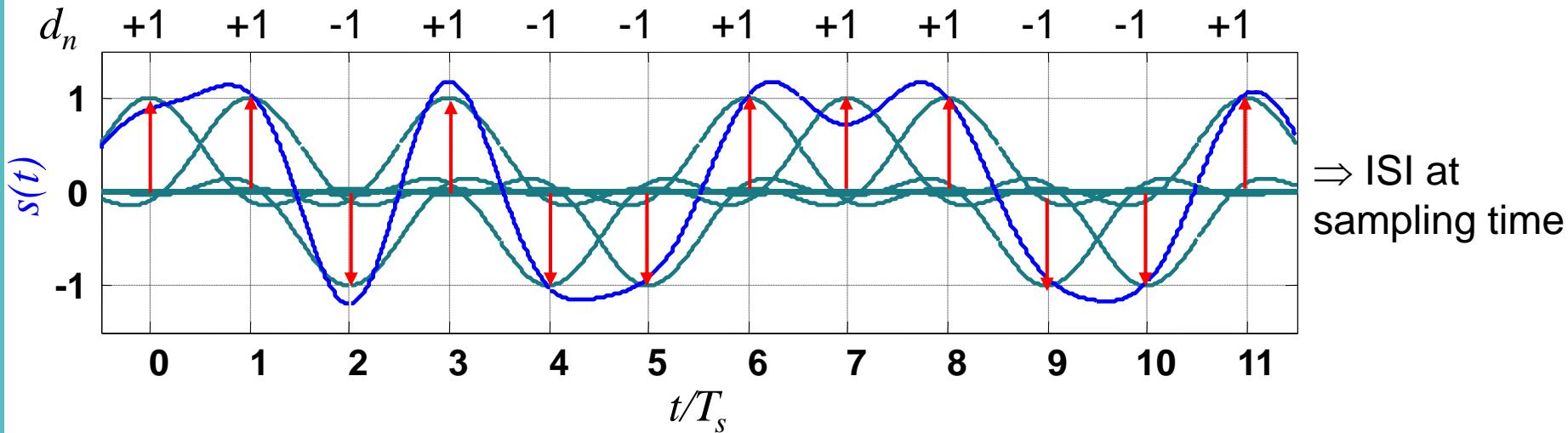
# Intersymbol Interference (ISI)

Transmit signal:  $s(t) = \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s)$

Raised-cosine pulses:



Square-Root-Raised-cosine pulses:



# First Nyquist Criterion

Transmit signal:

$$s(t) = \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s)$$

ISI-free transmission:

$$s(nT_s) = d_n$$

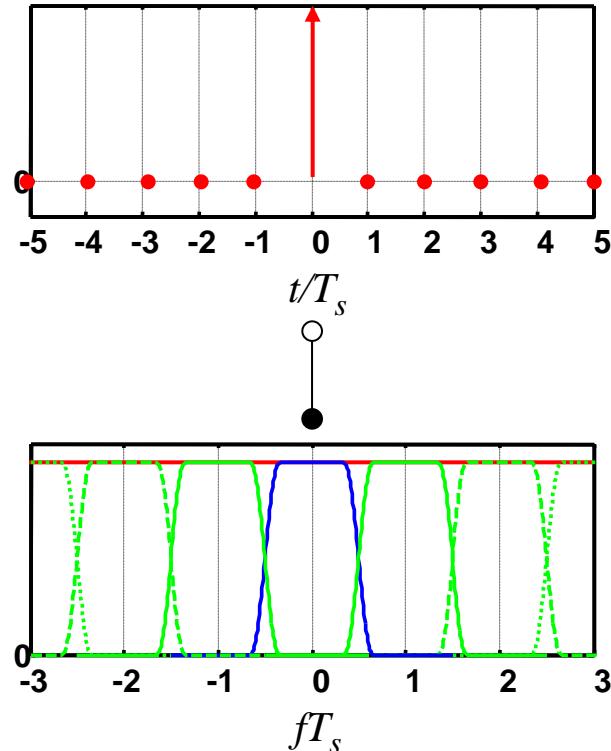
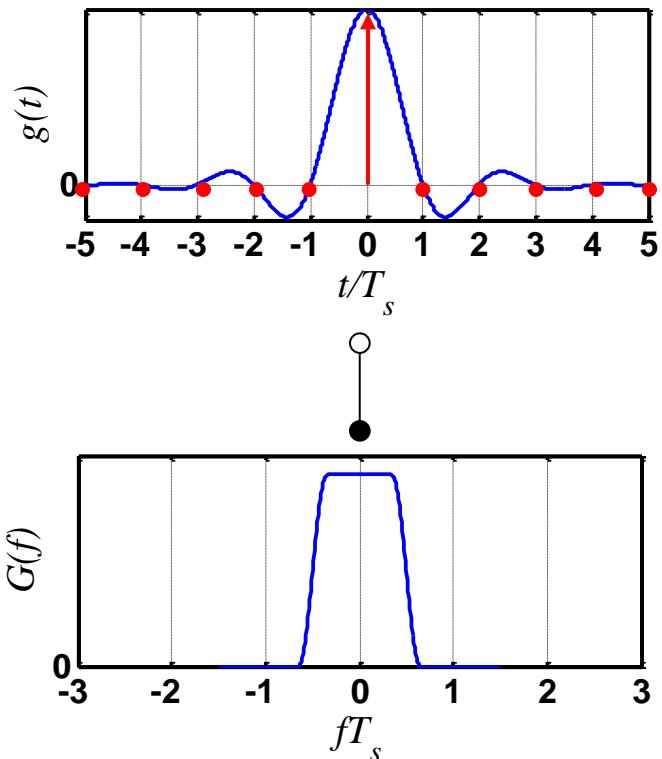
1.Nyquist criterion:

Choose pulse shaping such that no intersymbol interference occurs at the sampling time.

# First Nyquist Criterion

First Nyquist condition in time domain:

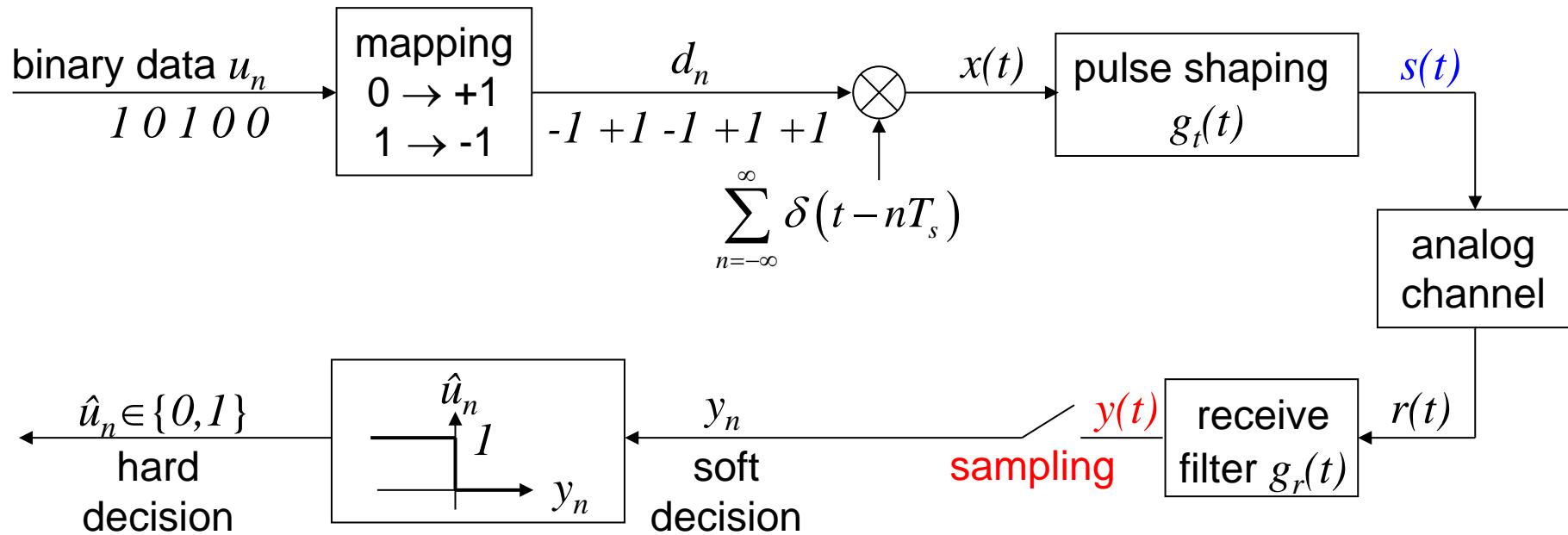
$$g(nT_s) = \begin{cases} 1 & \text{for } n = 0 \\ 0 & \text{for } n \neq 0 \end{cases}$$



First Nyquist condition in frequency domain:

$$\sum_{k=-\infty}^{\infty} G\left(f - \frac{k}{T_s}\right) = \text{const}$$

# First Nyquist Criterion



The 1. Nyquist criterion needs to be met at the output of the receive filter.

⇒ The combination

$$g(t) = g_t(t) * g_r(t) \quad \circ \bullet \quad G(f) = G_t(f) G_r(f)$$

of transmit filter  $g_t(t)$  and receive filter  $g_r(t)$  must meet the 1. Nyquist condition !

⇒ Use filters with square-root Nyquist characteristic (in frequency domain) at both transmitter and receiver, e.g. square-root-raised-cosine filters.

# Nyquist Conditions (1)

The Nyquist conditions are fundamental design criteria for a digital communications system. The first Nyquist criterion states that the signal should be free of intersymbol interference (ISI) at the sampling time, i.e. the sample is influenced only by the desired symbol. ISI would make the signal more sensitive to noise and can cause decision errors. ISI can be controlled by proper choice of the pulse shaping and receive filter. Since the received signal is sampled at the output of the receive filter, the overall transmission scheme, i.e. the combination of transmit and receive filter, should meet the first Nyquist condition.

The 1.Nyquist condition can be formulated in the time domain or in the frequency domain:

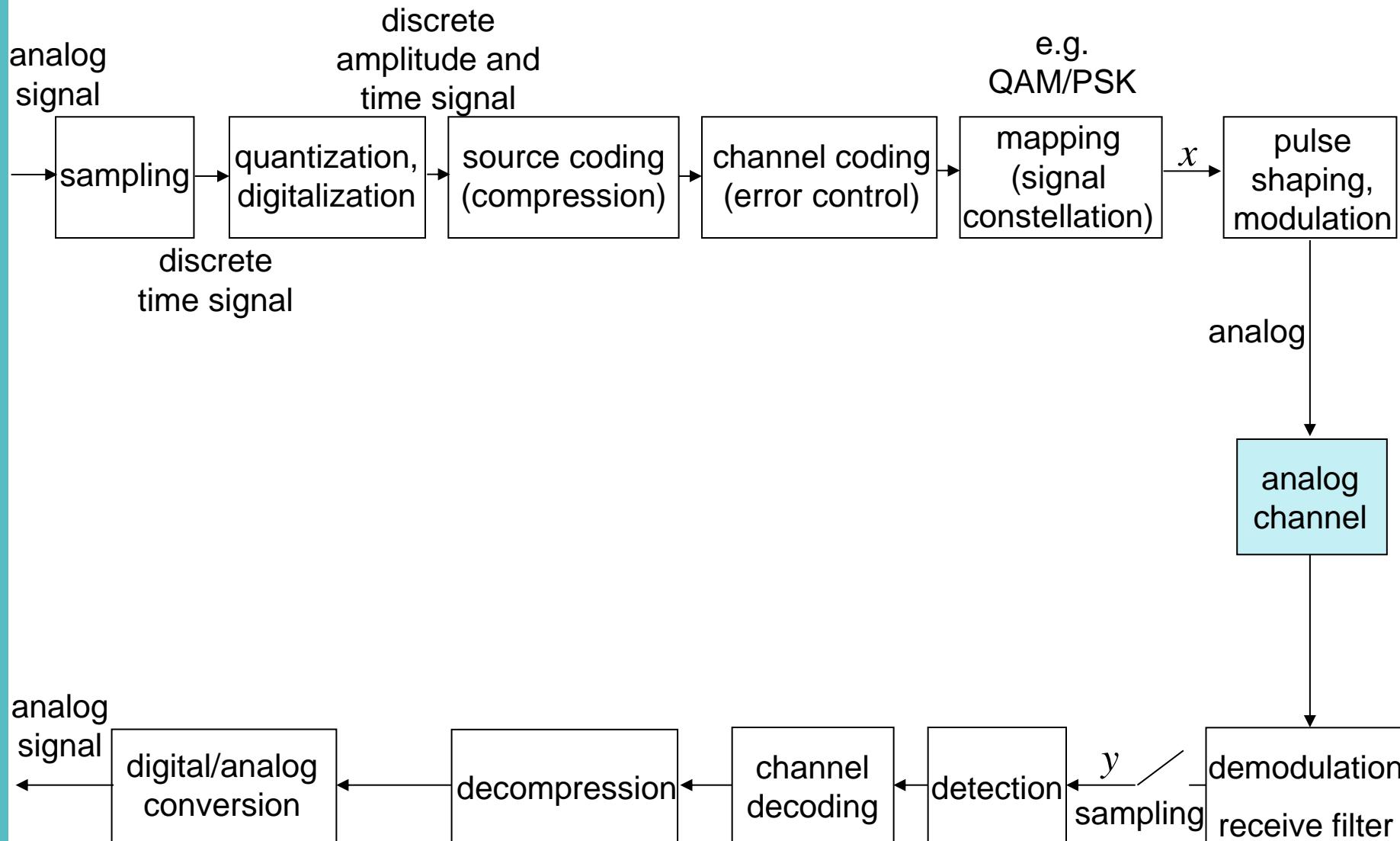
The convolution of the impulse responses of transmit and receive filter should be zero at all multiples of the sample spacing  $nT_s$  except for  $n=0$ . In the frequency domain, this corresponds to the condition that the periodical repetitions of the overall frequency response of transmit and receive filter at multiples of the sampling rate  $1/T_s$  should add up to a constant.

In the eye pattern, the 1.Nyquist condition is met if the vertical eye opening is maximized.

# Nyquist Conditions (2)

In practical systems, the sampling time has to be recovered at the receiver. Usually, the optimum sampling time can not be exactly recovered and there is a jitter. The 2.Nyquist condition aims at keeping the eye as wide open as possible at the actual sampling time. The 2.Nyquist condition is met if the horizontal eye  $A_h$  opening is maximized, i.e.  $A_h = T_s$ . However, with most pulses which are used in practical systems, only the 1.Nyquist condition is met but the 2.Nyquist condition is not exactly met. This is e.g. true for square-root raised cosine pulses.

# Digital Communications System



# White Noise

White noise is frequently used in order to model random distortions. White noise  $N(t)$  is characterized by a constant power spectral density which corresponds to a Dirac impulse as autocorrelation function:

$$r_{NN}(\tau) = \frac{N_0}{2} \delta(\tau) \quad \text{---} \quad S_{NN}(f) = \frac{N_0}{2}$$

Consequently, the power of continuous-time white noise

$$P_N = \int_{-\infty}^{\infty} S_{NN}(f) df = \int_{-\infty}^{\infty} \frac{N_0}{2} df = r_{xx}(0) = \infty$$

is infinite. Hence, white noise is a mathematical, theoretical model, which is not realistic. In practice, noise will be bandlimited and, therefore, have limited power. We call a random process, the power spectral density of which is constant within a band of width  $B$ , *bandlimited white noise*. Note, that the definition of the bandwidth  $B$  counts only the positive frequency components. The autocorrelation function of bandlimited white noise is not a Dirac impulse anymore, but its shape is determined by an si-function:

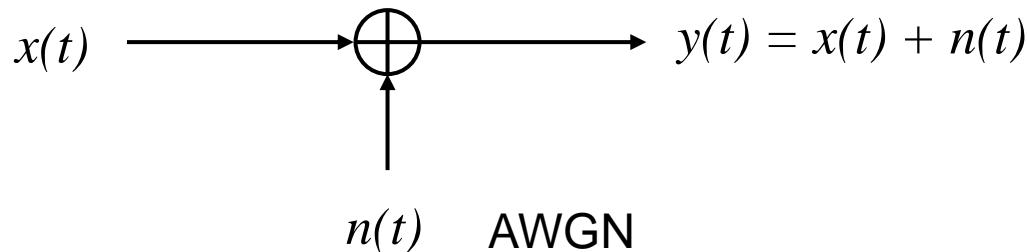
$$r_{NN}(\tau) = N_0 B \operatorname{si}(2B\pi\tau)$$

Hence, samples of the noise process may be correlated. However, since the si-function is zero at  $\tau = k/(2B)$ , time-domain samples of the noise process  $N(t)$  at a temporal distance of multiples of  $T=1/(2B)$  will be uncorrelated.

The power of bandlimited white noise in the baseband with bandwidth  $B$  is given by

$$P_N = \int_{-\infty}^{\infty} S_{NN}(f) df = \int_{-B}^{B} \frac{N_0}{2} df = r_{xx}(0) = N_0 B$$

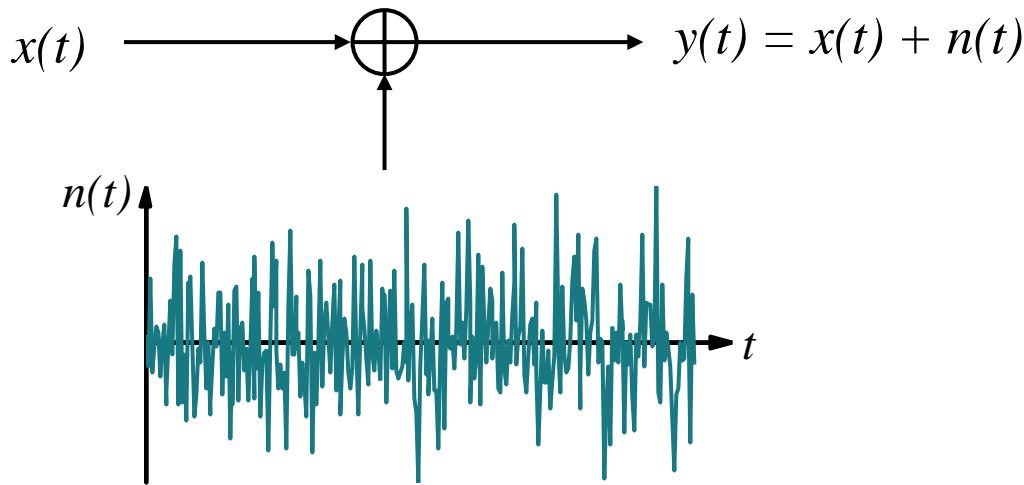
# Additive White Gaussian Noise Channel (AWGN Channel) (1)



AWGN caused by

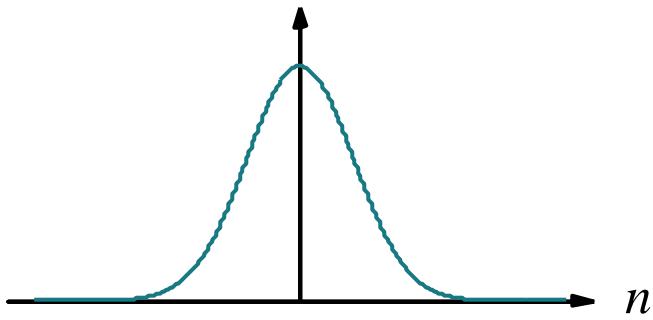
- electronic devices (thermal noise)
- interference (e.g. in a wireless system)
- cross-talk (e.g. due to coupling effects among wires within a cable)
- quantization

# Additive White Gaussian Noise Channel (AWGN Channel) (2)



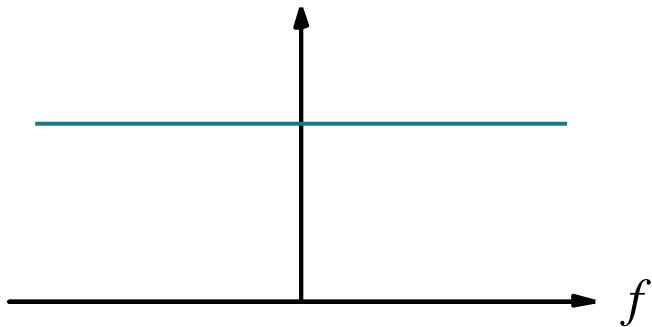
Gaussian amplitude distribution

$$f_N(n) = \frac{1}{\sqrt{2\pi\sigma_N^2}} e^{-\frac{(n-\mu_N)^2}{2\sigma_N^2}}$$

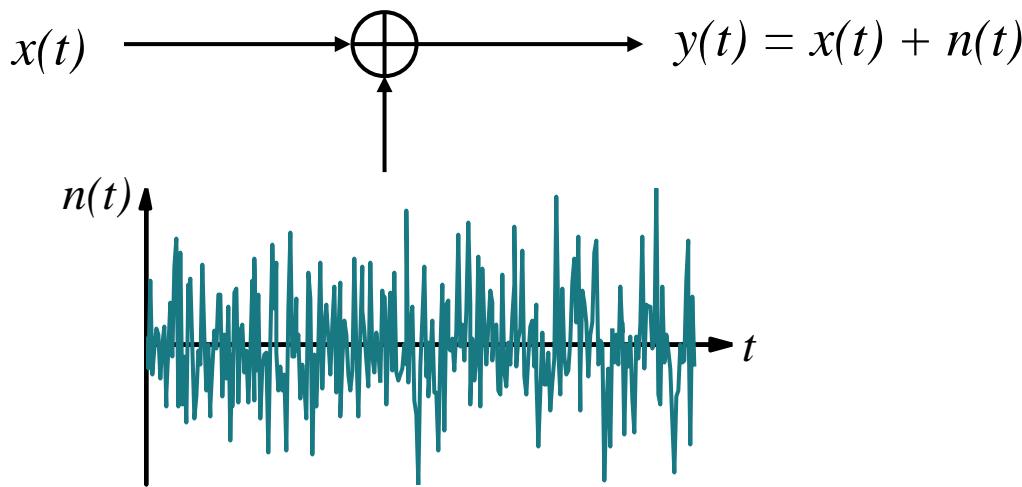


White power spectral density

$$S_{NN}(f) = \frac{N_0}{2}$$

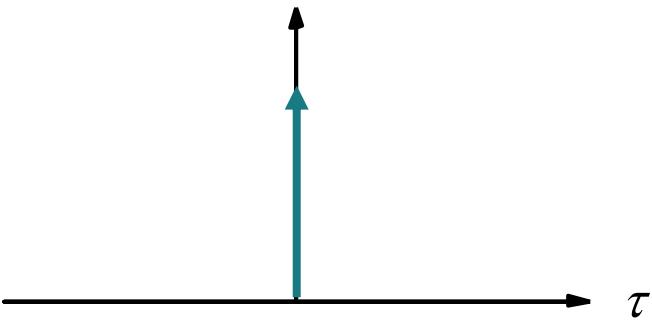


# Additive White Gaussian Noise Channel (AWGN Channel) (3)



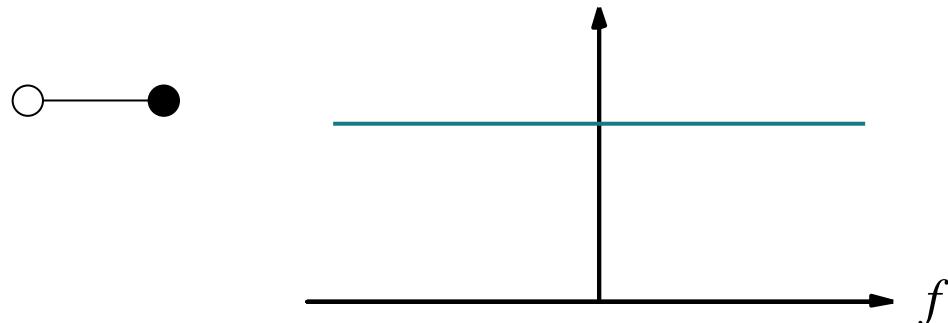
Autocorrelation function

$$r_{NN}(\tau) = \frac{N_0}{2} \delta(\tau)$$

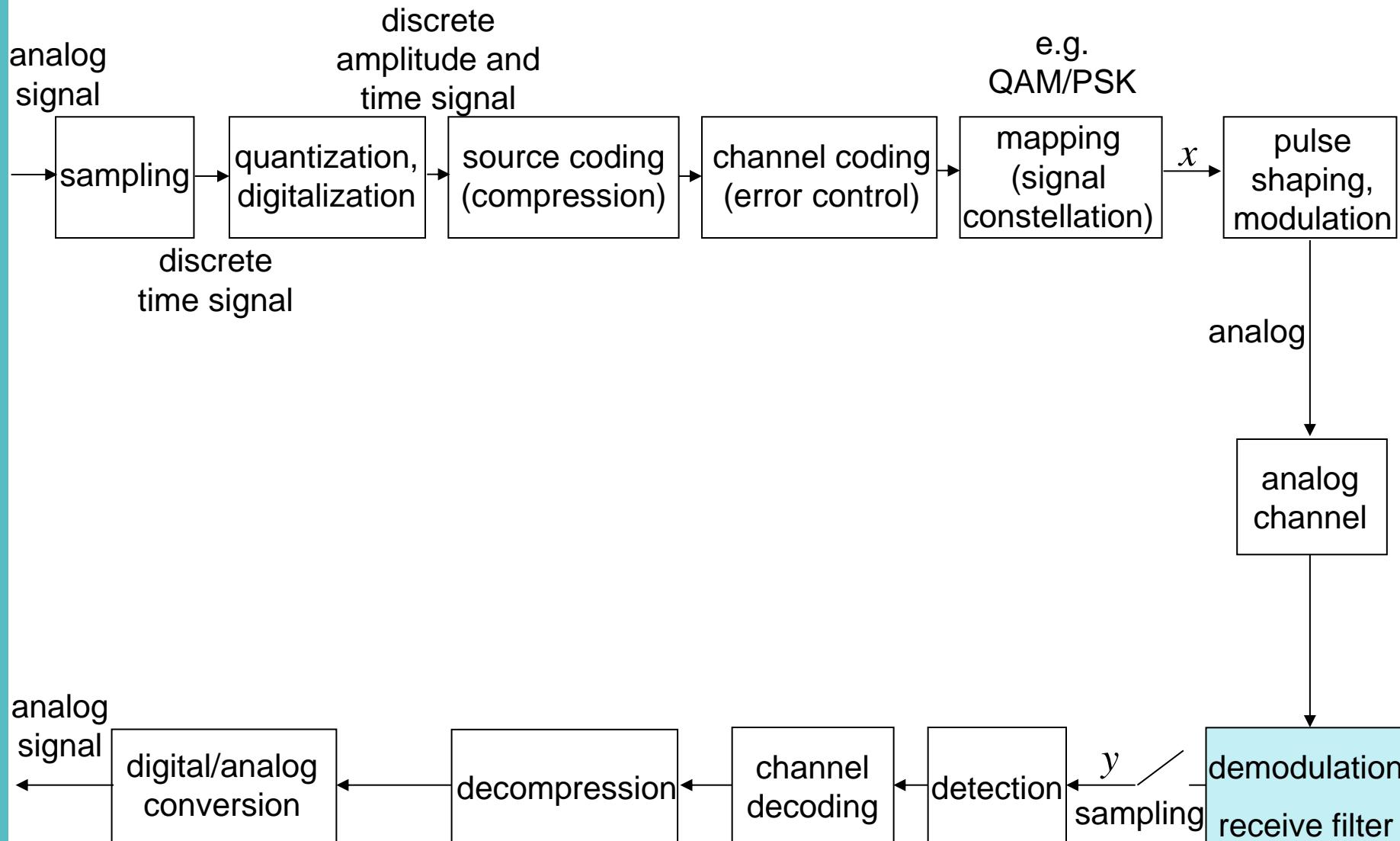


White power spectral density

$$S_{NN}(f) = \frac{N_0}{2}$$



# Digital Communications System



# Matched Filter (1)

The optimum receive filter which maximizes the SNR at the receive filter output in an AWGN channel is the *Matched Filter*. Its impulse response is the flipped conjugate complex impulse response of the transmit filter. In the derivation, we have assumed that the 1. Nyquist condition is met at the output of the matched filter, i.e. there is no ISI at the sampling time. Apart from that assumption, it is an interesting property of the matched filter that its output SNR does not depend on the particular characteristics of the pulse shaping filter  $g_t(t)$ .

The matched filter receiver can also be viewed as a correlation receiver: The received signal is correlated with all possible waveforms of the transmit symbol. The detector then decides for the transmit symbol, for which the correlation is maximized. The correlation receiver requires all transmit waveform candidates have the same energy, i.e.  $|d_{n,k}|^2$  is the same for all possible transmit symbols  $d_{n,1}, \dots, d_{n,M}$ .

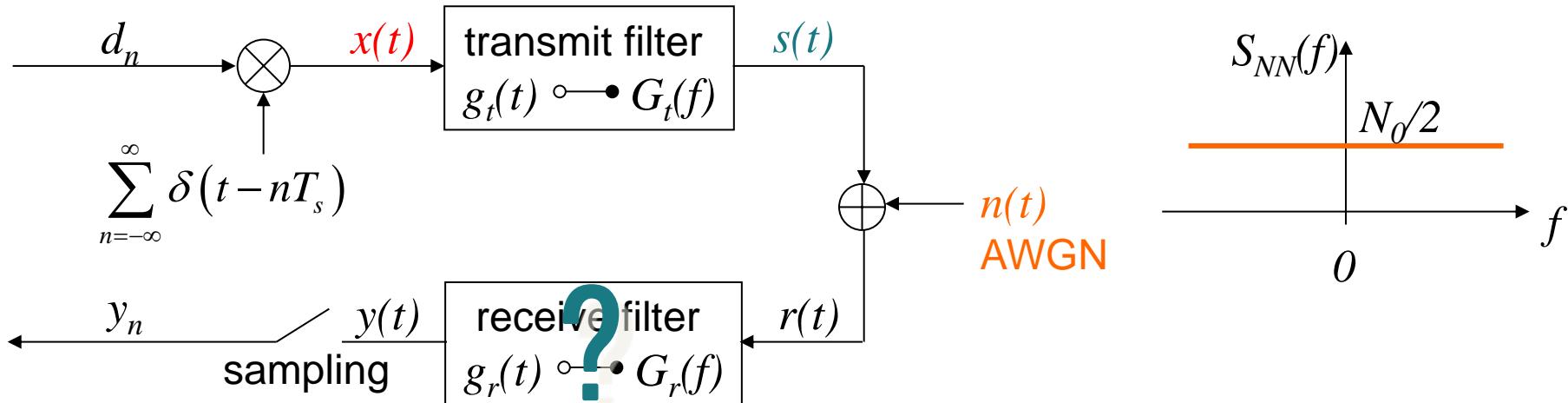
Further investigations show, that the Matched Filter is also optimum with respect to a different optimization criterion: The optimum receiver which minimizes the symbol error probability – the so-called maximum-a-posteriori-probability (APP) detector – is a matched filter at the output of which the signal can be sampled at the symbol rate  $1/T_s$  without loss of information.

# Matched Filter (2)

Most practical filters are real and symmetric such that the same filter is used as transmit and receive filter. A filter can only be realized if it is causal, i.e. the impulse response is zero for  $t < 0$ . In order to make the filters causal, we shift the impulse response along the time axis. This introduces delay. If the impulse response is not time-limited, we have to truncate the impulse response to a duration of  $LT_s$ . This introduces an error which is negligible if the impulse response is very small outside the considered range of  $LT_s$ .

For the receive filter design, two criteria have to be taken into account: The Nyquist condition should be met in order to avoid intersymbol interference (ISI) at the sampling time. Furthermore, the receive filter should be matched to the transmit filter in order to maximize the SNR at the receive filter output. Both criteria together suggest that we should use filters with square-root Nyquist characteristic at both transmitter and receiver. A square-root Nyquist receive filter has the further advantage that the noise in the sampled signal at the receive filter output remains white and Gaussian.

# Detection of Baseband Signals: Receive Filter Design: Matched Filter

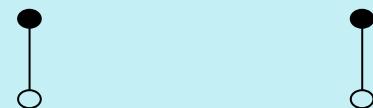


Design criterion:

Choose  $g_r(t)$  such that the SNR at the sampling time is maximized:

$$\frac{P_S}{P_N} \rightarrow \max$$

$$\frac{P_S}{P_N} = \frac{2E_s}{N_0} \text{ for } G_r(f) = \text{const} \cdot G_t^*(f)$$



$$g_r(t) = \text{const} \cdot g_t^*(-t)$$

The **Matched Filter** maximizes the SNR at the receive filter output at the sampling time.

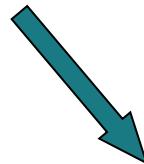
The receive filter  $g_r(t)$  is matched to the transmit filter  $g_t(t)$ .

# Receive Filter Design

## 1.Nyquist condition:

$$g(t) = g_t(t) * g_r(t)$$

meets the 1.Nyquist condition, i.e.  
 $g(t)$  has a Nyquist characteristic.

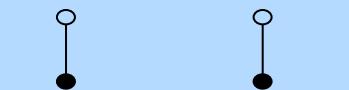


$$G(f) = G_t(f)G_t^*(f) = |G_t(f)|^2$$

$\Rightarrow G_t(f)$  should have a  $\sqrt{\text{Nyquist}}$  characteristic.

## Matched Filter condition:

$$g_r(t) = g_t^*(-t)$$

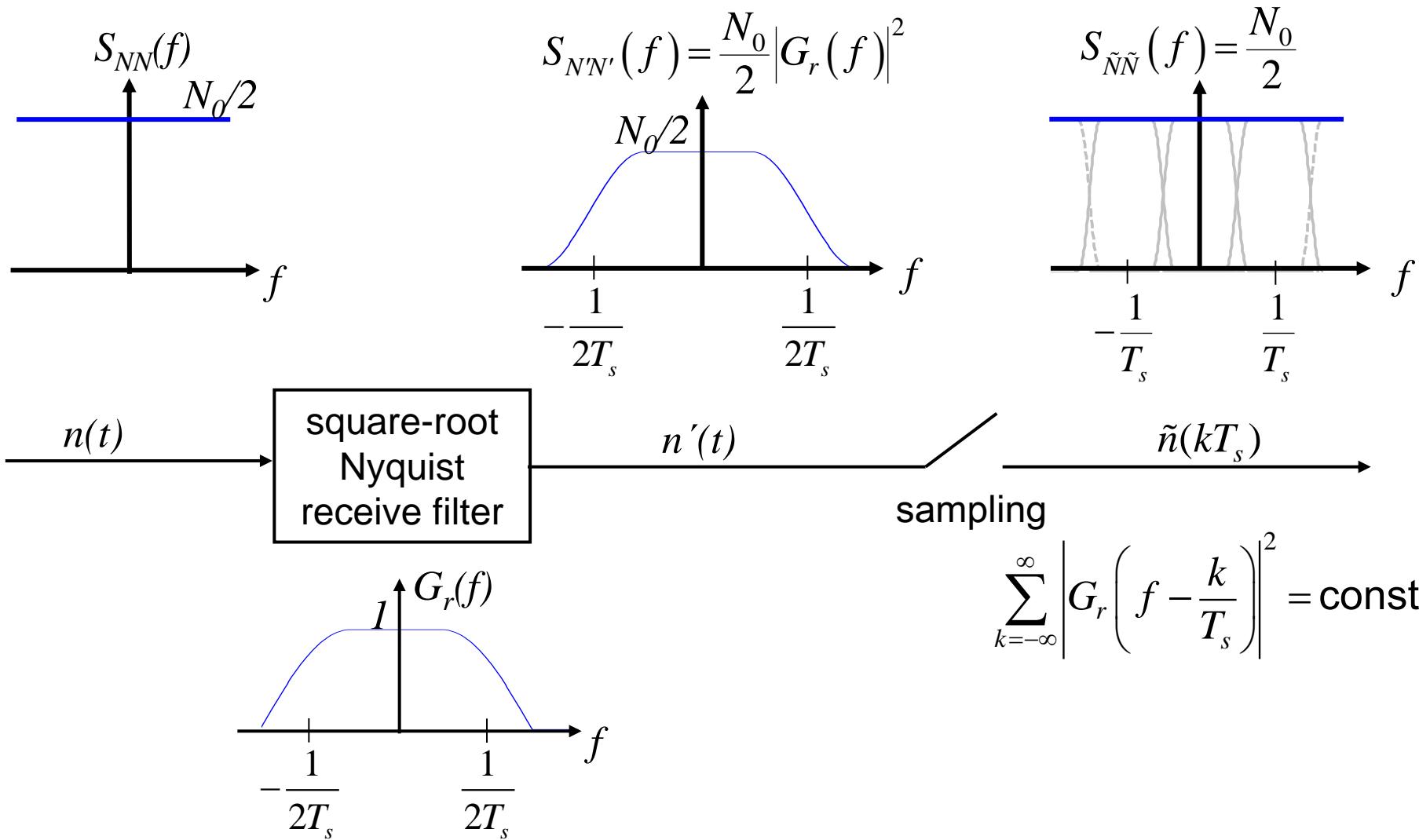

$$G_r(f) = G_t^*(f)$$

For real and symmetric pulses:

$$g_r(t) = g_t(t)$$

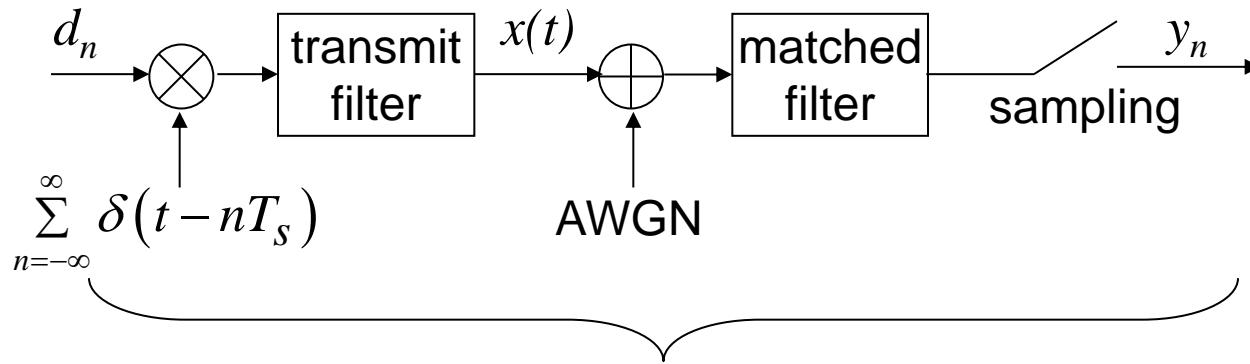


# Noise Power Spectral Density at Receive Filter Output



With square-root Nyquist receive filters, the noise in the sampled signal at the receive filter output remains white and Gaussian (AWGN).

# Discrete Time AWGN Channel Model



Discrete time channel model:

$$d_n = \pm \sqrt{\frac{E_s}{T_s}} \rightarrow y_n$$

$$\sigma^2 = \frac{N_0}{2T_s} \quad \text{AWGN}$$

Equivalent channel model:

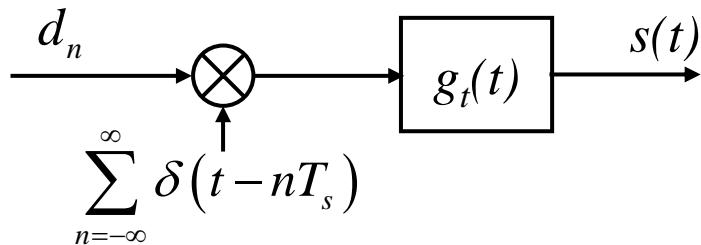
$$\tilde{d}_n = \pm 1 \rightarrow \tilde{y}_n$$

$$\tilde{\sigma}^2 = \frac{N_0}{2T_s} \frac{T_s}{E_s} = \frac{N_0}{2E_s} \quad \text{AWGN}$$

# Discrete Time AWGN Channel Model

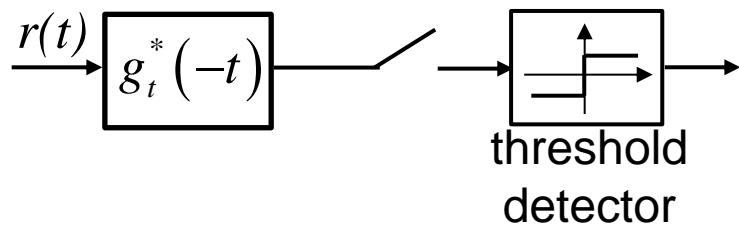
For analysis and simulation of digital communications systems, we can use a discrete time channel model. The performance of a transmission scheme depends only on the ratio between signal power and noise power but not on their absolute values. Consequently, we can normalize the transmit symbols such that their average energy per symbol is one, i.e.  $E\{|d_n|^2\}=1$ . The SNR can then be controlled by a single parameter, i.e. the noise variance  $\sigma^2=N_0/(2E_s)$ .

# Matched Filter Receiver and Correlation Receiver (1)

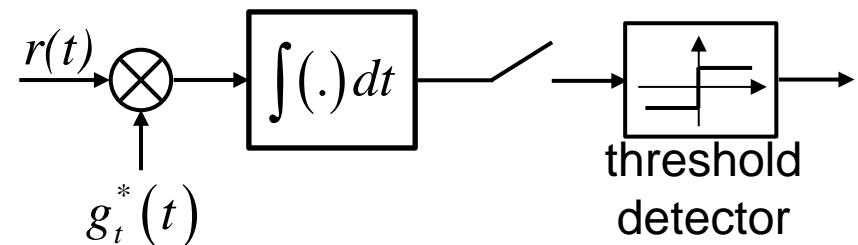


$M$  possible transmit symbols:  
 $d_n \in \{d_{n,1}, \dots, d_{n,M}\}$

**Matched filter receiver**



**Correlation receiver**

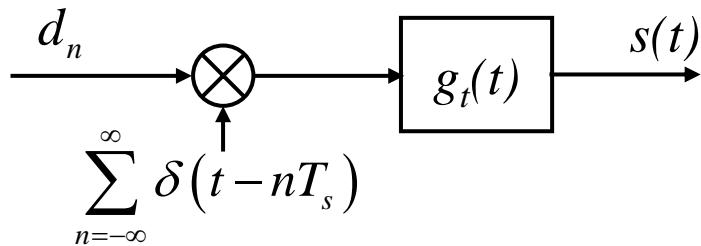


The matched filter maximizes the SNR at the sampling time.

The correlation receiver decides for the candidate symbol with the highest correlation to the received signal.

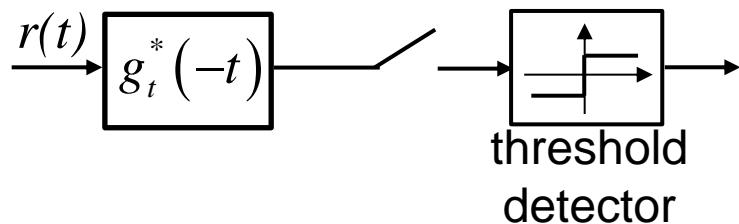
Matched filter receiver and correlation receiver are equivalent.

# Matched Filter Receiver and Correlation Receiver (2)



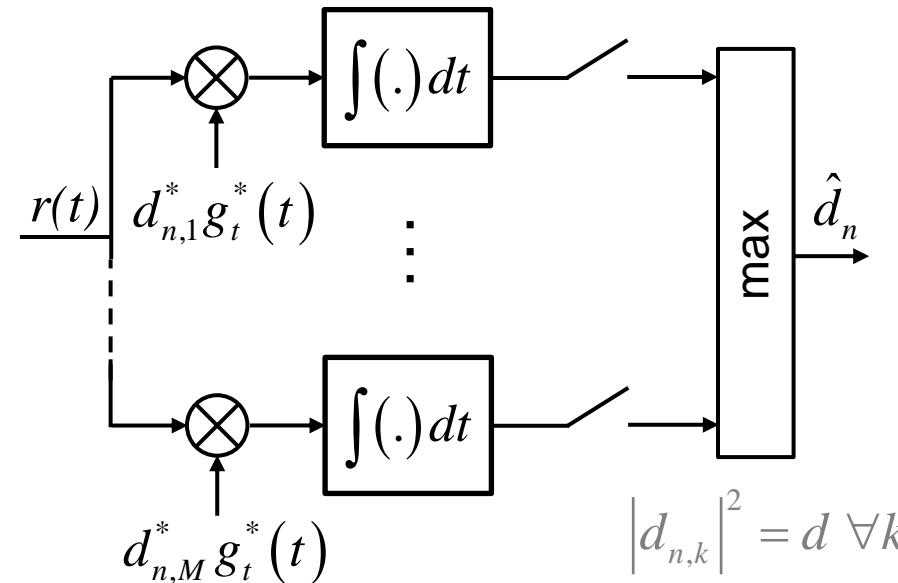
$M$  possible transmit symbols:  
 $d_n \in \{d_{n,1}, \dots, d_{n,M}\}$

**Matched filter receiver**



The matched filter maximizes the SNR at the sampling time.

**Correlation receiver**



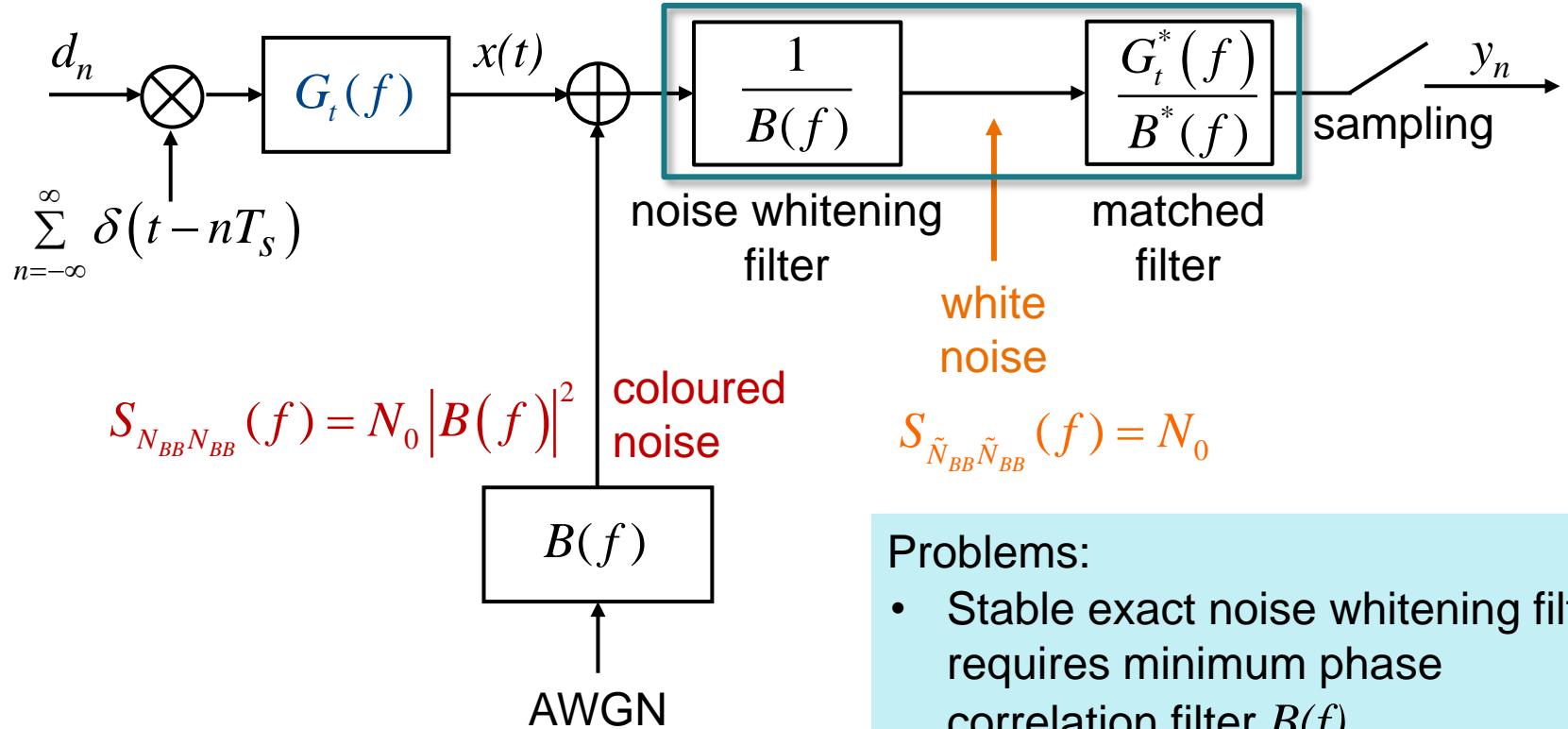
The correlation receiver decides for the candidate symbol with the highest correlation to the received signal.

Matched filter receiver and correlation receiver are equivalent.

# Matched Filter for Coloured Noise

$$G_r(f) = \frac{G_t^*(f)}{B(f)B^*(f)} = \frac{N_0}{S_{N_{BB}N_{BB}}(f)} G_t^*(f)$$

noise whitening matched filter



Problems:

- Stable exact noise whitening filter requires minimum phase correlation filter  $B(f)$ .
- Noise whitening filter causes intersymbol interference (ISI).  
→ equalizer required.

# Matched Filter for Coloured Noise (1)

In some applications, the additive noise is not white but coloured, i.e. the noise samples are correlated. However, it is desirable to have white noise at the detector input as uncorrelated noise simplifies the data detection process. Moreover, the matched filter  $g_r(t) = g_t^*(-t)$  or  $G_r(f) = G_t^*(f)$ , respectively, was derived assuming white noise. Therefore, a noise whitening filter can be applied at the receiver in order to decorrelate the additive noise. The successive matched filter has then to be matched to the concatenation of transmit filter and noise whitening filter.

Coloured noise can be modelled as white Gaussian noise which has been filtered by a frequency-selective filter with frequency response  $B(f)$ . The average power spectral density of the coloured noise in the equivalent baseband is then given by

$$S_{N_{BB}N_{BB}}(f) = N_0 |B(f)|^2.$$

A filter with frequency response  $1/B(f)$  will decorrelate the noise and yield white Gaussian noise with average power spectral density

$$S_{\tilde{N}_{BB}\tilde{N}_{BB}}(f) = S_{N_{BB}N_{BB}}(f) \frac{1}{|B(f)|^2} = N_0$$

at its output. The whitening filter will only be stable if  $B(f)$  is a minimum phase filter. Otherwise, the whitening filter can only be approximated by a non-recursive filter. The frequency response of the effective system seen at the output of the noise whitening filter is determined by the concatenation of transmit filter and noise whitening filter, i.e. by

$$G_t(f) \cdot \frac{1}{B(f)}.$$

# Matched Filter for Coloured Noise (2)

Consequently, the matched filter which maximizes the SNR at its output at the sampling time has to be matched to this effective system. The frequency response of the matched filter is then given by

$$\frac{G_t^*(f)}{B^*(f)}.$$

The receive filter consists of the concatenation of the noise whitening filter and the matched filter and is called *noise whitening matched filter*. Its frequency response is given by

$$G_r(f) = \frac{G_t^*(f)}{|B(f)|^2}.$$

As the denominator of the equation above is proportional to the average power spectral density of the coloured noise, the noise whitening filter can also be expressed by

$$G_r(f) = \frac{N_0 G_t^*(f)}{S_{N_{BB}N_{BB}}(f)}$$

which illustrates the noise whitening effect.

Unfortunately, filtering the received signal by a noise whitening matched filter typically results in intersymbol-interference (ISI). Hence, the first Nyquist condition is not met at the output of the noise whitening matched filter and a successive equalizer has to be applied.

# Matched Filter for Coloured Noise (3)

The relation of transmit filter and receive filter can also be characterized in the time domain using the autocorrelation function  $r_{N_{BB}N_{BB}}(\tau)$  of the coloured noise: Reformulating the equation above and applying the inverse Fourier transform, we obtain

$$G_r(f)S_{N_{BB}N_{BB}}(f) = N_0 G_t^*(f)$$



$$g_r(\tau) * r_{N_{BB}N_{BB}}(\tau) = N_0 g_t^*(-\tau)$$

# Bit Error Probability and Bit Error Rate (BER) of Binary Signaling

The bit error probability for binary transmission over a baseband AWGN channel can be expressed by the complementary error function  $\text{erfc}(.)$ . The bit error probability decreases exponentially with increasing SNR.

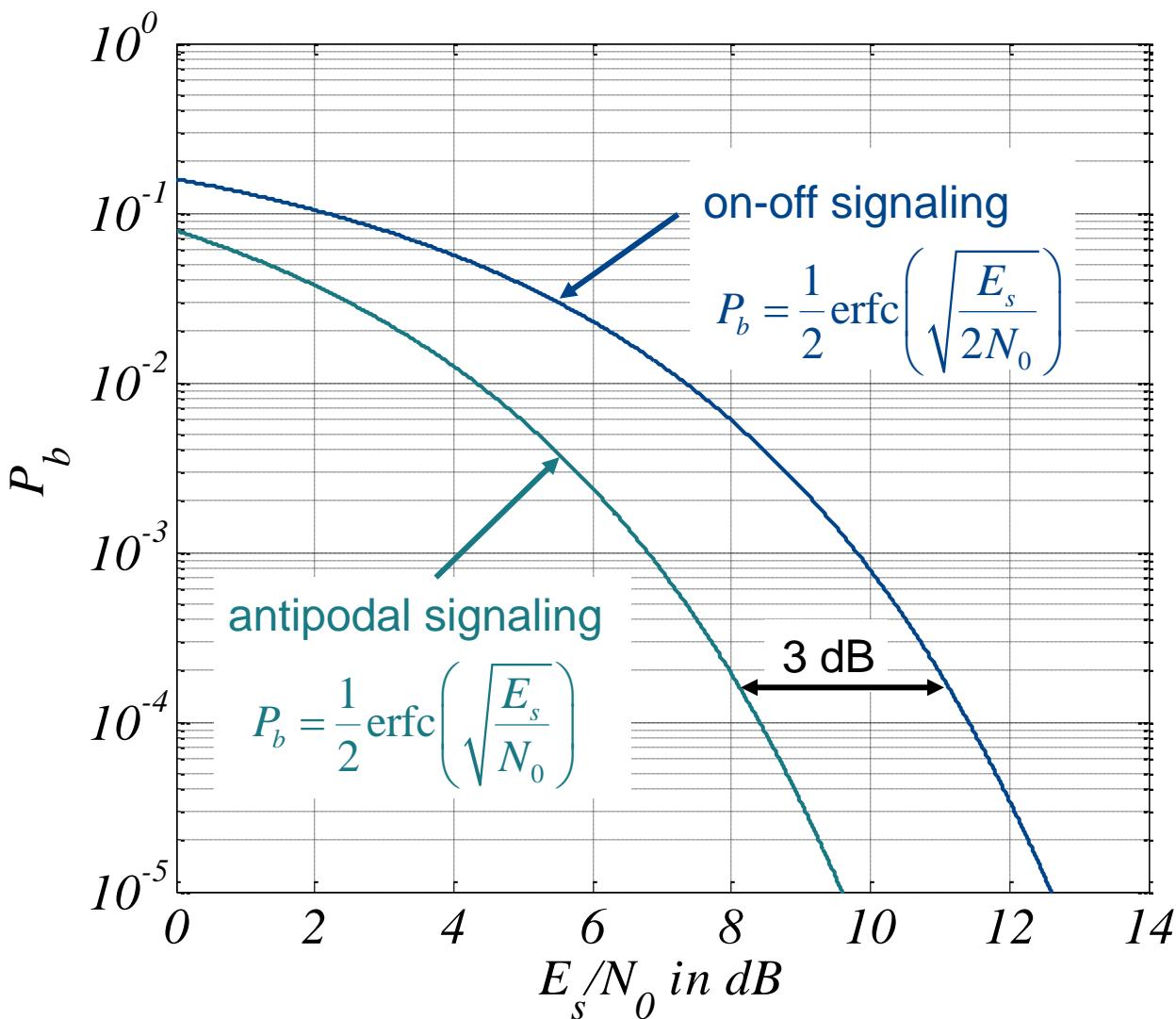
For on-off signaling, a 3 dB higher SNR is required compared to antipodal signaling in order to achieve the same bit error probability. This is a consequence of the fact that the two possible transmit symbols for a logical 0 or 1 in on-off signaling are orthogonal (correlation coefficient  $\rho=0$ ), whereas the respective transmit symbols for antipodal signaling are antipodal (correlation coefficient  $\rho=-1$ ). Hence, it is easier for the receiver to distinguish between the two possible waveforms in antipodal signaling than in on-off signaling.

Note, that in order to obtain a bit error probability of  $10^{-5}$  with antipodal signaling, an SNR of 9.6 dB is required.

Typical target bit error rates (after error control decoding) are  $10^{-3}$  for speech transmission and  $10^{-5}$  or lower for data transmission.

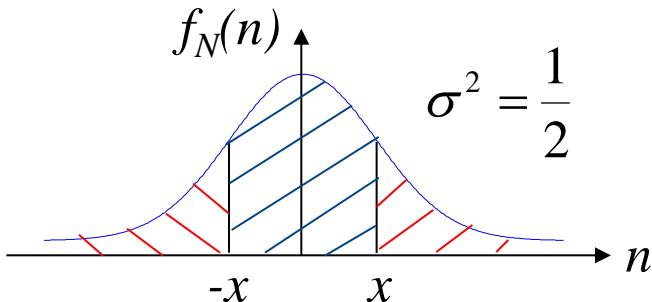
In measurements or computer simulations of communications systems, the bit error rate (BER) is counted. Strictly speaking, a measurement or simulation delivers a bit error rate, while the bit error probability is determined analytically. However, in general usage, the terms BER and bit error probability are often used synonymously.

# Bit Error Probability of Binary Signaling in AWGN Channel



# Error Functions

$$f_N(n) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{n^2}{2\sigma^2}}$$

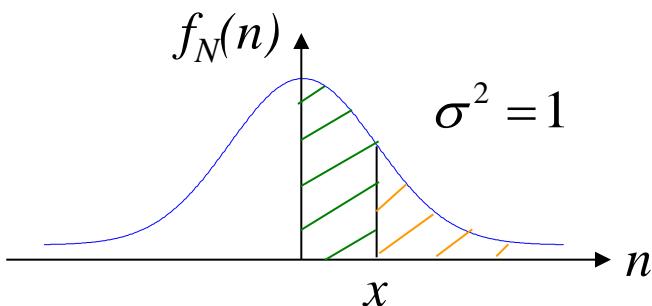


**Error function:**

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt = P(|n| < x)$$

**Complementary error function:**

$$\text{erfc}(x) = 1 - \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt = 2P(n > x)$$



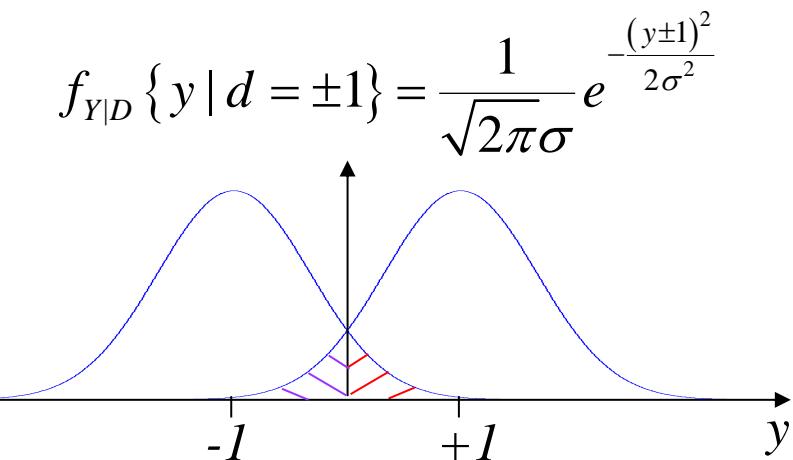
**$\Phi_0$ -function:**

$$\Phi_0(x) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{t^2}{2}} dt = P(0 \leq n \leq x)$$

**Q-function:**

$$\begin{aligned} Q(x) &= \frac{1}{2} - \Phi_0(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt \\ &= \frac{1}{2} \text{erfc}\left(\frac{x}{\sqrt{2}}\right) = P(n > x) \end{aligned}$$

# Bit Error Probability in AWGN Channel for Binary Antipodal Signaling



Assumptions:

- Nyquist condition is met
- Matched filter reception
- AWGN channel
- Equally likely transmit symbols  $d \in \{\pm 1\}$

Bit error probability:  $P_b = p\{y \leq 0 | d = +1\} \underbrace{P\{d = +1\}}_{1/2} + p\{y \geq 0 | d = -1\} \underbrace{P\{d = -1\}}_{1/2}$

Equally likely transmit symbols:

$$p\{y \leq 0 | d = +1\} = p\{y \geq 0 | d = -1\} = p\{n \geq 1\} = \int_1^\infty \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{n^2}{2\sigma^2}} dn$$

$$P_b = \int_1^\infty \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{n^2}{2\sigma^2}} dn = \int_{\frac{1}{\sqrt{2\sigma}}}^\infty \frac{\sqrt{2\sigma}}{\sqrt{2\pi}\sigma} e^{-t^2} dt = \frac{1}{2} \operatorname{erfc}\left(\frac{1}{\sqrt{2\sigma}}\right) \Rightarrow P_b = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_s}{N_0}}\right)$$

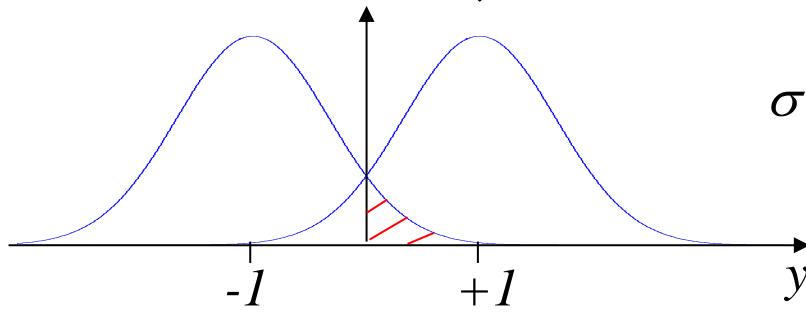
$$t = \frac{n}{\sqrt{2\sigma}} \Rightarrow \frac{dt}{dn} = \frac{1}{\sqrt{2\sigma}}$$

$$\sigma^2 = \frac{N_0}{2E_s}$$

# Bit Error Probability in AWGN Channel for Binary Antipodal Signaling and On-Off Signaling

## Antipodal signaling

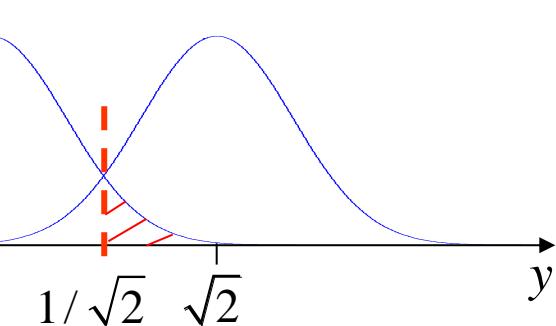
$$f_{Y|D} \{y | d = \pm 1\} = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y \pm 1)^2}{2\sigma^2}}$$



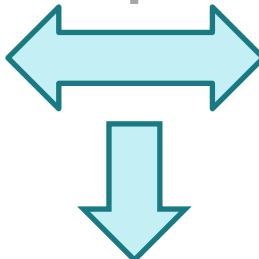
## On-off signaling

$$f_{Y|D} \{y | d\}$$

$$\sigma^2 = \frac{N_0}{2\bar{E}_s}$$



$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_s}{N_0}} \right)$$



$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{\bar{E}_s}{2N_0}} \right)$$

On-off signaling requires a 3dB higher SNR than antipodal signaling in order to obtain the same  $P_b$ .

# Baseband and Bandpass Signals

A message signal is typically a *baseband* signal, i.e. its spectrum contains low frequencies including zero up to a maximum frequency.

However, for many applications, baseband transmission is not suitable but the spectrum of the transmit signal has to be shifted to a higher frequency range, called the *passband*.

This might be necessary in order to adapt the transmit signal to the passband of the transmission medium. E.g. radio transmission is not possible in the baseband but requires higher frequencies.

Another reason for bandpass transmission is that different frequencies are allocated for certain services. E.g. in Germany, the radio frequency transmission range around 900 MHz is allocated to second generation mobile communications services (GSM) whereas the transmission range around 2 GHz is allocated to third generation mobile communications services (UMTS).

Furthermore, adapting the used frequency range allows to support multiple users at the same time at different frequencies (Frequency Division Multiple Access, FDMA). Bandpass transmission is realized by carrier modulation. I.e. the information carrying baseband signal modulates a parameter (amplitude, frequency, phase) of a cosine or sine wave. Depending on the transmission medium, we have an electromagnetic, acoustical or optical wave. The modulated parameter of the cosine or sine wave determines the name of the modulation scheme: Amplitude modulation, frequency modulation or phase modulation.

# Quadrature Modulator (1)

The quadrature modulator is a fundamental module of a communications system. A message signal is typically a *baseband* signal  $x_{BB}(t)$ , i.e. its spectrum  $X_{BB}(f)$  contains low frequencies including zero up to a maximum frequency  $f_{\max}$ . By multiplying the message signal by a cosine or sine oscillation, respectively, of frequency  $f_c$ , the spectrum of the signal is shifted to another frequency band, called the *passband*, around the frequency  $f_c$  and  $-f_c$ . The oscillation is called the carrier and its frequency is the *carrier frequency*  $f_c$ . The shift operation from baseband to passband is called upconversion or modulation. Since any physical transmit signal  $x(t)$  is a real signal, the spectrum  $X(f)$  of the bandpass signal  $x(t)$  must be conjugate even, i.e. the real part  $\text{Re}\{X(f)\}$  is even whereas the imaginary part  $\text{Im}\{X(f)\}$  is odd. This emphasizes that the spectrum  $X(f)$  must have components around both  $f_c$  and  $-f_c$ .

At the receiver, the signal has to be downconverted to the baseband. This is achieved by multiplying the received bandpass signal by a cosine or sine oscillation, respectively, of the same carrier frequency  $f_c$  and successive low pass filtering. We assume that the signal has not been distorted by the channel. Hence, the received signal  $r(t)=x(t)$  is equal to the transmitted signal  $x(t)$ . The multiplication of the received bandpass signal shifts the spectrum  $X(f)$  by  $f_c$  both to the left and to the right. Consequently, we obtain components in the resulting spectrum in the baseband and around  $\pm 2f_c$ . The baseband component is the spectrum of the desired message signal, whereas the demodulation products around  $\pm 2f_c$  are undesired. The undesired components are eliminated by a successive low pass filter.

# Quadrature Modulator (2)

The quadrature modulator makes use of the fact that cosine and sine are orthogonal functions in order to use the frequency band more efficiently, i.e. in order to improve the bandwidth-efficiency: A second message signal  $x_{BB,Q}(t)$  can be transmitted simultaneously in the same frequency band, if the first message signal  $x_{BB,I}(t)$  is upconverted by a cosine oscillation, whereas the second signal  $x_{BB,Q}(t)$  is upconverted using a sine oscillation. The transmit signal in the passband is then given by

$$x(t) = x_{BB,I}(t)\sqrt{2} \cos(2\pi f_c t) - x_{BB,Q}(t)\sqrt{2} \sin(2\pi f_c t).$$

The message signals  $x_{BB,I}(t)$  and  $x_{BB,Q}(t)$  are called the *quadrature components* of the baseband signal  $x_{BB}(t)$ , where  $x_{BB,I}(t)$  is called the *inphase component* and  $x_{BB,Q}(t)$  is called the *quadrature component*.

Due to the orthogonality of cosine and sine, the quadrature components can be separated at the receiver: A quadrature demodulator multiplies the received signal in the inphase branch by a cosine oscillation of carrier frequency  $f_c$  and in the quadrature component by a sine oscillation of carrier frequency  $f_c$ . Again, the demodulation products around  $\pm 2f_c$  are eliminated by low pass filters. Due to the orthogonality of cosine and sine, only the desired signal components  $x_{BB,I}(t)$  and  $x_{BB,Q}(t)$ , respectively, are observed at the outputs of the low pass filters in inphase and quadrature branch, respectively.

Remark: We normalize the carrier oscillations at both transmitter and receiver by a factor  $\sqrt{2}$ . Another representation which is sometimes found in literature is obtained, if the factor  $\sqrt{2}$  is dropped at the transmitter and instead a normalization factor of 2 is used at the receiver.

# Equivalent Baseband Signal (1)

The mathematical treatment of bandpass signals is inconvenient since several multiplications with cosine and sine carrier oscillations are involved. A significantly simplified mathematical treatment of bandpass signals and systems can be achieved based on a complex equivalent baseband model.

We distinguish between equivalent baseband signals and equivalent baseband systems.

A bandpass transmit signal can be completely described by the complex equivalent baseband signal: We interprete the inphase component  $x_{BB,I}(t)$  as the real part  $\text{Re}\{x_{BB}(t)\}$  of a complex equivalent baseband signal  $x_{BB}(t)$  and the quadrature component  $x_{BB,Q}(t)$  as imaginary part  $\text{Im}\{x_{BB}(t)\}$  of the complex equivalent baseband signal  $x_{BB}(t)$ . Hence, the equivalent baseband signal is given by

$$x_{BB}(t) = x_{BB,I}(t) + j x_{BB,Q}(t).$$

The relation of the complex equivalent baseband signal  $x_{BB}(t)$  and the real bandpass transmit signal  $x(t)$  is then given by

$$x(t) = x_{BB,I}(t)\sqrt{2} \cos(2\pi f_c t) - x_{BB,Q}(t)\sqrt{2} \sin(2\pi f_c t) = \sqrt{2} \text{Re}\{x_{BB}(t)e^{j2\pi f_c t}\}.$$

# Equivalent Baseband Signal (2)

The factor  $\sqrt{2}$  in the definition of the equivalent baseband signal is not always used in literature. It is included here in order to achieve the same power of bandpass signal and equivalent baseband signal. This definition of the equivalent baseband signal is also the motivation for using the normalization factor  $\sqrt{2}$  at both transmitter and receiver in the quadrature modulator. Moreover, the relationship

$$x(t) = \sqrt{2} \operatorname{Re} \left\{ x_{BB}(t) e^{j2\pi f_c t} \right\}$$

between bandpass signal and equivalent baseband signal only holds, if the quadrature component is upconverted using a multiplication by  $- \sin(2\pi f_c t)$ . The minus is only required for mathematical convenience. From a technical point of view, the concept of quadrature modulation would also work with a positive sine oscillation  $+ \sin(2\pi f_c t)$  in the quadrature branch.

The complex equivalent baseband signal  $x_{BB}(t)$  has a physical interpretation and, therefore, is also called the *complex envelope* of the bandpass signal. The magnitude  $\sqrt{2} |x_{BB}(t)|$  of the scaled equivalent baseband signal is the real envelope of the bandpass signal  $x(t)$ .

We emphasize again that a physical signal is always real. Complex signals do not exist in nature but are used for more convenient mathematical treatment. Consequently, the physical bandpass signal  $x(t)$  is always real. The complex equivalent baseband signal  $x_{BB}(t)$  is only a mathematical representation of  $x(t)$ .

# Equivalent Baseband System

In the same way as we describe bandpass signals  $x(t)$  by an equivalent baseband signal  $x_{BB}(t)$ , we can characterize a bandpass LTI system, e.g. a transmission channel, by its equivalent baseband system.

A bandpass system can be characterized by a real impulse response  $h(t)$ . The complex impulse response of its equivalent baseband system is denoted by  $h_{BB}(t)$ .

However, we need to make sure that the scaling of the input signal by the equivalent baseband system is the same as for the bandpass signal. Therefore, unlike for the equivalent baseband signal, the factor  $\sqrt{2}$  has to be dropped in the definition of the equivalent baseband system.

Consequently, the relationship of impulse response  $h_{BB}(t)$  of the equivalent baseband system and the impulse response  $h(t)$  of the bandpass system is given by

$$h(t) = 2 \operatorname{Re} \left\{ h_{BB}(t) e^{j2\pi f_c t} \right\}$$

The processing of an input signal  $x(t)$  by a bandpass system  $h(t)$  can be equivalently described in the equivalent baseband:

The output signal  $y(t)$  of the bandpass system  $h(t)$  is given by the convolution of the input signal  $x(t)$  and the impulse response of the bandpass system, i.e.

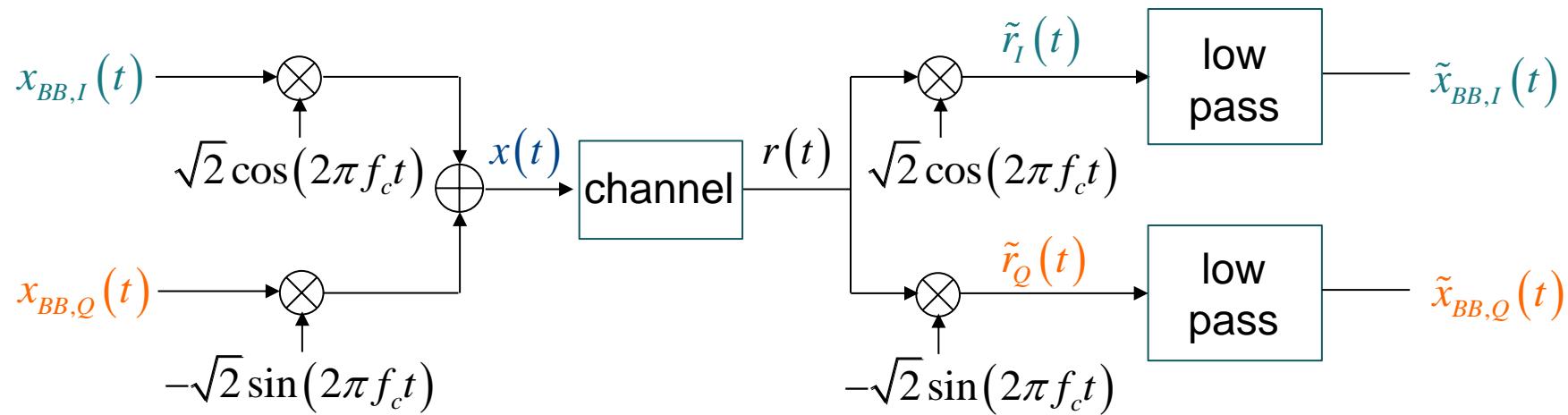
$$y(t) = x(t) * h(t).$$

Equivalently, the operation can be described in the equivalent baseband by

$$y_{BB}(t) = x_{BB}(t) * h_{BB}(t).$$

In the frequency domain, we obtain  $Y(f) = H(f) X(f)$  or in the equivalent baseband  $Y_{BB}(f) = X_{BB}(f) H_{BB}(f)$ .

# Quadrature Amplitude Modulation (QAM) (1)



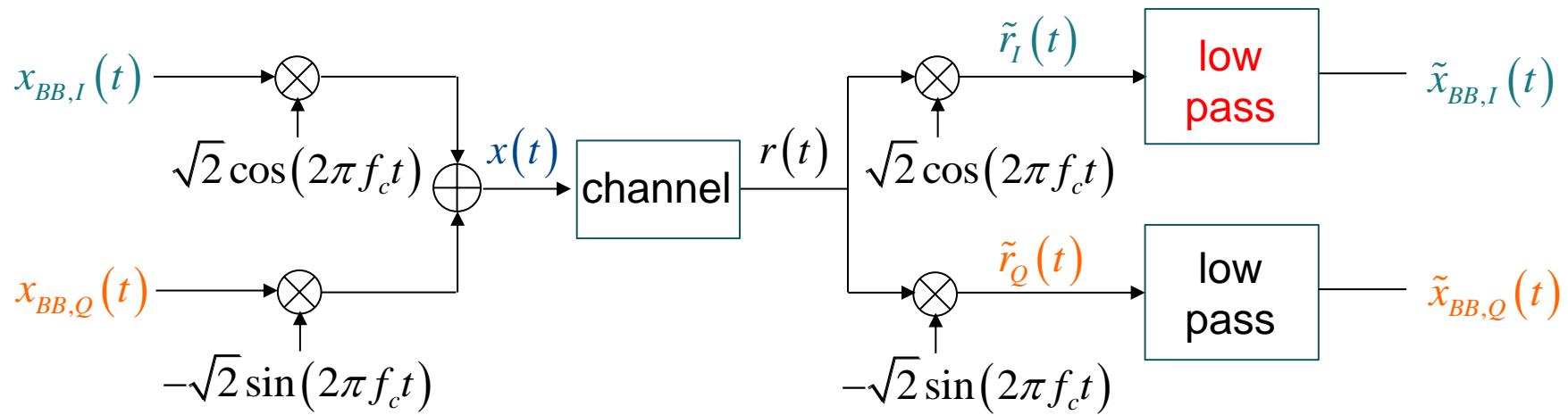
Transmit signal:  $x(t) = x_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - x_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t)$

Assumption:

no distortion during transmission through channel

→ received signal  $r(t) = \text{transmitted signal } x(t)$ .

# Quadrature Amplitude Modulation (QAM) (2)



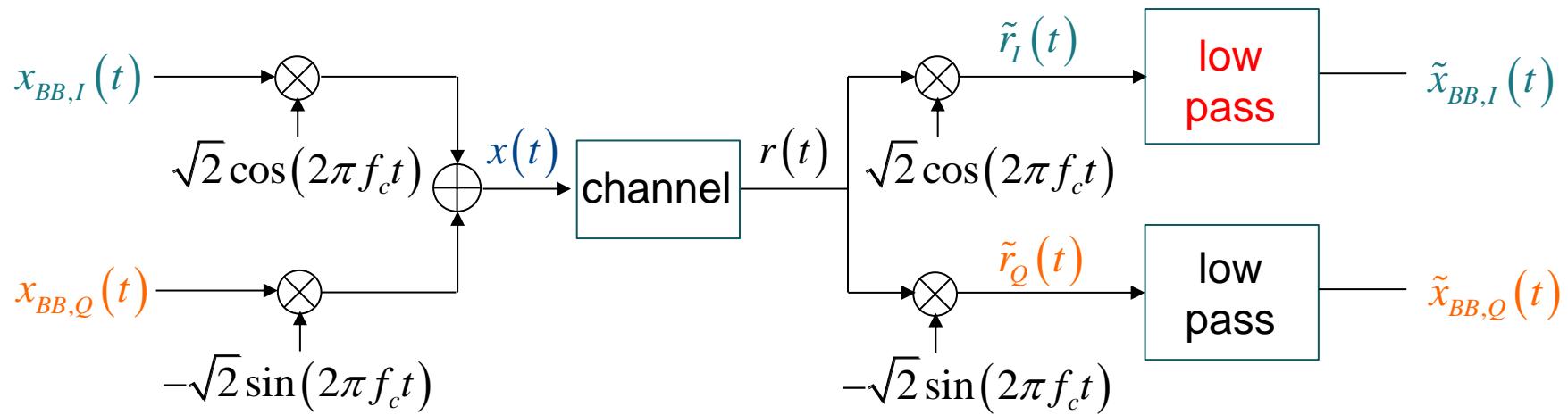
$$\tilde{r}_I(t) = r(t) \sqrt{2} \cos(2\pi f_c t)$$

$$= x_{BB,I}(t) \sqrt{2} \cos(2\pi f_c t) \sqrt{2} \cos(2\pi f_c t) - x_{BB,Q}(t) \sqrt{2} \sin(2\pi f_c t) \sqrt{2} \cos(2\pi f_c t)$$

eliminated by low pass filter

$$= x_{BB,I}(t) [1 + \cos(4\pi f_c t)] - x_{BB,Q}(t) [0 + \sin(4\pi f_c t)] \rightarrow \tilde{x}_{BB,I}(t) = x_{BB,I}(t)$$

# Quadrature Amplitude Modulation (QAM) (3)



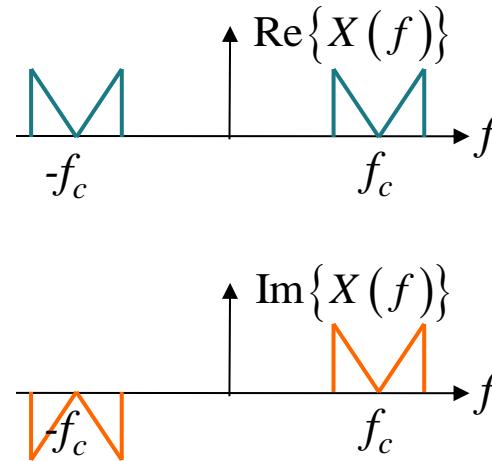
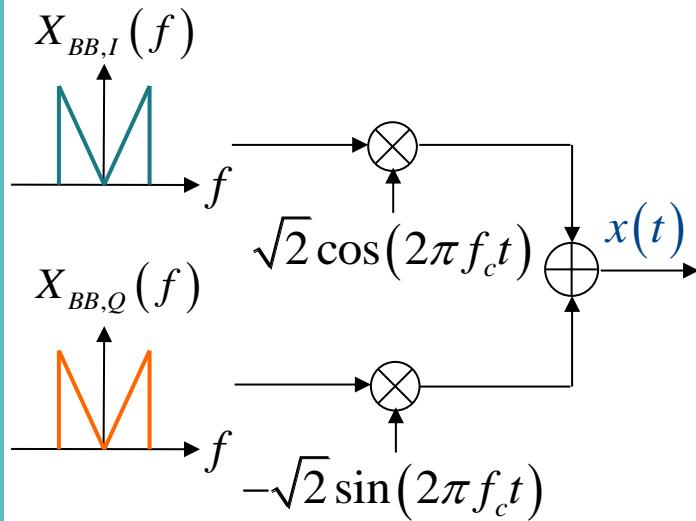
$$\tilde{r}_Q(t) = r(t)(-1)\sqrt{2} \sin(2\pi f_c t)$$

$$= -x_{BB,I}(t)\sqrt{2} \cos(2\pi f_c t)\sqrt{2} \sin(2\pi f_c t) + x_{BB,Q}(t)\sqrt{2} \sin(2\pi f_c t)\sqrt{2} \sin(2\pi f_c t)$$

eliminated by low pass filter

$$= x_{BB,I}(t)[0 + \sin(4\pi f_c t)] + x_{BB,Q}(t)[1 - \cos(4\pi f_c t)] \rightarrow \tilde{x}_{BB,Q}(t) = x_{BB,Q}(t)$$

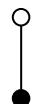
# Quadrature Amplitude Modulation (QAM) (4)



The multiplication with cosine and sine, respectively, shifts the spectrum of the signals to the carrier frequency  $f_c$ .

$$\begin{aligned} \cos(2\pi f_c t) &= \frac{1}{2} [e^{j2\pi f_c t} + e^{-j2\pi f_c t}] \\ x_{BB,I}(t) \sqrt{2} \cos(2\pi f_c t) &= \end{aligned}$$

$$= \frac{1}{\sqrt{2}} x_{BB,I}(t) e^{j2\pi f_c t} + \frac{1}{\sqrt{2}} x_{BB,I}(t) e^{-j2\pi f_c t}$$



$$\frac{1}{\sqrt{2}} X_{BB,I}(f - f_c) + \frac{1}{\sqrt{2}} X_{BB,I}(f + f_c)$$

$$\begin{aligned} \sin(2\pi f_c t) &= \frac{1}{2j} [e^{j2\pi f_c t} - e^{-j2\pi f_c t}] \\ -x_{BB,Q}(t) \sqrt{2} \sin(2\pi f_c t) &= \end{aligned}$$

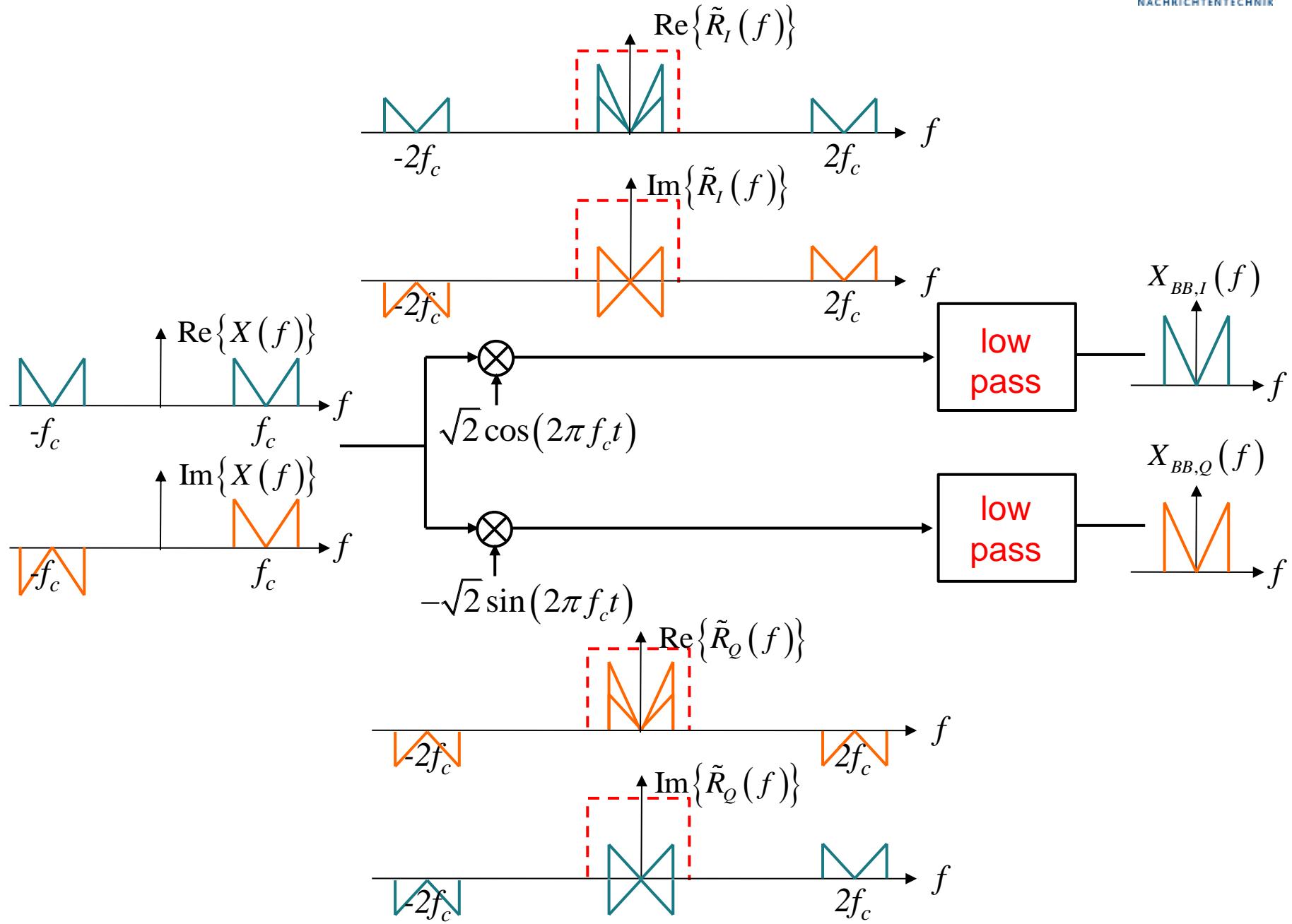
$$= \frac{j}{\sqrt{2}} x_{BB,Q}(t) e^{j2\pi f_c t} - \frac{j}{\sqrt{2}} x_{BB,Q}(t) e^{-j2\pi f_c t}$$



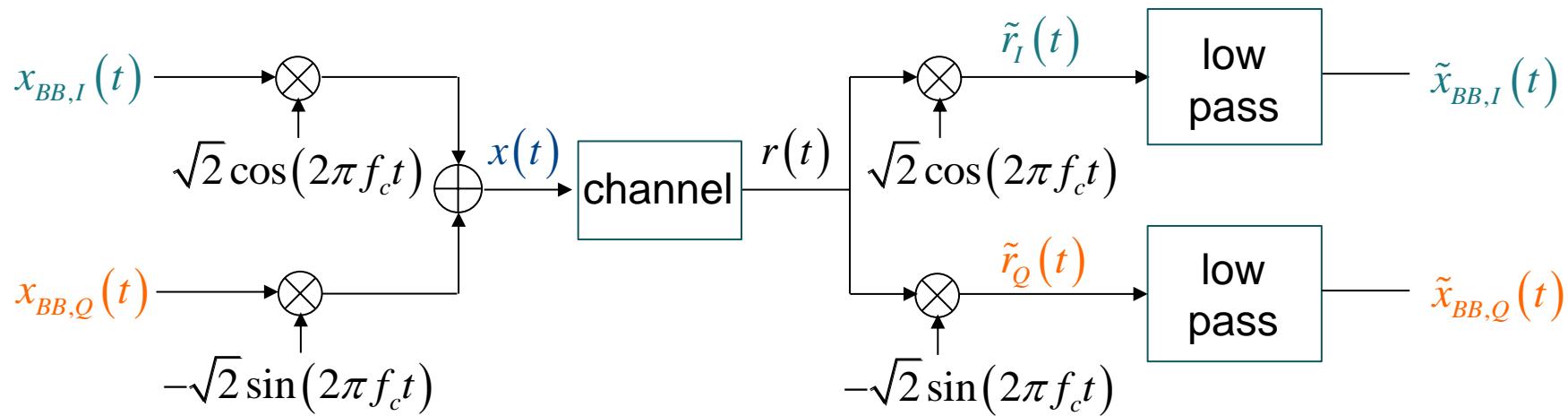
$$\frac{j}{\sqrt{2}} X_{BB,Q}(f - f_c) - \frac{j}{\sqrt{2}} X_{BB,Q}(f + f_c)$$



# Quadrature Amplitude Demodulation



# Equivalent Baseband Signal (1)



Transmit signal:  $x(t) = x_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - x_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t)$

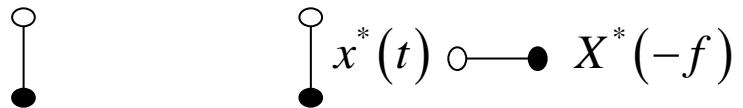
Equivalent baseband signal:  $x_{BB}(t) = x_{BB,I}(t) + jx_{BB,Q}(t)$

$$x(t) = \sqrt{2} \operatorname{Re} \left\{ x_{BB}(t) e^{j2\pi f_c t} \right\} = \operatorname{Re} \left\{ x_+(t) \right\}$$

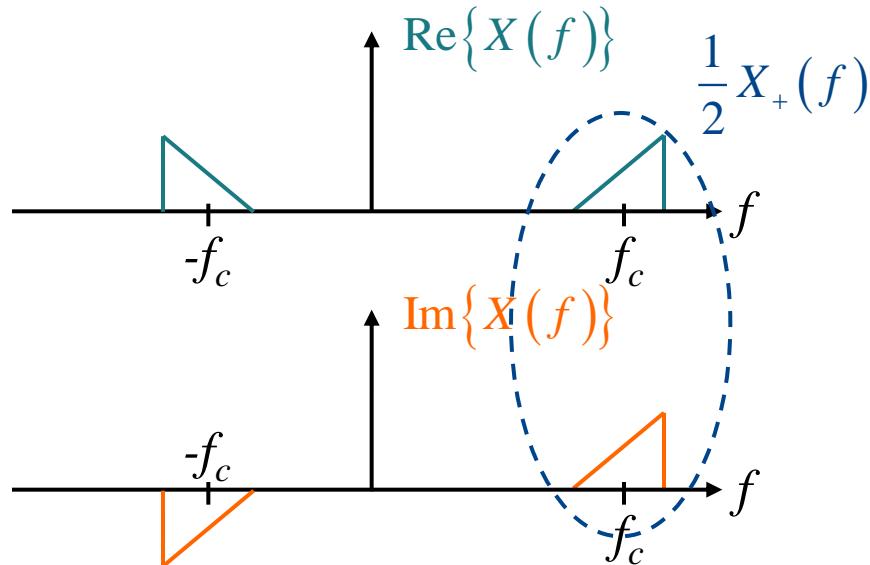
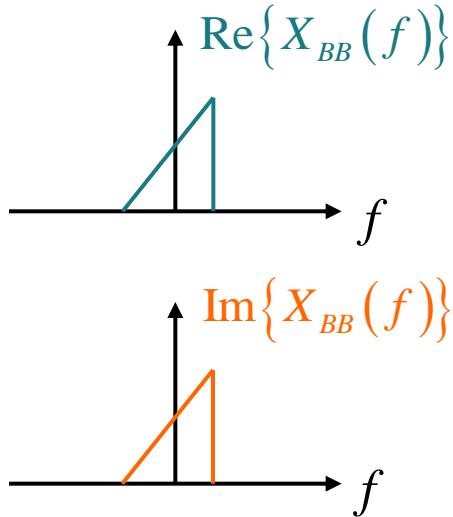
$x_+(t)$ : analytical signal

# Equivalent Baseband Signal (2)

$$x(t) = \sqrt{2} \operatorname{Re}\left\{ x_{BB}(t) e^{j2\pi f_c t} \right\} = \sqrt{2} \cdot \frac{1}{2} \left( x_{BB}(t) e^{j2\pi f_c t} + x_{BB}^*(t) e^{-j2\pi f_c t} \right)$$



$$X(f) = \frac{1}{\sqrt{2}} \left[ X_{BB}(f - f_c) + X_{BB}^*(-(f + f_c)) \right]$$



# Analytical Signal (1)

A causal signal is zero for  $t < 0$ . The analytical signal  $x_+(t)$  can be viewed as the counterpart of a causal signal in the frequency domain: The frequency domain representation  $X_+(f)$  of an analytical signal is zero for  $f < 0$ . The + is used in order to indicate that the spectrum is unequal zero only at positive frequencies.

From the symmetry properties of the spectrum  $X_+(f)$ , we can conclude that the time domain representation  $x_+(t)$  of the analytical signal must be complex. A real signal would require a conjugate even spectrum.

The relationship of real part and imaginary part of an analytical signal is given by the Hilbert transform: The imaginary part is the Hilbert transform of the real part:

$$x_+(t) = x(t) + j \mathcal{H}\{x(t)\},$$

where  $x(t)$  is real. We denote  $x_+(t)$  as the analytical signal of the real signal  $x(t)$ . The Hilbert transform is easy to explain in the frequency domain: The Hilbert transform simply multiplies the spectrum  $X(f)$  of the input signal at positive frequencies by  $-j$  and the spectrum at negative frequencies by  $j$ .  $X(0)$  is multiplied by zero. Hence, the frequency response of the Hilbert transform is given by

$$H_{\mathcal{H}}(f) = -j \operatorname{sgn}(f).$$

# Analytical Signal (2)

Consequently, the spectrum  $X_+(f)$  of the analytical signal can be obtained from  $X(f)$  by setting the spectrum to zero for  $f \leq 0$  and doubling the spectrum of  $X(f)$  for  $f > 0$ :

$$X_+(f) = X(f) + \begin{cases} j \cdot j X(f) & \text{for } f < 0 \\ -j \cdot j X(f) & \text{for } f \geq 0 \end{cases} = \begin{cases} 0 & \text{for } f < 0 \\ 2X(f) & \text{for } f \geq 0 \end{cases}$$

We will use the analytical signal in order to describe a real bandpass signal. The analytical signal has a close relation to the equivalent baseband signal  $x_{BB}(t)$  of a bandpass signal  $x(t)$ : Shifting the spectrum  $X_+(f)$  of the analytical signal to the baseband and multiplying by  $1/\sqrt{2}$  yields the spectrum  $X_{BB}(f)$  of the equivalent baseband signal:

$$x_{BB}(t) = \frac{1}{\sqrt{2}} x_+(t) e^{-j2\pi f_c t} \quad \circ \bullet \quad X_{BB}(f) = \frac{1}{\sqrt{2}} X_+(f + f_c)$$

The real bandpass signal  $x(t)$  is obtained from

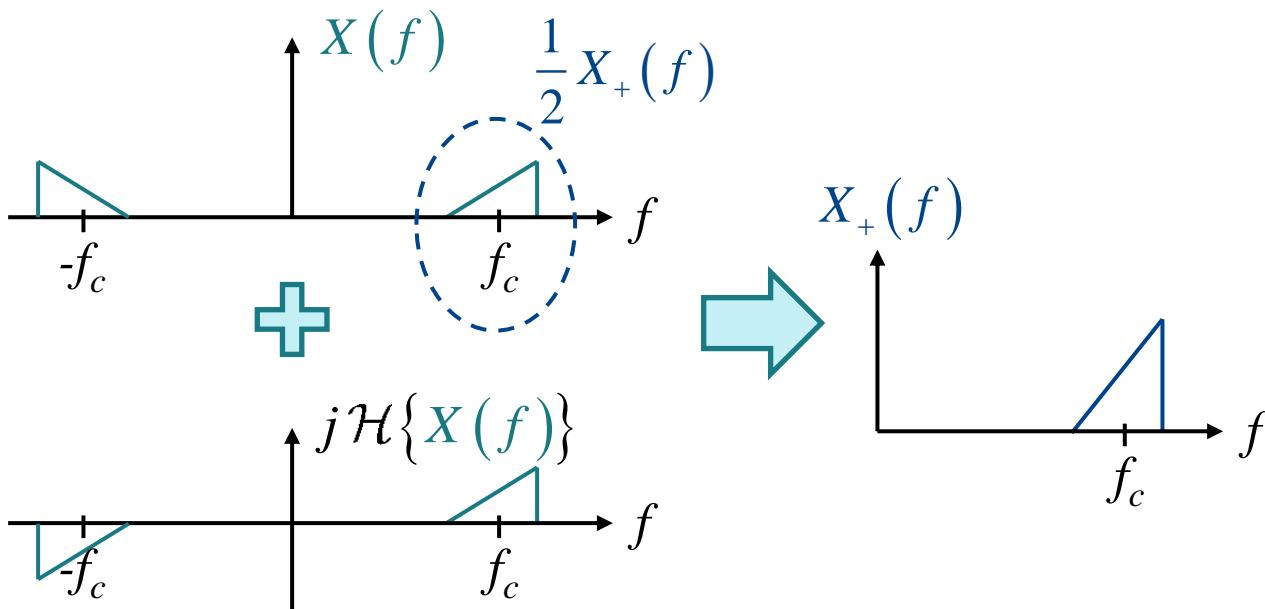
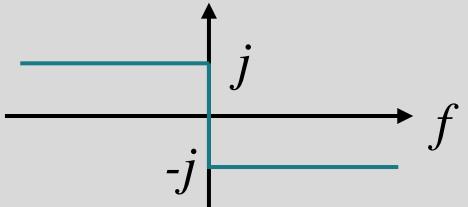
$$x(t) = \operatorname{Re}\{x_+(t)\} = \sqrt{2} \operatorname{Re}\{x_{BB}(t) e^{j2\pi f_c t}\}.$$

# Analytical Signal (1)

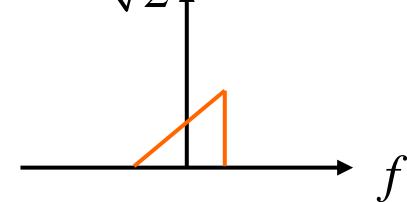
Transmit signal:  $x(t) = \operatorname{Re}\{x_+(t)\} = \sqrt{2} \operatorname{Re}\{x_{BB}(t)e^{j2\pi f_c t}\}$

Analytical signal:  $x_+(t) = x(t) + j\mathcal{H}\{x(t)\}$

Hilbert Transform  $\mathcal{H}\{.\}$ :



$$X_{BB}(f) = \frac{1}{\sqrt{2}} X_+(f + f_c)$$



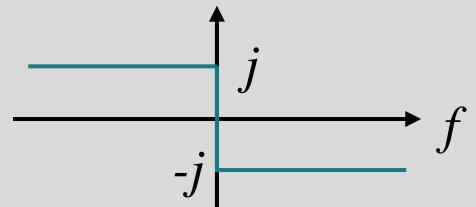
# Analytical Signal (2)

Transmit signal:  $x(t) = \operatorname{Re}\{x_+(t)\} = \sqrt{2} \operatorname{Re}\{x_{BB}(t)e^{j2\pi f_c t}\}$

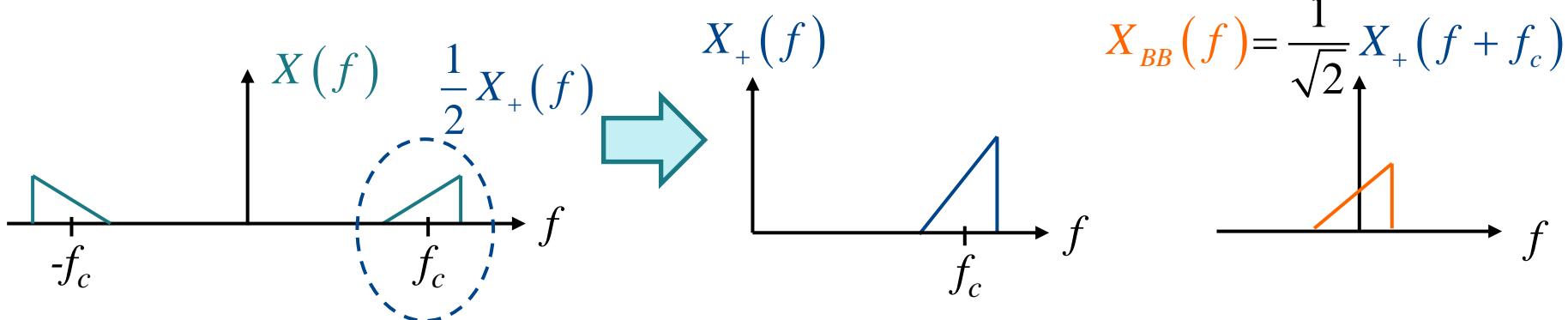
Analytical signal:  $x_+(t) = x(t) + j\mathcal{H}\{x(t)\}$



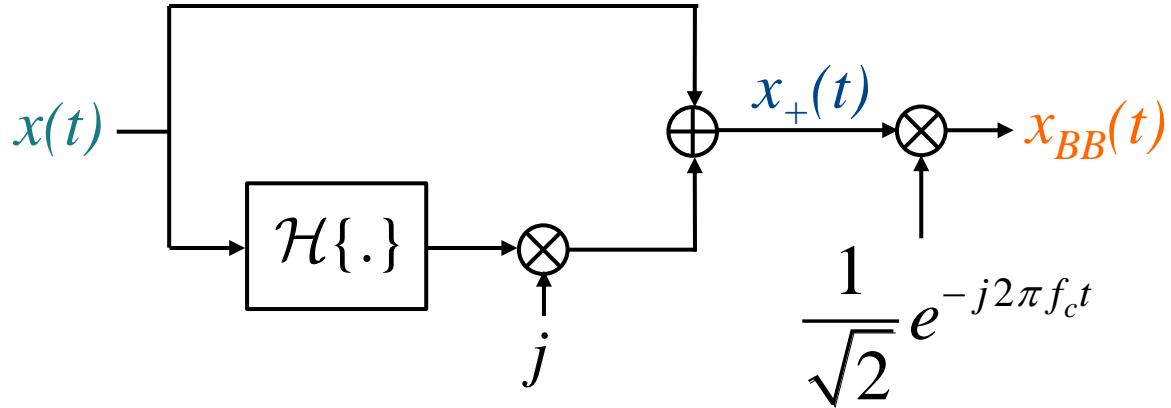
Hilbert Transform  $\mathcal{H}\{\cdot\}$ :



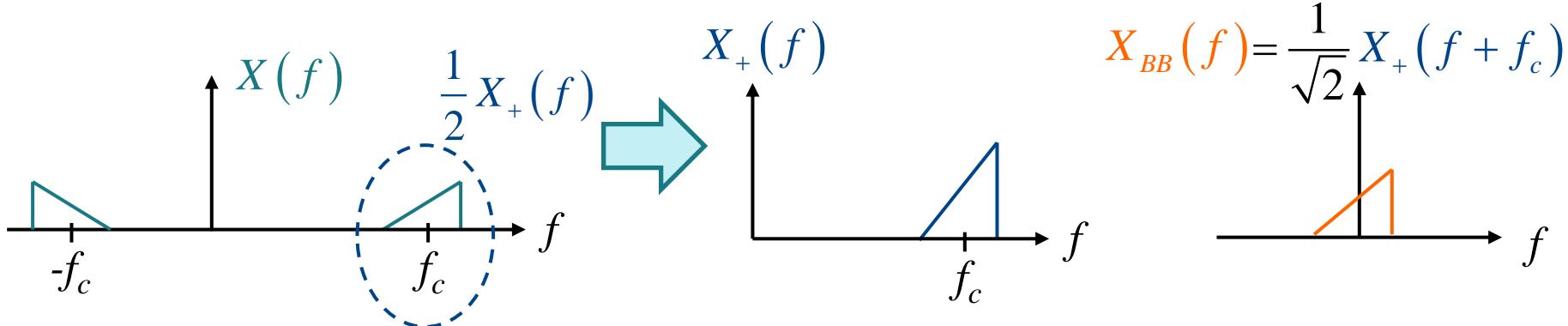
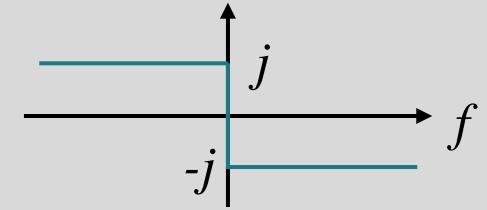
$$X_+(f) = X(f) + \begin{cases} j \cdot j X(f) & \text{for } f < 0 \\ -j \cdot j X(f) & \text{for } f \geq 0 \end{cases} = \begin{cases} 0 & \text{for } f < 0 \\ 2 X(f) & \text{for } f \geq 0 \end{cases}$$



# Generation of Equivalent Baseband Signal



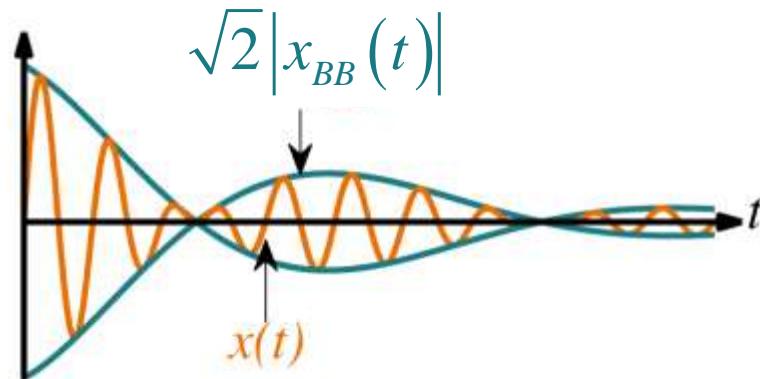
Hilbert Transform  $\mathcal{H}\{.\}$ :



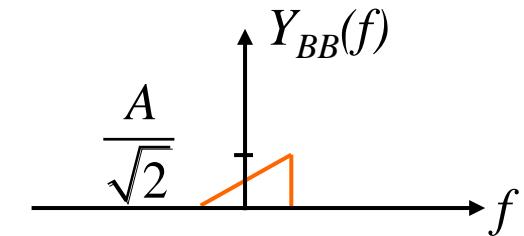
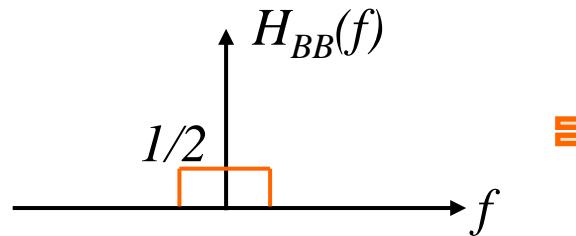
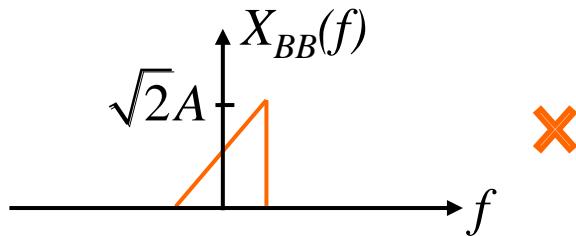
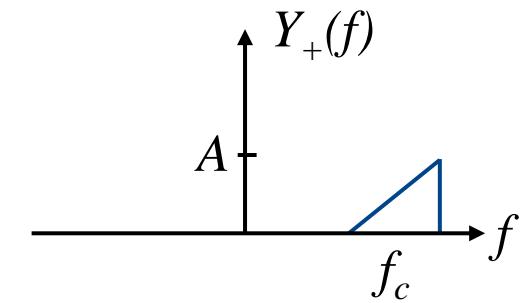
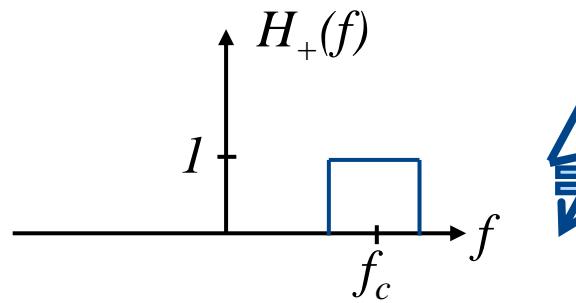
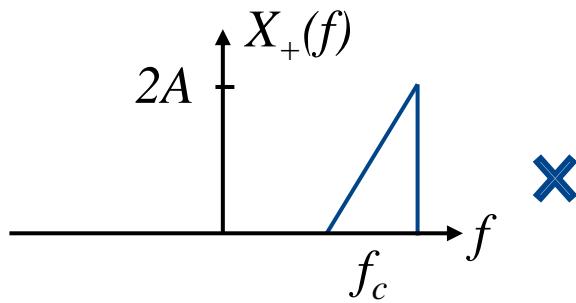
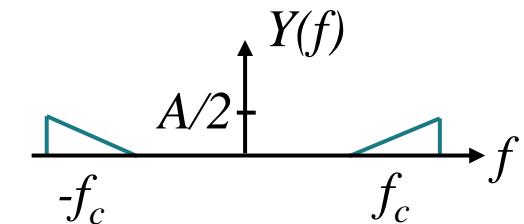
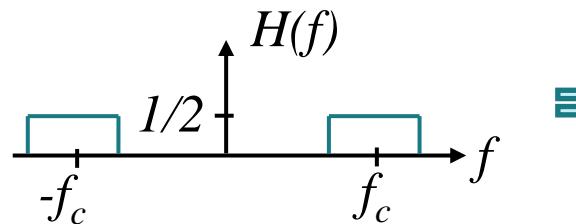
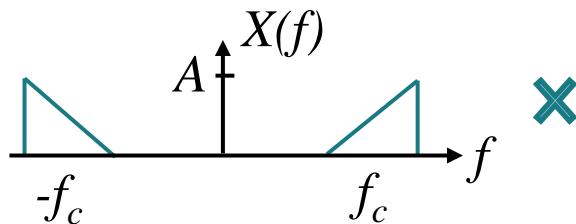
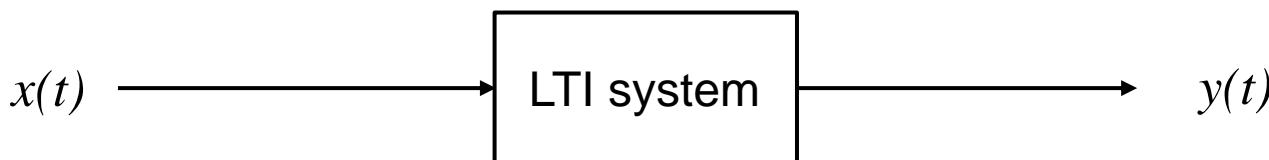
# Equivalent Baseband Signal: Complex Envelope

The magnitude  $|x_{BB}(t)|$  of the equivalent baseband signal describes the envelope of the radio frequency (RF) signal  $x(t)$ :

$$\begin{aligned}x(t) &= \sqrt{2} \operatorname{Re} \left\{ x_{BB}(t) e^{j2\pi f_c t} \right\} \\&= \sqrt{2} \operatorname{Re} \left\{ |x_{BB}(t)| e^{j(2\pi f_c t + \varphi_{BB}(t))} \right\} \\&= \sqrt{2} |x_{BB}(t)| \cos(2\pi f_c t + \varphi_{BB}(t))\end{aligned}$$

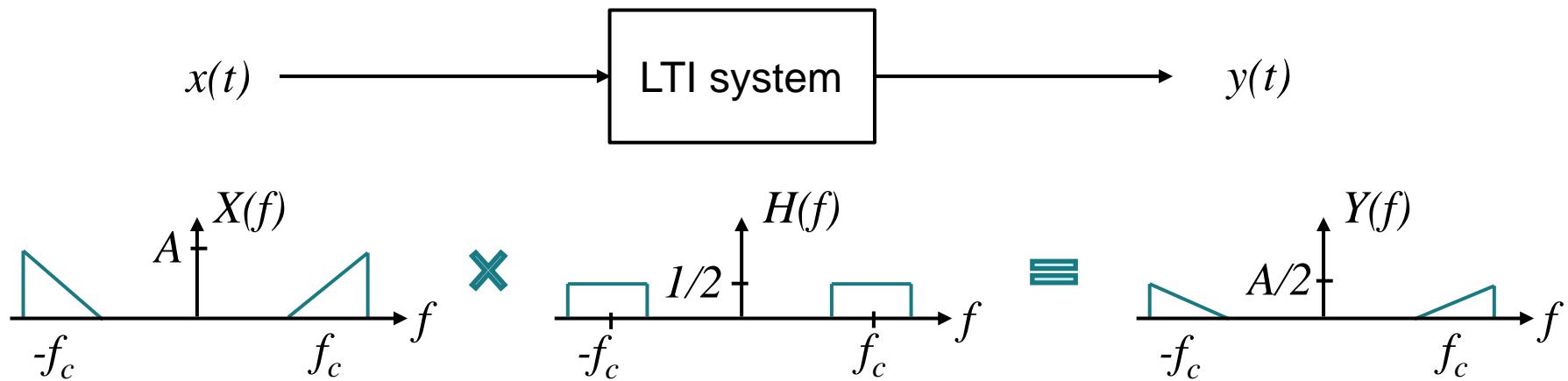


# Equivalent Baseband System (1)

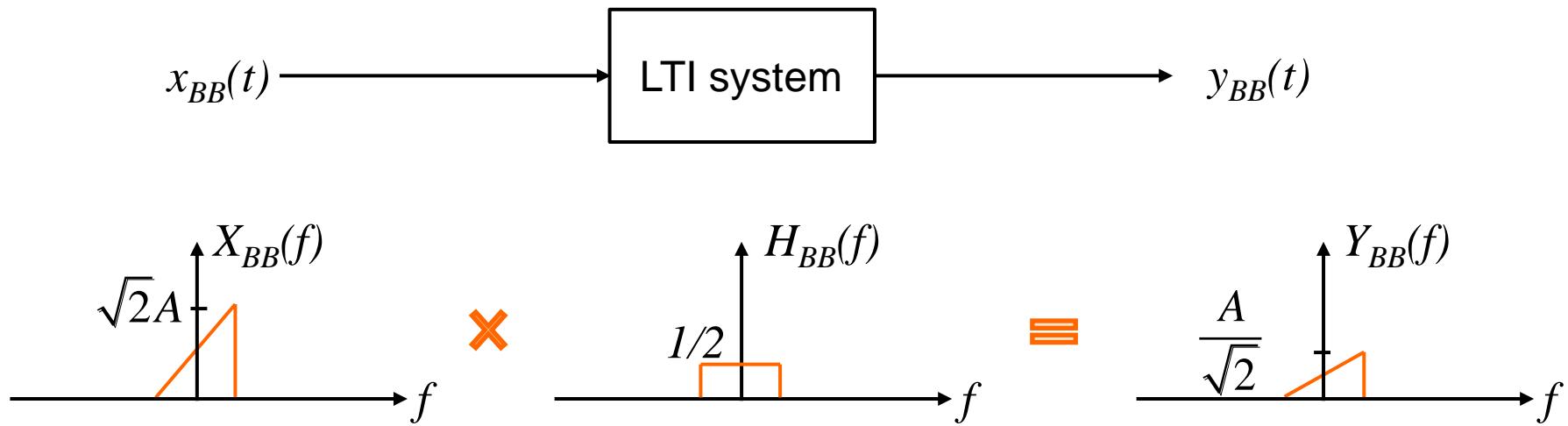


# Equivalent Baseband System (2)

**Passband** → **real** in time domain



**Equivalent baseband** → **complex** in time domain



**Signals**

$$x(t) = \sqrt{2} \operatorname{Re} \left\{ x_{BB}(t) e^{j2\pi f_c t} \right\}$$

$$= x_{BB,I}(t) \sqrt{2} \cos(2\pi f_c t) - x_{BB,Q}(t) \sqrt{2} \sin(2\pi f_c t)$$

$$= \operatorname{Re} \left\{ x_+(t) \right\}$$

**Systems**

$$h(t) = \sqrt{2} \operatorname{Re} \left\{ h_{BB}(t) e^{j2\pi f_c t} \right\}$$

$$= h_{BB,I}(t) \sqrt{2} \cos(2\pi f_c t) - h_{BB,Q}(t) \sqrt{2} \sin(2\pi f_c t)$$

$$= \operatorname{Re} \left\{ h_+(t) \right\}$$

$x(t)$ : passband signal → real in time domain

$x_{BB}(t)$ : equivalent baseband signal → complex in time domain

$x_+(t)$ : analytical signal → complex in time domain

**Signals**

$$x_{BB}(t) = \frac{1}{\sqrt{2}} [x(t) + j\mathcal{H}\{x(t)\}] e^{-j2\pi f_c t}$$

$$x_{BB}(t) = \frac{1}{\sqrt{2}} x_+(t) e^{-j2\pi f_c t}$$



$$X_{BB}(f) = \frac{1}{\sqrt{2}} X_+(f + f_c)$$

$x(t)$ : passband signal → real in time domain

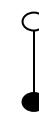
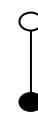
$x_{BB}(t)$ : equivalent baseband signal → complex in time domain

$x_+(t)$ : analytical signal → complex in time domain

**Systems**

$$h_{BB}(t) = \frac{1}{2} [h(t) + j\mathcal{H}\{h(t)\}] e^{-j2\pi f_c t}$$

$$h_{BB}(t) = \frac{1}{2} h_+(t) e^{-j2\pi f_c t}$$



$$H_{BB}(f) = \frac{1}{2} H_+(f + f_c)$$

# Equivalent Baseband Random Process (1)

Similar as for deterministic bandpass signals, we can describe a random bandpass process  $N(t)$  in the complex equivalent baseband.

The autocorrelation function (ACF) of the equivalent baseband process

$$N_{BB}(t) = N_{BB,I}(t) + j N_{BB,Q}(t)$$

which corresponds to the bandpass process

$$N(t) = N_{BB,I}(t)\sqrt{2} \cos(2\pi f_c t) - N_{BB,Q}(t)\sqrt{2} \sin(2\pi f_c t) = \sqrt{2} \operatorname{Re}\{N_{BB}(t)e^{j2\pi f_c t}\}$$

is given by

$$r_{N_{BB}N_{BB}}(\tau) = E\{N_{BB}^*(t)N_{BB}(t+\tau)\} = 2r_{N_{BB,I}N_{BB,I}}(\tau) + 2jr_{N_{BB,I}N_{BB,Q}}(\tau),$$

where  $r_{N_{BB,I}N_{BB,I}}(\tau)$  is the autocorrelation function of the real part (inphase component) of the equivalent baseband process and  $r_{N_{BB,I}N_{BB,Q}}(\tau)$  is the crosscorrelation function between real part and imaginary part (quadrature component) of the complex equivalent baseband process.

The autocorrelation functions of real part and imaginary part are identical, i.e.

$$r_{N_{BB,I}N_{BB,I}}(\tau) = r_{N_{BB,Q}N_{BB,Q}}(\tau)$$

and the crosscorrelation function

$$r_{N_{BB,I}N_{BB,Q}}(\tau) = -r_{N_{BB,I}N_{BB,Q}}(-\tau) = -r_{N_{BB,Q}N_{BB,I}}(\tau)$$

is odd. For white Gaussian noise, real part  $N_{BB,I}(t)$  and imaginary part  $N_{BB,Q}(t)$  are uncorrelated (even orthogonal), i.e. the crosscorrelation functions are zero:

$$r_{N_{BB,I}N_{BB,Q}}(\tau) = r_{N_{BB,Q}N_{BB,I}}(\tau) = 0$$

# Equivalent Baseband Random Process (2)

The relation between the autocorrelation function  $r_{NN}(\tau)$  of the bandpass process and the AKF  $r_{N_{BB}N_{BB}}(\tau)$  of the equivalent baseband process is given by

$$r_{NN}(\tau) = \operatorname{Re}\left\{ r_{N_{BB}N_{BB}}(\tau) e^{j2\pi f_c \tau} \right\}.$$

The ACF  $r_{NN}(\tau)$  of the real bandpass process  $N(t)$  is real and even. The ACF  $r_{N_{BB}N_{BB}}(\tau)$  of the complex equivalent baseband process  $N_{BB}(t)$  is conjugate even, i.e. its real part  $r_{N_{BB,I}N_{BB,I}}(\tau) = r_{N_{BB,Q}N_{BB,Q}}(\tau)$  is even and its imaginary part  $r_{N_{BB,I}N_{BB,Q}}(\tau) = -r_{N_{BB,Q}N_{BB,I}}(\tau)$  is odd. Consequently, the power spectral density  $S_{N_{BB}N_{BB}}(f)$  is real as any power spectral density must be.

Note, that the power spectral density  $S_{N_{BB}N_{BB}}(f)$  of the equivalent baseband process  $N_{BB}(t)$  has twice the weight of the power spectral density  $S_{NN}(f)$  of the bandpass process  $N(t)$ .

For white Gaussian noise, the power spectral density  $S_{N_{BB}N_{BB}}(f)$  of the equivalent baseband process is real and even. Hence, the ACF of white Gaussian noise is real and even.

The power of a random process can be obtained either from the bandpass representation or from the equivalent baseband representation from the AKF at  $\tau=0$  or from the integral over the power spectral density:

$$P_{N_{BB}} = P_N = r_{N_{BB}N_{BB}}(0) = r_{NN}(0) = \int_{-\infty}^{\infty} S_{N_{BB}N_{BB}}(f) df = \int_{-\infty}^{\infty} S_{NN}(f) df$$

# Equivalent Baseband White Gaussian Noise Process

For bandlimited white bandpass noise with bandwidth  $B$  and noise power spectral density

$$S_{NN}(f) = \begin{cases} \frac{N_0}{2} & \text{for } |f \pm f_c| < \frac{B}{2}, \\ 0 & \text{otherwise} \end{cases}$$

the power spectral density of the equivalent baseband process is given by

$$S_{N_{BB}N_{BB}}(f) = \begin{cases} N_0 & \text{for } |f_c| < \frac{B}{2}. \\ 0 & \text{otherwise} \end{cases}$$

Hence, we obtain for the noise power

$$P_{N_{BB}} = \sigma_{N_{BB}}^2 = P_N = \sigma_N^2 = r_{N_{BB}N_{BB}}(0) = r_{NN}(0) = \int_{-\infty}^{\infty} S_{N_{BB}N_{BB}}(f) df = \int_{-\infty}^{\infty} S_{NN}(f) df = N_0 B$$

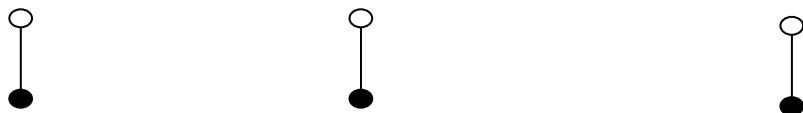
The power of the complex equivalent baseband process is uniformly split between the uncorrelated white Gaussian noise processes  $N_{BB,I}(t)$  and  $N_{BB,Q}(t)$  in real part and imaginary part, i.e.

$$\begin{aligned} P_{N_{BB,I}} = \sigma_{N_{BB,I}}^2 &= P_{N_{BB,Q}} = \sigma_{N_{BB,Q}}^2 = \frac{N_0}{2} B \\ &= r_{N_{BB,I}N_{BB,I}}(0) = r_{N_{BB,Q}N_{BB,Q}}(0) = \int_{-\infty}^{\infty} S_{N_{BB,I}N_{BB,I}}(f) df = \int_{-\infty}^{\infty} S_{N_{BB,Q}N_{BB,Q}}(f) df. \end{aligned}$$

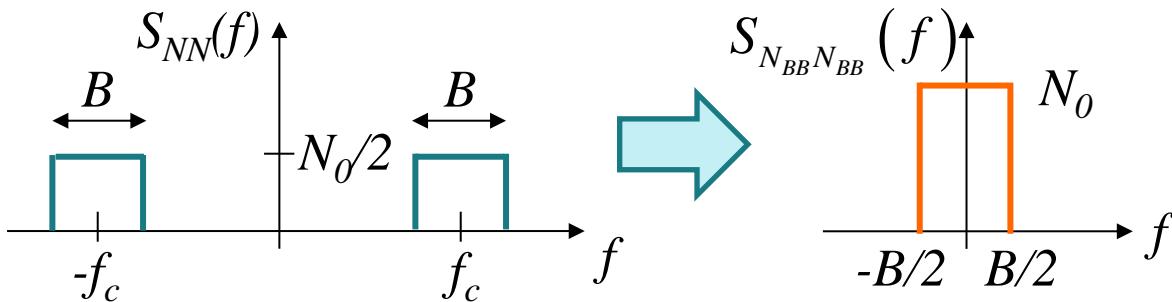
# Equivalent Baseband AWGN Channel

$$r_{NN}(\tau) = E\{N(t)N(t+\tau)\} = \operatorname{Re}\{r_{N_{BB}N_{BB}}(\tau)e^{j2\pi f_c \tau}\}$$

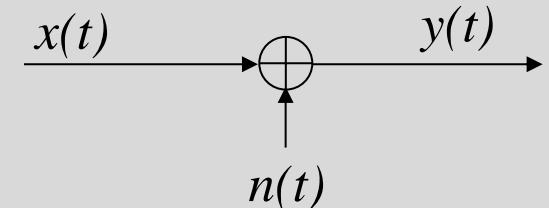
$$= \frac{1}{2} [ r_{N_{BB}N_{BB}}(\tau)e^{j2\pi f_c \tau} + r_{N_{BB}N_{BB}}^*(\tau)e^{-j2\pi f_c \tau} ]$$



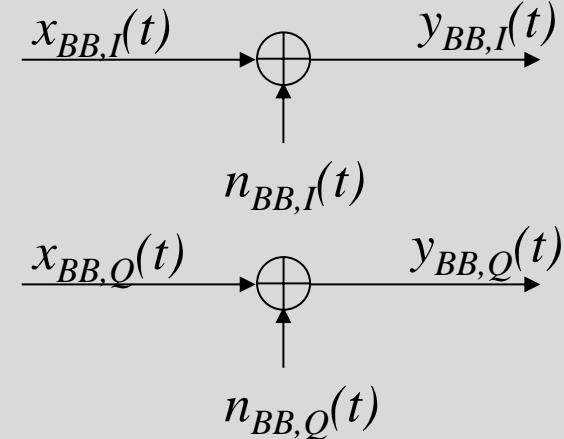
$$S_{NN}(f) = \frac{1}{2} [ S_{N_{BB}N_{BB}}(f - f_c) + S_{N_{BB}N_{BB}}(-(f + f_c)) ]$$



Real bandpass AWGN channel model:



Equivalent complex baseband AWGN channel model:



The AWGN in both quadrature components of the equivalent baseband channel is uncorrelated with variance  $\sigma_N^2 = 2\sigma_{N_{BB,I}}^2 = 2\sigma_{N_{BB,Q}}^2 = N_0 B$ .

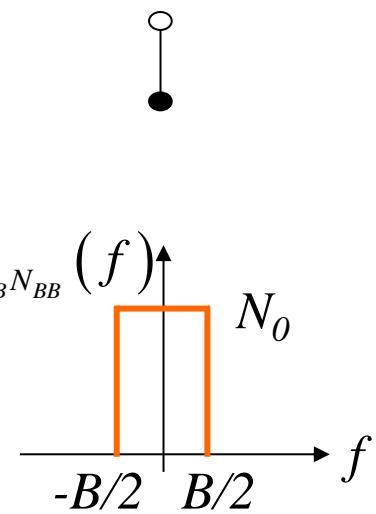
# Autocorrelation Function (ACF) of Equivalent Baseband AWGN Process

$$\begin{aligned}
 r_{N_{BB}N_{BB}}(\tau) &= E\left\{N_{BB}^*(t)N_{BB}(t+\tau)\right\} \\
 &= 2r_{N_{BB,I}N_{BB,I}}(\tau) + 2jr_{N_{BB,I}N_{BB,Q}}(\tau) = 2r_{N_{BB,I}N_{BB,I}}(\tau) \\
 &= N_0B \operatorname{si}(\pi B\tau)
 \end{aligned}$$

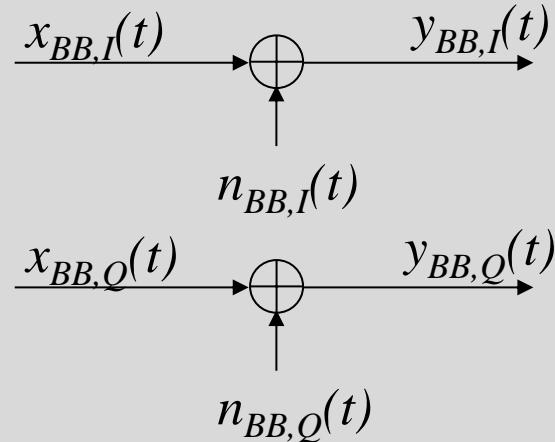
= 0 for AWGN

Power of equivalent baseband AWGN process  $N_{BB}(t)$ :

$$\begin{aligned}
 P_{N_{BB}} &= r_{N_{BB}N_{BB}}(0) \\
 &= \int_{-\infty}^{\infty} S_{N_{BB}N_{BB}}(f) df \\
 &= N_0B
 \end{aligned}$$



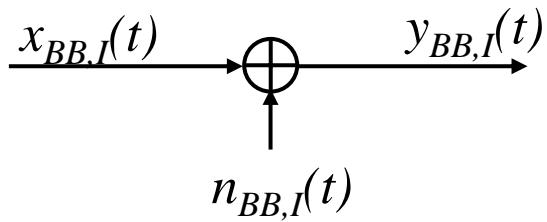
Equivalent complex baseband AWGN channel model:



The AWGN in both quadrature components of the equivalent baseband channel is uncorrelated with variance  $\sigma_N^2 = 2\sigma_{N_{BB,I}}^2 = 2\sigma_{N_{BB,Q}}^2 = N_0B$ .

# Computer Simulation of Complex Equivalent Baseband AWGN Channel

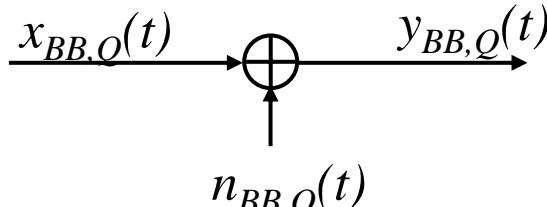
Inphase component (real part):



$$N\left(\mu_{N_{BB,I}} = 0, \sigma_{N_{BB,I}}^2 = \frac{N_0}{2} B\right)$$



Quadrature component (imaginary part):



$$N\left(\mu_{N_{BB,Q}} = 0, \sigma_{N_{BB,Q}}^2 = \frac{N_0}{2} B\right)$$

Generate two uncorrelated white Gaussian noise processes.



# Autocorrelation Function (ACF) of Bandpass and Equivalent Baseband AWGN Process

Real bandpass noise process:

$$\begin{aligned}
 N(t) &= N_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - N_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t) = \sqrt{2} \operatorname{Re}\{N_{BB}(t)e^{j2\pi f_c t}\} \\
 &= \sqrt{2} \cdot \frac{1}{2} \left[ N_{BB}(t)e^{j2\pi f_c t} + N_{BB}^*(t)e^{-j2\pi f_c t} \right] \quad (1)
 \end{aligned}$$

Complex equivalent baseband noise process:  $N_{BB}(t) = N_{BB,I}(t) + jN_{BB,Q}(t)$

Autocorrelation function of bandpass noise process:

$$r_{NN}(\tau) = E\{N(t)N(t+\tau)\}$$

$$\begin{aligned}
 &= \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2}} \left[ E\{N_{BB}(t)N_{BB}(t+\tau)\} e^{j2\pi f_c(2t+\tau)} + E\{N_{BB}^*(t)N_{BB}^*(t+\tau)\} e^{-j2\pi f_c(2t+\tau)} \right. \\
 &\quad \left. + E\{N_{BB}(t)N_{BB}^*(t+\tau)\} e^{-j2\pi f_c\tau} + E\{N_{BB}^*(t)N_{BB}(t+\tau)\} e^{j2\pi f_c\tau} \right] \\
 &\quad \underbrace{\qquad\qquad}_{r_{N_{BB}N_{BB}}^*(\tau)} \qquad \underbrace{\qquad\qquad}_{r_{N_{BB}N_{BB}}(\tau)}
 \end{aligned}$$

ACF of **stationary** process  
must be independent of  $t$

$$r_{NN}(\tau) = \frac{1}{2} \left[ r_{N_{BB}N_{BB}}(\tau) e^{j2\pi f_c\tau} + r_{N_{BB}N_{BB}}^*(\tau) e^{-j2\pi f_c\tau} \right] = \operatorname{Re}\{r_{N_{BB}N_{BB}}(\tau) e^{j2\pi f_c\tau}\}$$

# Autocorrelation Function (ACF) of Equivalent Baseband AWGN Process (1)

Complex equivalent baseband noise process:  $N_{BB}(t) = N_{BB,I}(t) + jN_{BB,Q}(t)$

$$E\{N_{BB}(t)N_{BB}(t+\tau)\} = 0 \quad (\text{from previous slide})$$

$$\Rightarrow E\{[N_{BB,I}(t) + jN_{BB,Q}(t)][N_{BB,I}(t+\tau) + jN_{BB,Q}(t+\tau)]\} = 0$$

$$\underbrace{E\{N_{BB,I}(t)N_{BB,I}(t+\tau)\}}_{r_{N_{BB,I}N_{BB,I}}(\tau)} + j \underbrace{E\{N_{BB,I}(t)N_{BB,Q}(t+\tau)\}}_{r_{N_{BB,I}N_{BB,Q}}(\tau)}$$

$$+ j \underbrace{E\{N_{BB,Q}(t)N_{BB,I}(t+\tau)\}}_{r_{N_{BB,Q}N_{BB,I}}(\tau)} - \underbrace{E\{N_{BB,Q}(t)N_{BB,Q}(t+\tau)\}}_{r_{N_{BB,Q}N_{BB,Q}}(\tau)} = 0$$

$$r_{N_{BB,Q}N_{BB,I}}(\tau) = r_{N_{BB,I}N_{BB,Q}}(-\tau) \quad r_{N_{BB,Q}N_{BB,Q}}(\tau)$$

→ Real part and imaginary part must be zero.



$$r_{N_{BB,I}N_{BB,I}}(\tau) = r_{N_{BB,Q}N_{BB,Q}}(\tau)$$

ACF in real part and imaginary part of equivalent baseband process are identical.

$$r_{N_{BB,I}N_{BB,Q}}(\tau) = -r_{N_{BB,Q}N_{BB,I}}(\tau) = -r_{N_{BB,I}N_{BB,Q}}(-\tau)$$

CCF of real part and imaginary part of equivalent baseband process is odd.

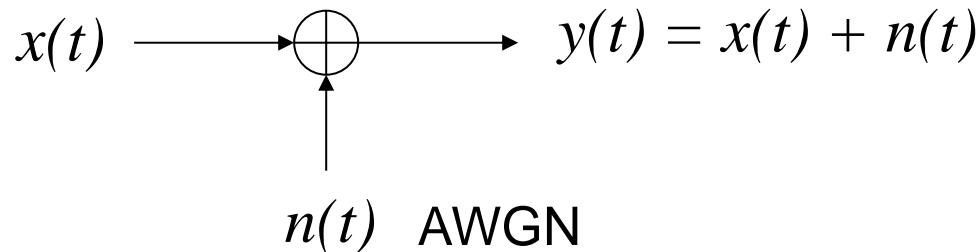
# Autocorrelation Function (ACF) of Equivalent Baseband Noise Process (2)

$$\begin{aligned}
 r_{N_{BB}N_{BB}}(\tau) &= E\left\{N_{BB}^*(t)N_{BB}(t+\tau)\right\} \\
 &= 2r_{N_{BB,I}N_{BB,I}}(\tau) + 2j r_{N_{BB,I}N_{BB,Q}}(\tau) = 2r_{N_{BB,I}N_{BB,I}}(\tau)
 \end{aligned}$$

↑  
for AWGN

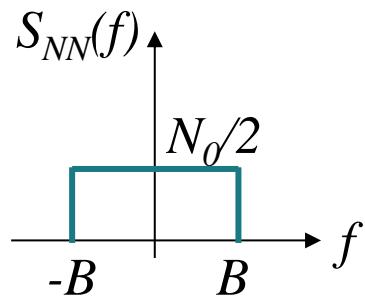
$$\begin{aligned}
 r_{N_{BB}N_{BB}}(\tau) &= E\left\{\left[N_{BB,I}(t) - jN_{BB,Q}(t)\right]\left[N_{BB,I}(t+\tau) + jN_{BB,Q}(t+\tau)\right]\right\} \\
 &= \underbrace{E\left\{N_{BB,I}(t)N_{BB,I}(t+\tau)\right\}}_{r_{N_{BB,I}N_{BB,I}}(\tau)} + j \underbrace{E\left\{N_{BB,I}(t)N_{BB,Q}(t+\tau)\right\}}_{r_{N_{BB,I}N_{BB,Q}}(\tau)} \\
 &\quad - j \underbrace{E\left\{N_{BB,Q}(t)N_{BB,I}(t+\tau)\right\}}_{-r_{N_{BB,I}N_{BB,Q}}(\tau)} + \underbrace{E\left\{N_{BB,Q}(t)N_{BB,Q}(t+\tau)\right\}}_{r_{N_{BB,Q}N_{BB,Q}}(\tau)} \\
 r_{N_{BB,Q}N_{BB,I}}(\tau) &= -r_{N_{BB,I}N_{BB,Q}}(\tau) \quad r_{N_{BB,Q}N_{BB,Q}}(\tau) = r_{N_{BB,I}N_{BB,I}}(\tau) \\
 &= 2r_{N_{BB,I}N_{BB,I}}(\tau) + 2j \underbrace{r_{N_{BB,I}N_{BB,Q}}(\tau)}_{=0 \text{ for AWGN}} \\
 &= 0 \text{ for AWGN (since power spectral density } S_{N_{BB}N_{BB}}(f) \text{ is real and even)}
 \end{aligned}$$

# AWGN Channel Models



## Baseband model

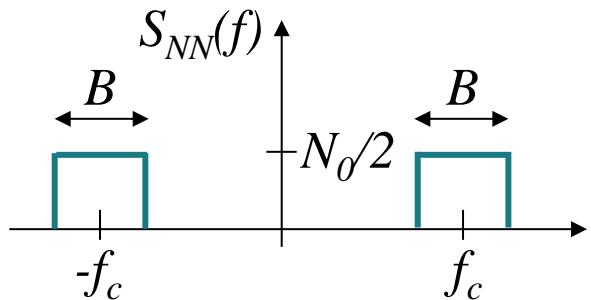
$x(t), n(t), y(t)$  real



$$P_N = N_0 B$$

## Bandpass model

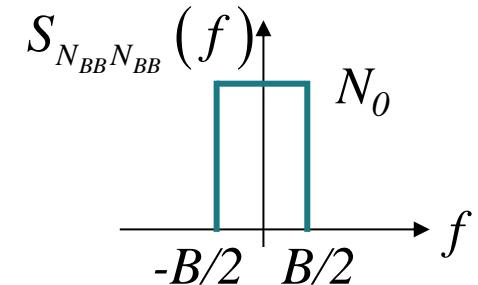
$x(t), n(t), y(t)$  real



$$P_N = \frac{N_0}{2} \cdot 2B = N_0 B$$

## Equivalent baseband model

$x_{BB}(t), n_{BB}(t), y_{BB}(t)$  complex



$$P_N = N_0 B$$

=

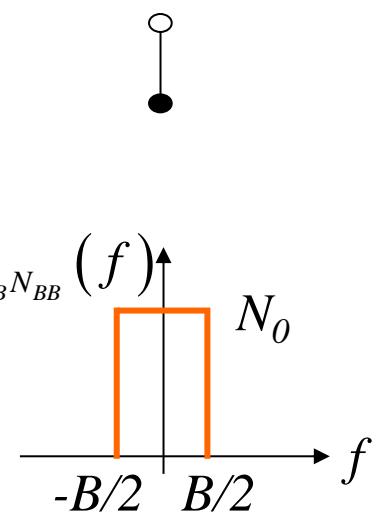
# Autocorrelation Function (ACF) of Equivalent Baseband AWGN Process

$$\begin{aligned}
 r_{N_{BB}N_{BB}}(\tau) &= E\left\{N_{BB}^*(t)N_{BB}(t+\tau)\right\} \\
 &= 2r_{N_{BB,I}N_{BB,I}}(\tau) + 2jr_{N_{BB,I}N_{BB,Q}}(\tau) = 2r_{N_{BB,I}N_{BB,I}}(\tau) \\
 &= N_0B \operatorname{si}(\pi B\tau)
 \end{aligned}$$

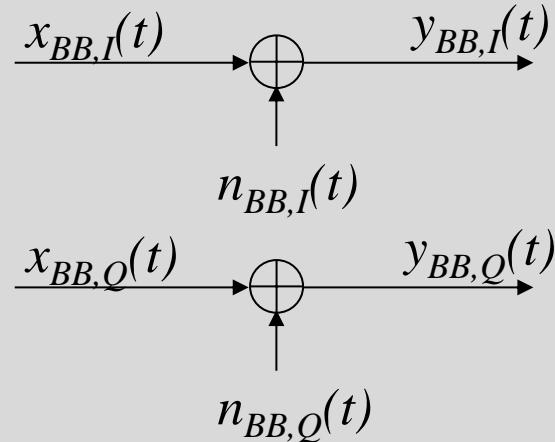
= 0 for AWGN

Power of equivalent baseband AWGN process  $N_{BB}(t)$ :

$$\begin{aligned}
 P_{N_{BB}} &= r_{N_{BB}N_{BB}}(0) \\
 &= \int_{-\infty}^{\infty} S_{N_{BB}N_{BB}}(f) df \\
 &= N_0B
 \end{aligned}$$

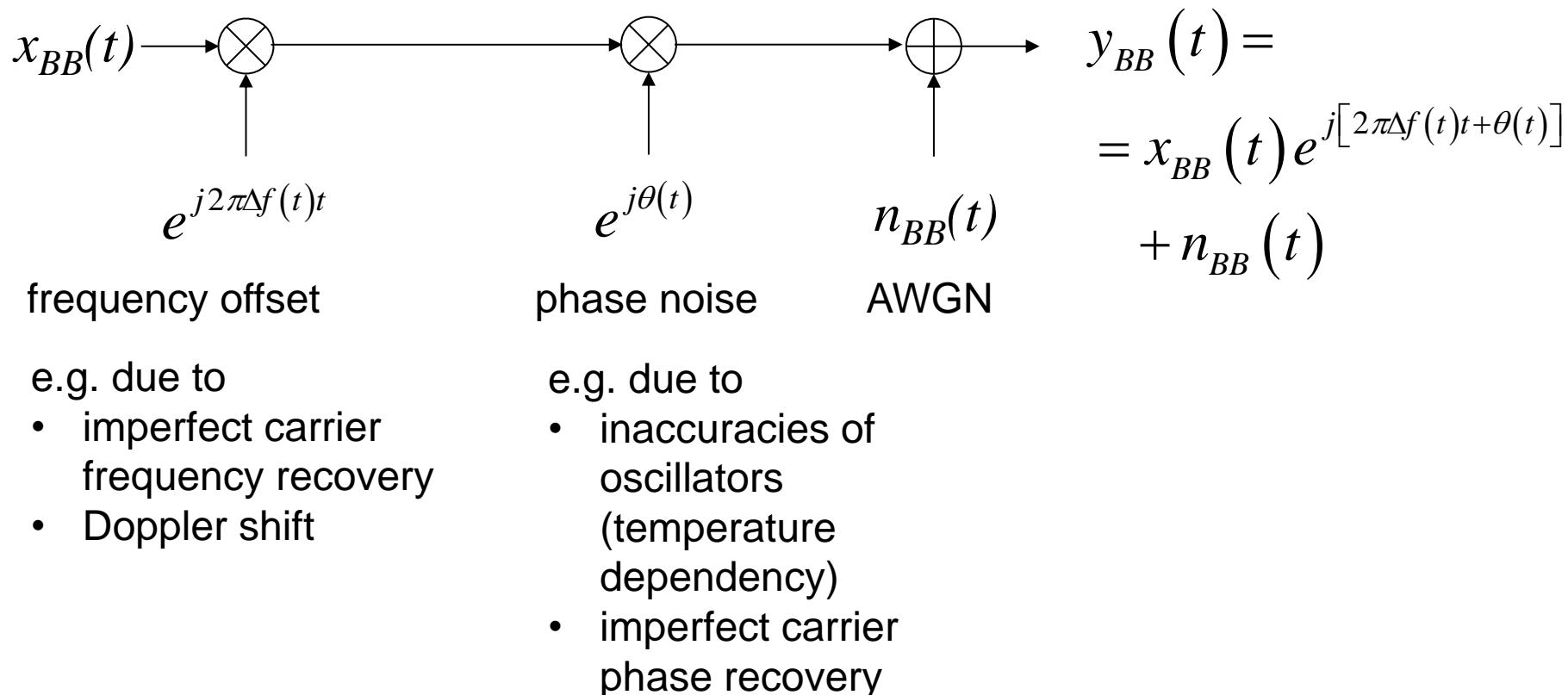


Equivalent complex baseband AWGN channel model:



The AWGN in both quadrature components of the equivalent baseband channel is uncorrelated with variance  $\sigma_N^2 = 2\sigma_{N_{BB,I}}^2 = 2\sigma_{N_{BB,Q}}^2 = N_0B$ .

# Equivalent Baseband Channel Model with Frequency-Offset and Phase Noise



# Rayleigh Fading Channel Model

A frequency-flat wireless multipath channel without a dominating line-of-sight (LOS) component can be accurately modelled in the equivalent baseband by a Rayleigh fading model. The channel coefficient  $h(t)$  is a complex Gaussian distributed random variable, where real part and imaginary part are statistically independent with zero mean and variance  $\sigma_h^2 = \frac{1}{2}$ . The variance of  $\frac{1}{2}$  makes sure that the channel on average does neither amplify nor attenuate the transmitted signal, i.e.  $E\{|h|^2\} = 1$ .

Modelling a multipath channel by a complex Gaussian random variable is justified by the central limit theorem, which states that the probability distribution of a random variable, which is the sum of  $N$  statistically independent random variables, approaches a Gaussian distribution as the number  $N$  of random variables approaches infinity. The idea is, that the channel coefficient  $h$  results from a large number of superimposed multipath components. Note, that in this model, the propagation delay differences of the multipath components are assumed to be negligible.

The magnitude  $|h|$  of a complex random variable  $h$  is Rayleigh distributed. This explains the name “Rayleigh fading.” A Rayleigh distribution is characterized by a high probability of low amplitudes  $|h| < 1$ , i.e. a high probability for deep fades in the received signal. Consequently, the transmission quality over a Rayleigh fading channel is typically poor and more sophisticated transmission schemes are required for reliable transmission of data over a Rayleigh fading channel.

The phase  $\varphi_h$  of a complex Gaussian random variable with statistically independent components in real and imaginary part is uniformly distributed between  $0$  and  $2\pi$ .

# Rice Channel Model

In scenarios, where a line-of-sight (LOS) exists between transmitter and receiver, the line-of-sight component in the multipath channel is typically much stronger than the other multipath components which have been reflected at objects and usually propagate via a longer path.

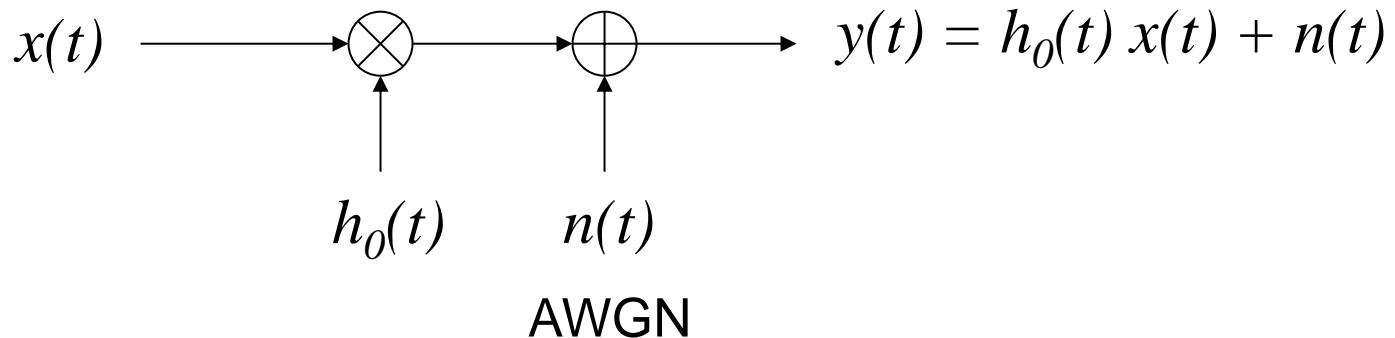
Such a line-of-sight wireless channel is appropriately modelled by a Rician channel model. Here, the channel coefficient  $h$  is characterized by a Rice distribution, i.e. it consists of two components:

A static component  $h_{LOS}$  which models the line-of-sight component and a Rayleigh distributed non-line-of-sight (NLOS) component  $h_{NLOS}$ , which models the remaining multipath components. The Rice factor

$$K = \frac{P_{LOS}}{P_{NLOS}}$$

defines the power ratio of LOS and NLOS component. The special case  $K=0$  yields a Rayleigh fading channel model, while the special case  $K \rightarrow \infty$  yields an AWGN channel model.

# Rayleigh Fading Channel Model (1)



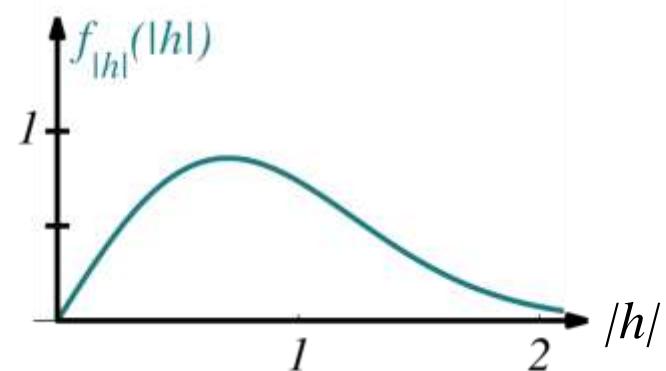
$$h_0(t) = \underbrace{\text{Re}\{h_0(t)\}}_{\text{Real part}} + j \underbrace{\text{Im}\{h_0(t)\}}_{\text{Imaginary part}}$$

$$\xleftarrow{N\left(0, \frac{1}{2}\right)}$$

$$= \underbrace{|h_0(t)|}_{\text{Magnitude}} e^{j\varphi_{h_0}(t)}$$

$$f_{|h_0|}(|h_0|) = 2|h_0| e^{-|h_0|^2}$$

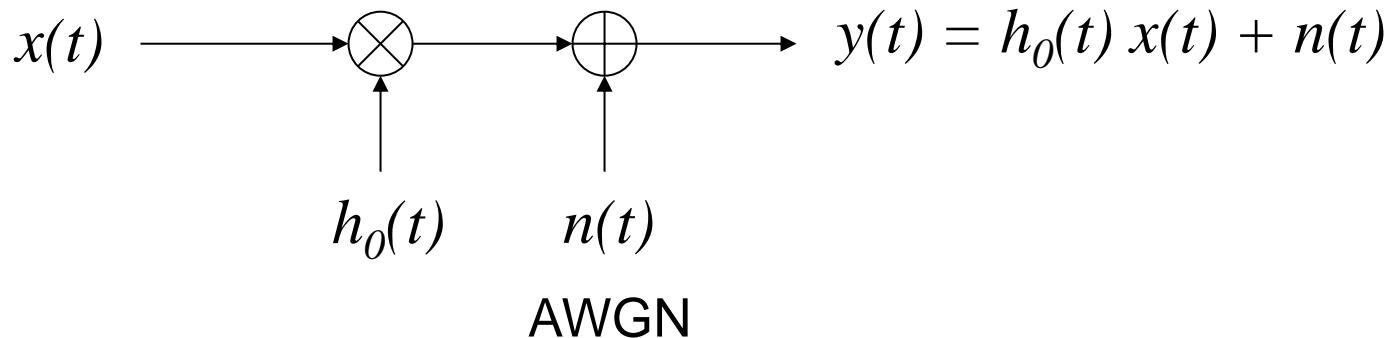
Rayleigh distribution



$$f_{\varphi_{h_0}}(\varphi_{h_0})$$



# Rayleigh Fading Channel Model (2)



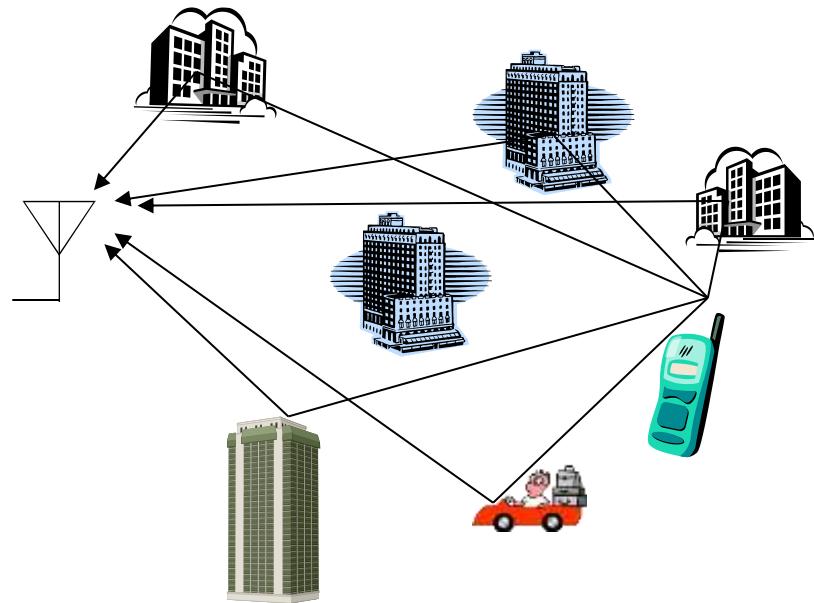
$$h_0(t) = \underbrace{\operatorname{Re}\{h_0(t)\}}_{\sim N(0, \frac{1}{2})} + j \underbrace{\operatorname{Im}\{h_0(t)\}}_{\sim N(0, \frac{1}{2})} = \lim_{N \rightarrow \infty} \frac{1}{\sqrt{N}} \sum_{n=1}^N e^{j[\theta_n + 2\pi f_{D,n} t]}$$

$$\sim N\left(0, \frac{1}{2}\right)$$

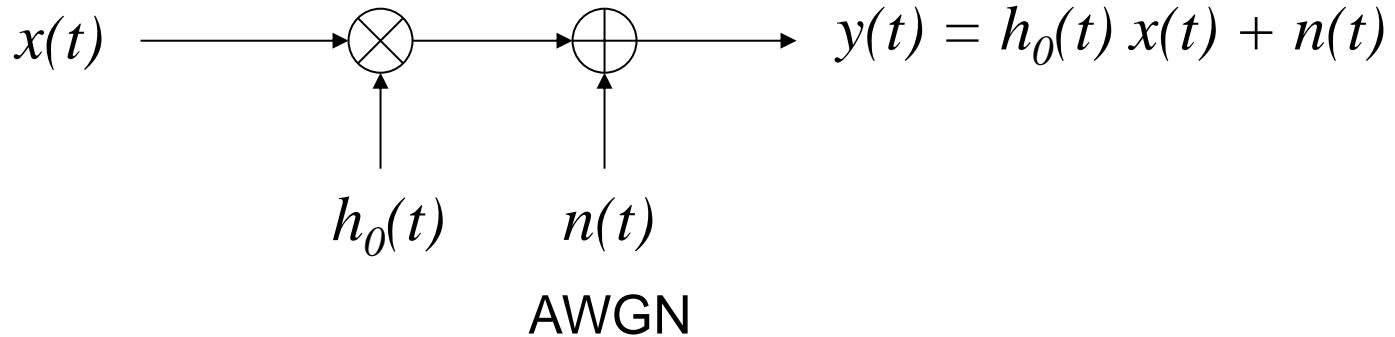
$$= \underbrace{|h_0(t)|}_{\sim \text{Rayleigh distribution}} e^{j\varphi_{h_0}(t)}$$

$$f_{|h_0|}(|h_0|) = 2|h_0| e^{-|h_0|^2}$$

Rayleigh distribution



# Rice Fading Channel Model (1)

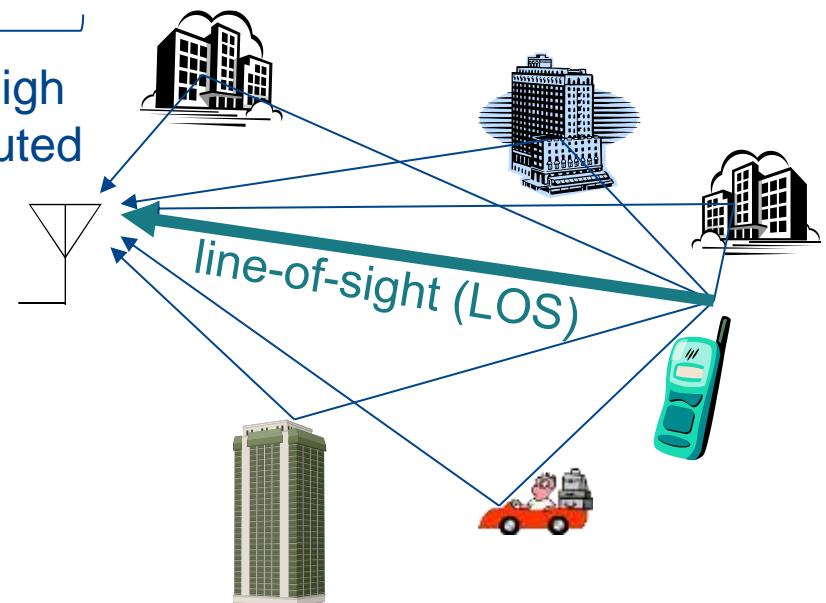


$$h_0(t) = \underbrace{\sqrt{\frac{K}{1+K}} h_{\text{LOS}}(t)}_{\text{static}} + \underbrace{\sqrt{\frac{1}{1+K}} h_{\text{NLOS}}(t)}_{\text{Rayleigh distributed}}$$

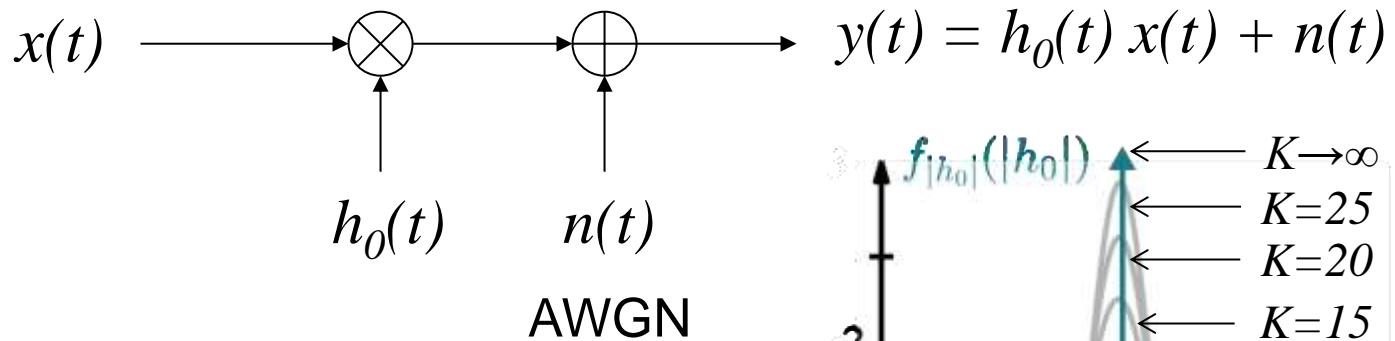
Rician  $K$ -factor:

$$K = \frac{P_{\text{LOS}}}{P_{\text{NLOS}}}$$

(power ratio of LOS and NLOS components)

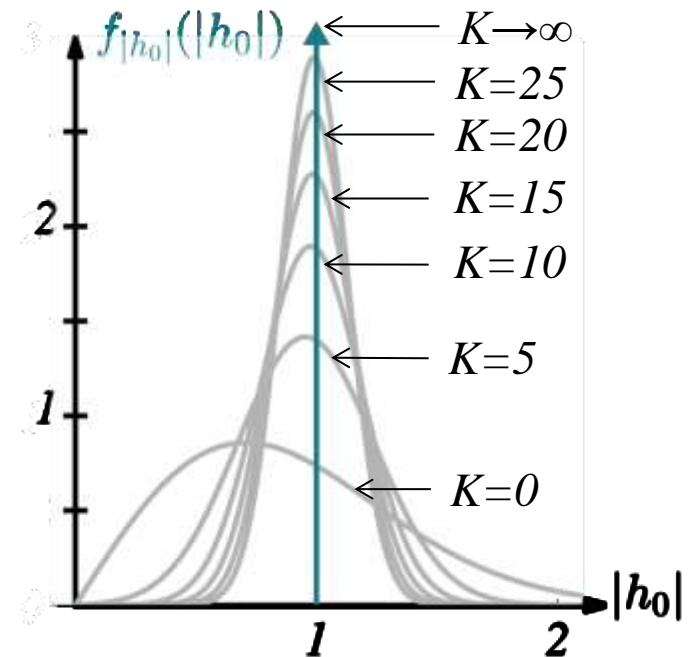


# Rice Fading Channel Model (2)



$$h_0(t) = \sqrt{\frac{K}{1+K}} h_{\text{LOS}}(t) + \sqrt{\frac{1}{1+K}} h_{\text{NLOS}}(t)$$

$\underbrace{\phantom{0}}_{\text{static}}$        $\underbrace{\phantom{0}}_{\text{Rayleigh distributed}}$



Rice distribution:

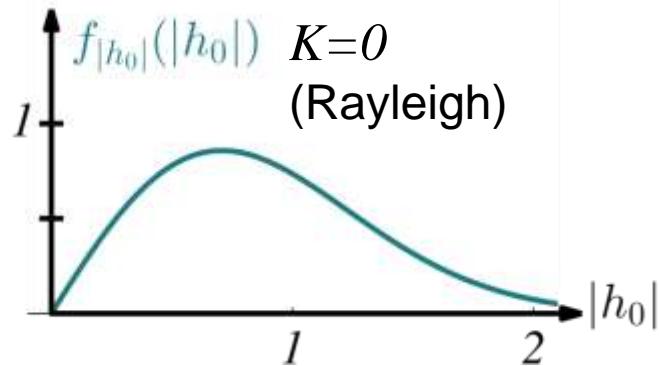
$$f_{|h_0|}(|h_0|) = 2|h_0|(1+K)e^{-K-|h_0|^2(1+K)} I_0\left(2|h_0|\sqrt{K(1+K)}\right)$$

↑

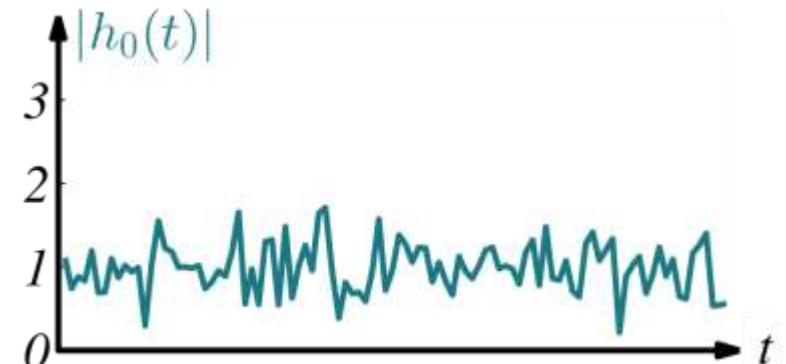
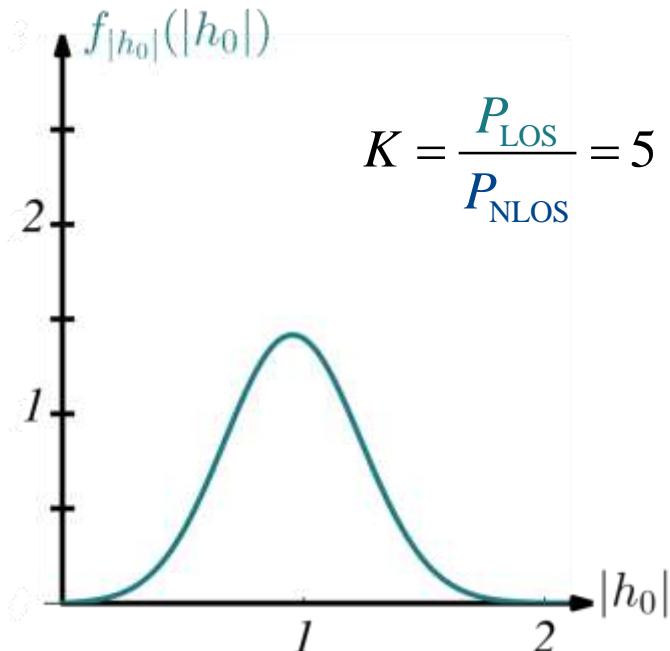
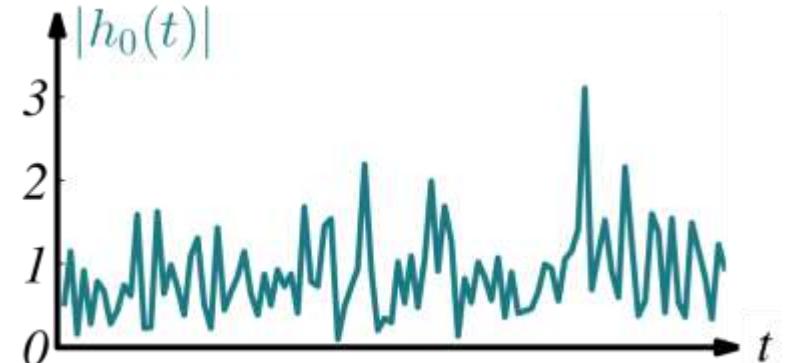
modified Bessel function of order 0

# Rice Fading Channel Model (3)

Probability density function (PDF)

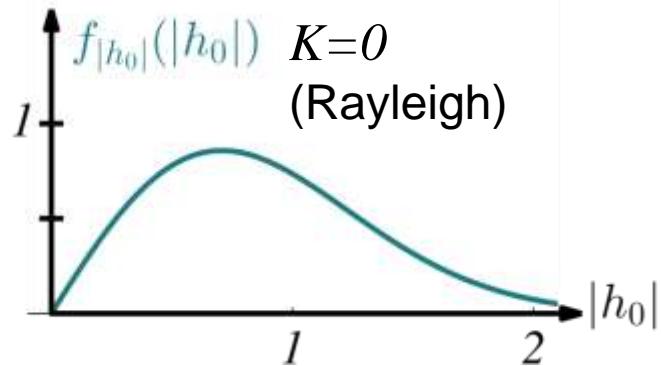


Time domain signal

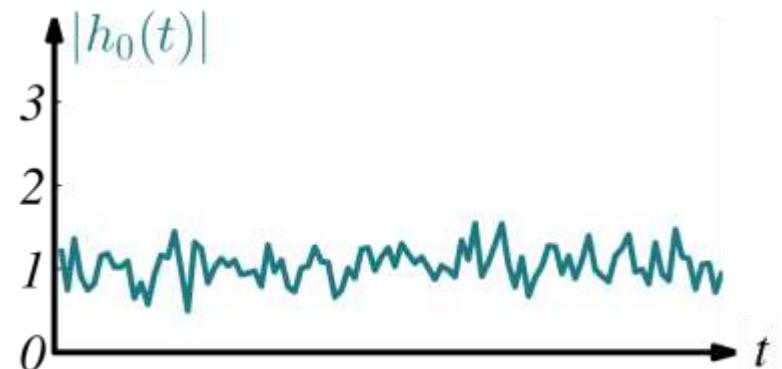
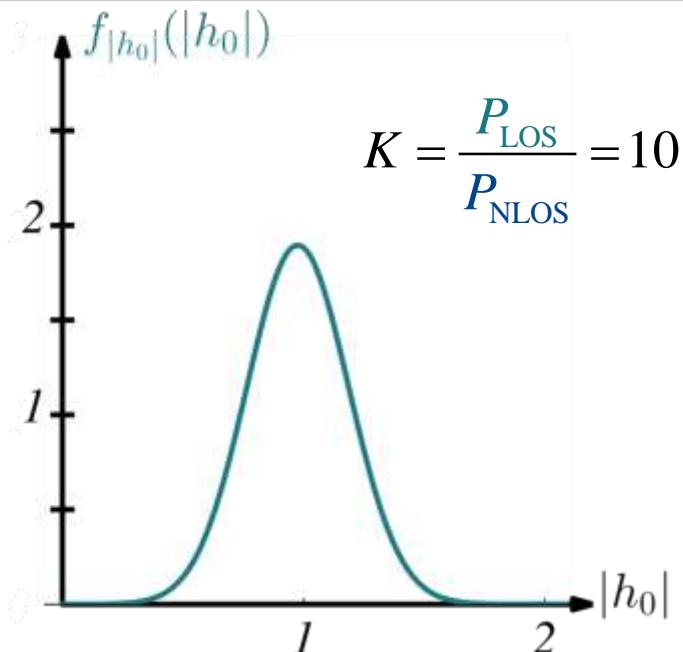
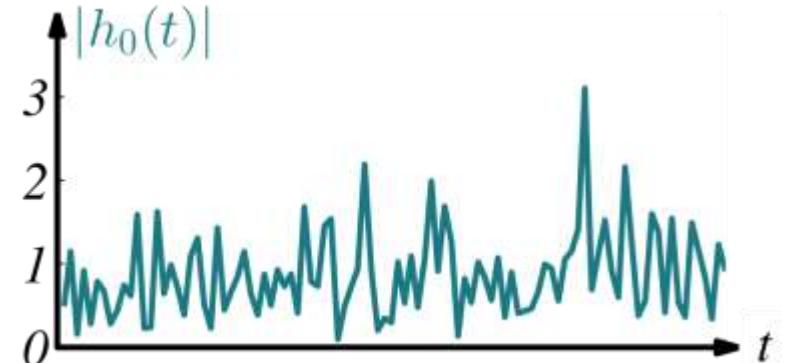


# Rice Fading Channel Model (4)

Probability density function (PDF)

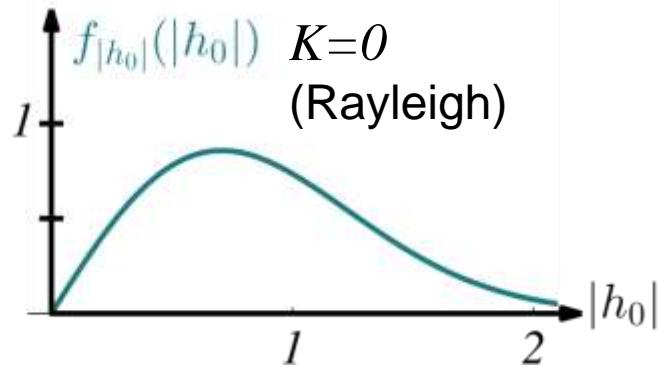


Time domain signal

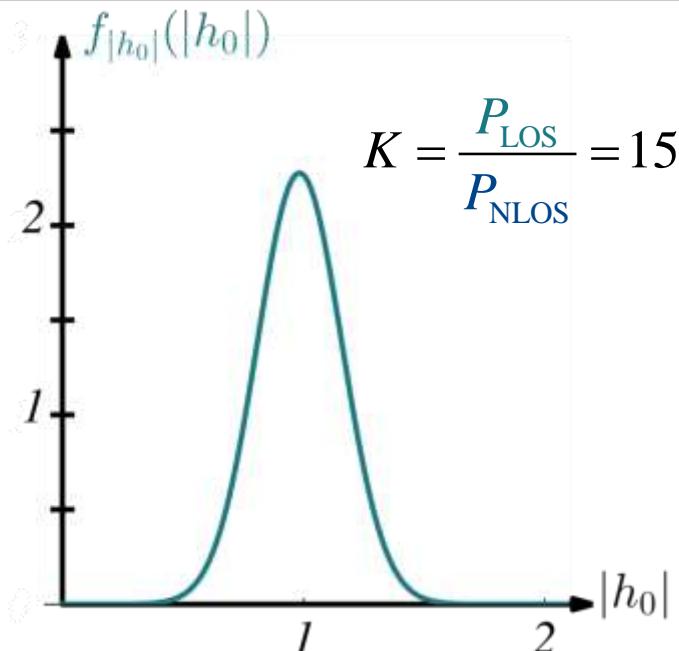
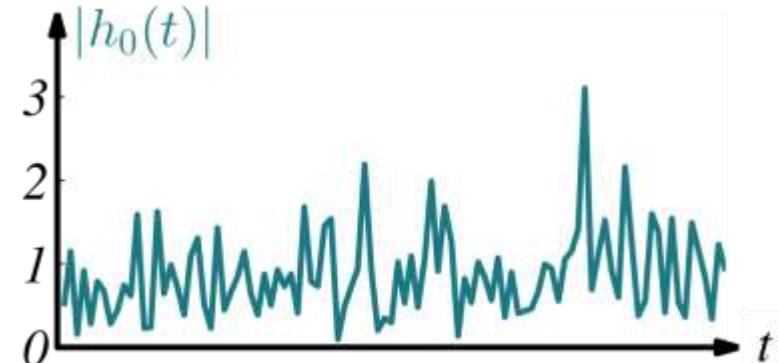


# Rice Fading Channel Model (5)

Probability density function (PDF)

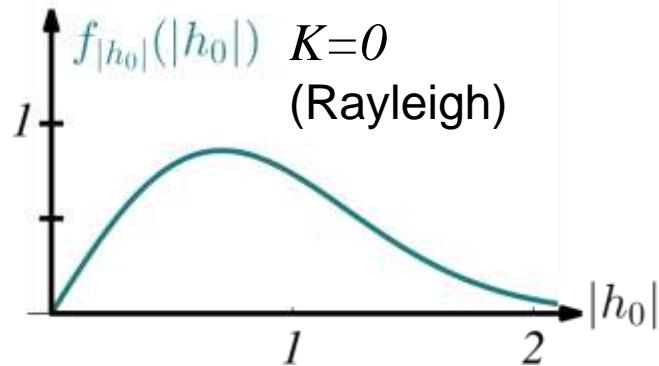


Time domain signal

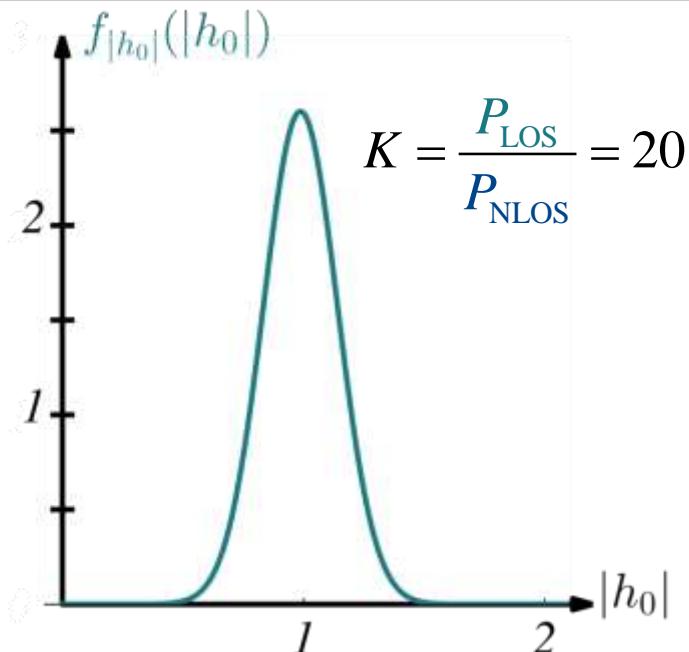
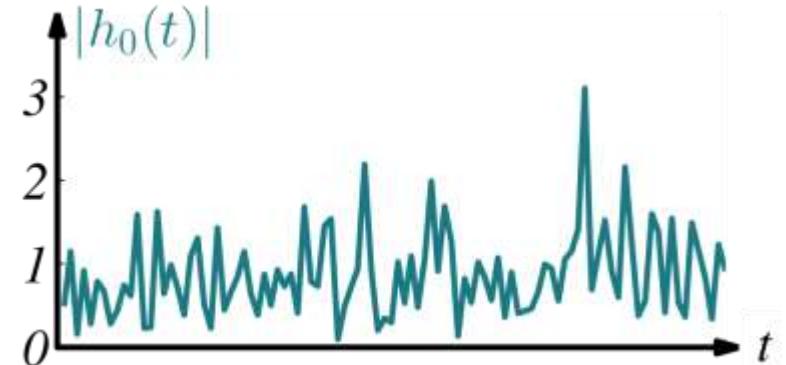


# Rice Fading Channel Model (6)

Probability density function (PDF)

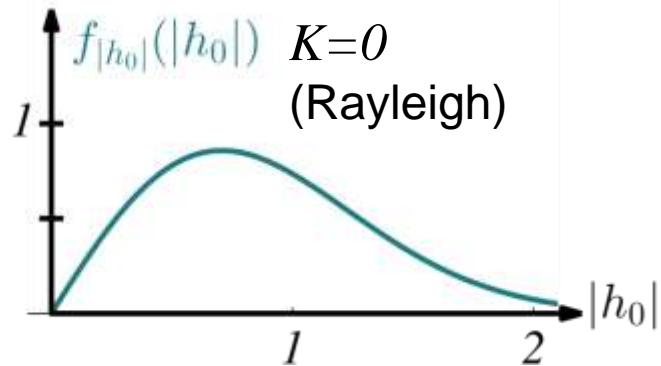


Time domain signal

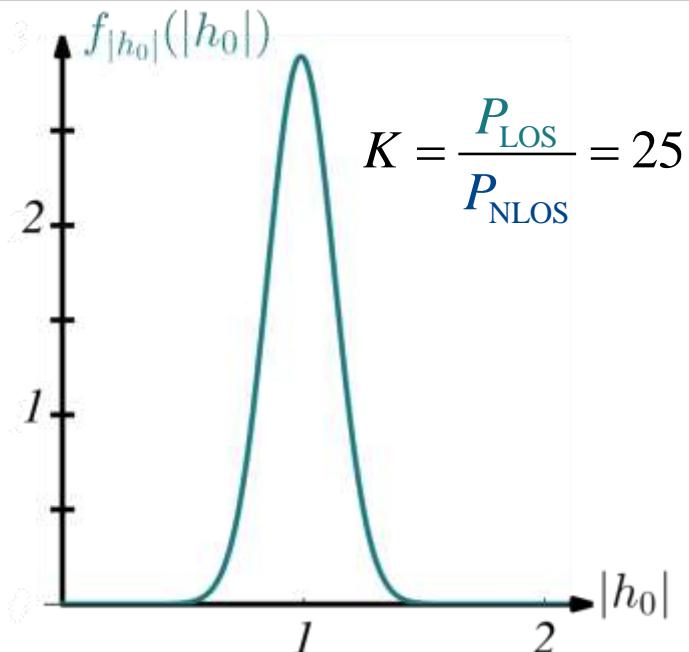
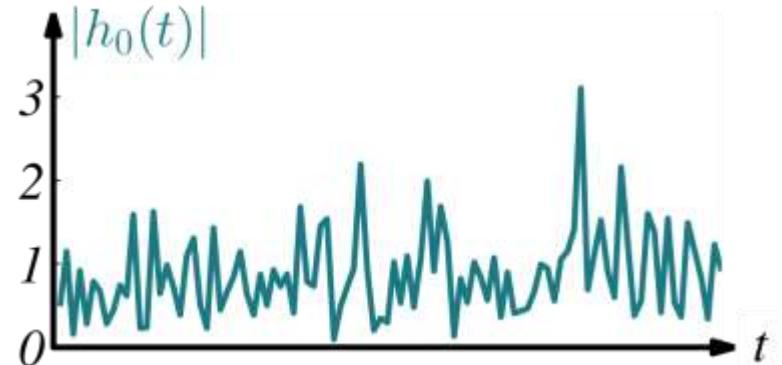


# Rice Fading Channel Model (7)

Probability density function (PDF)

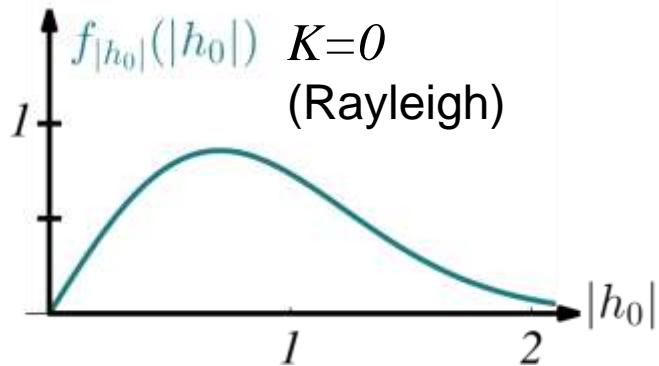


Time domain signal

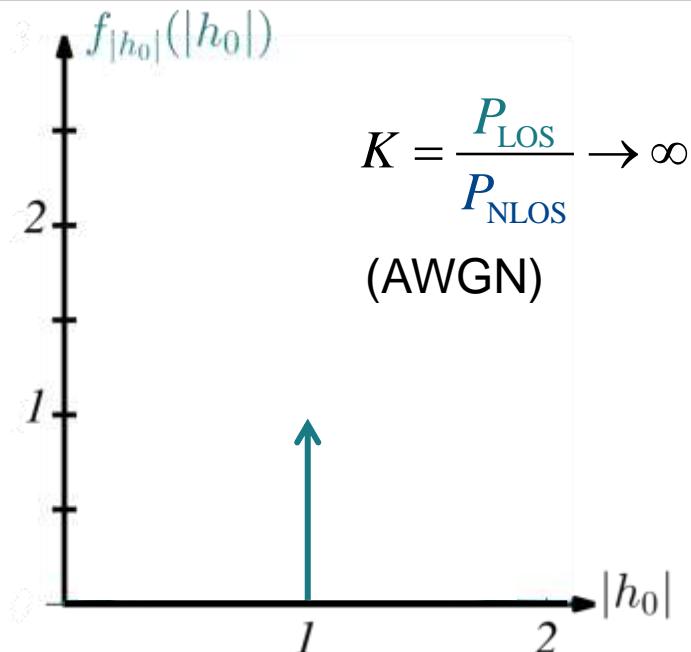
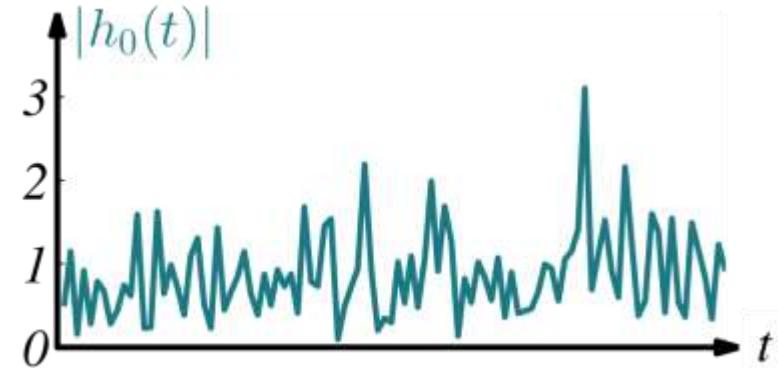


# Rice Fading Channel Model (8)

Probability density function (PDF)

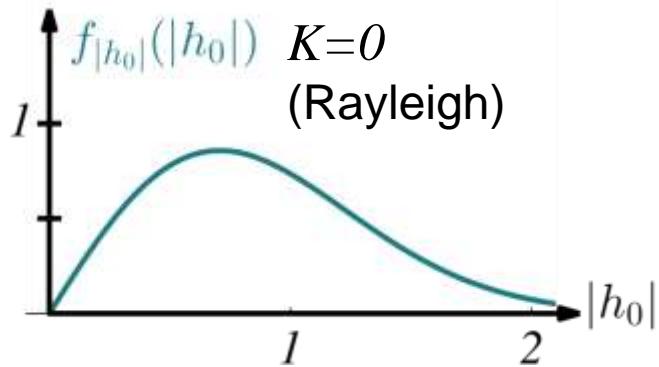


Time domain signal

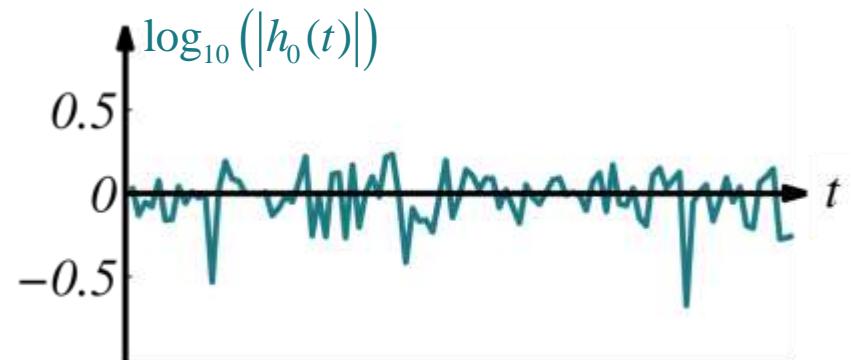
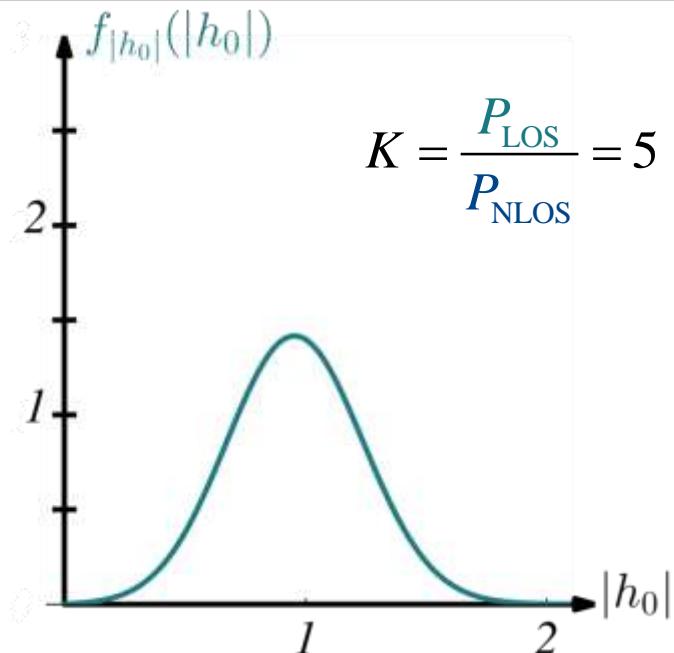
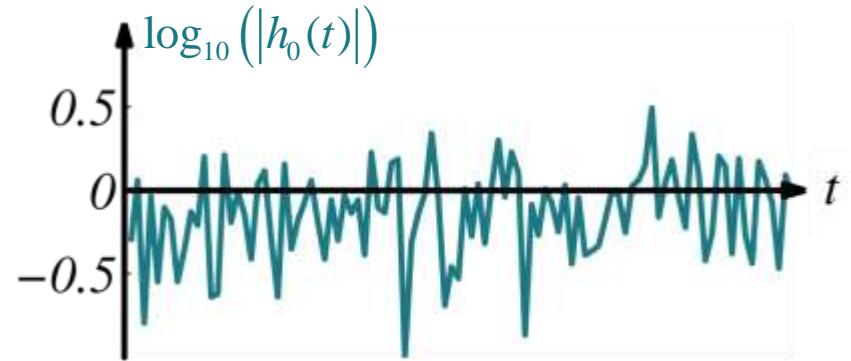


# Rice Fading Channel Model (9)

Probability density function (PDF)

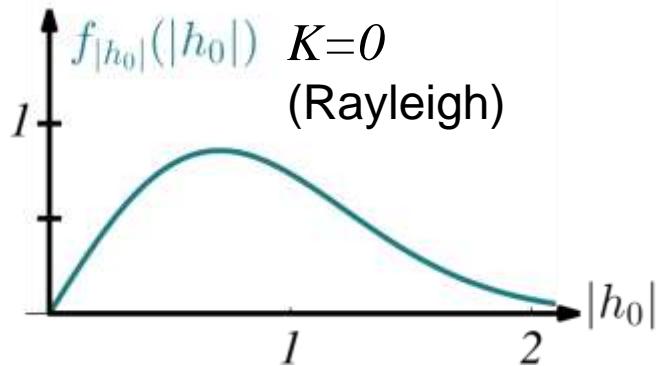


Time domain signal

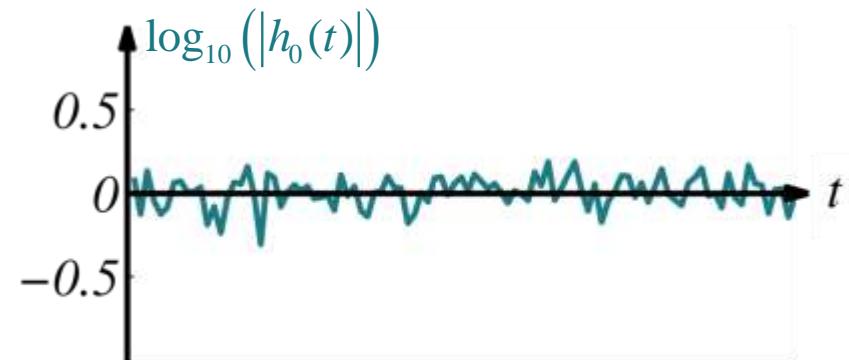
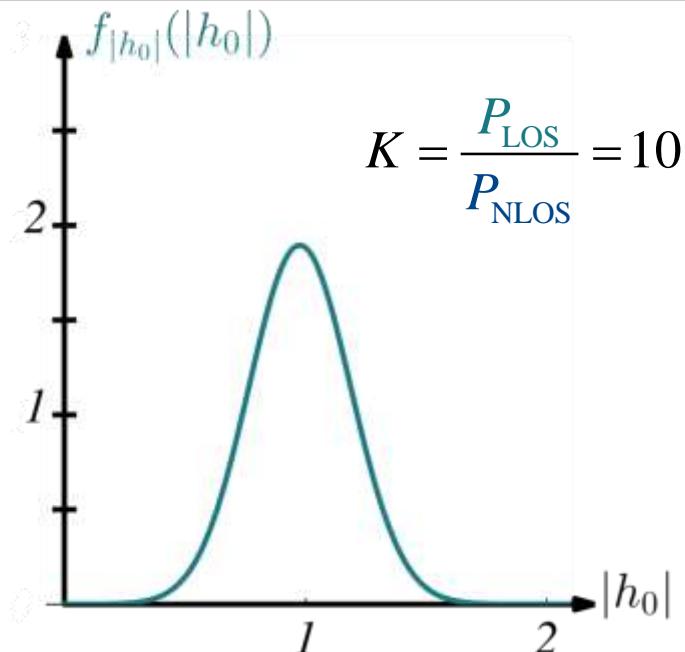
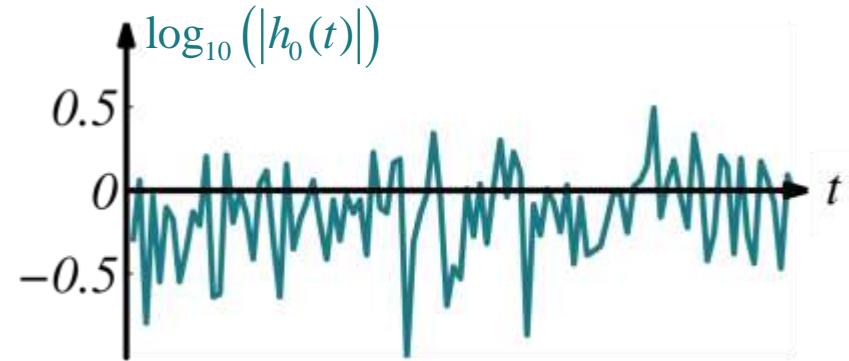


# Rice Fading Channel Model (10)

Probability density function (PDF)

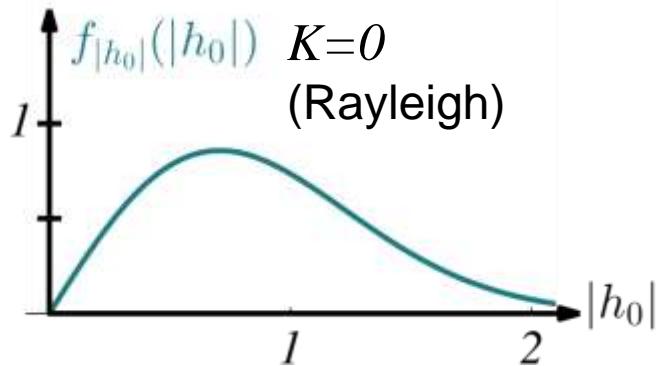


Time domain signal

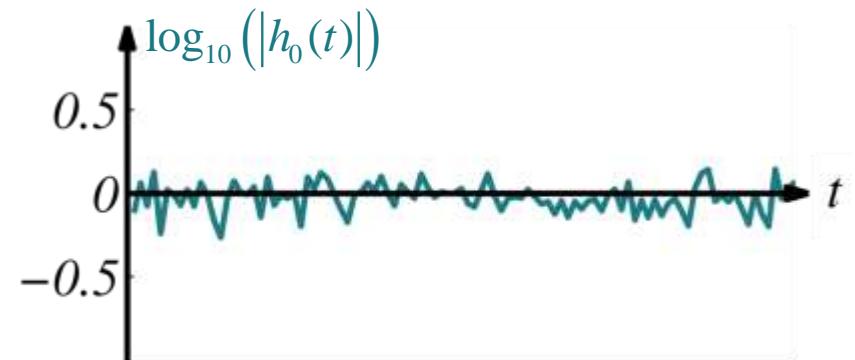
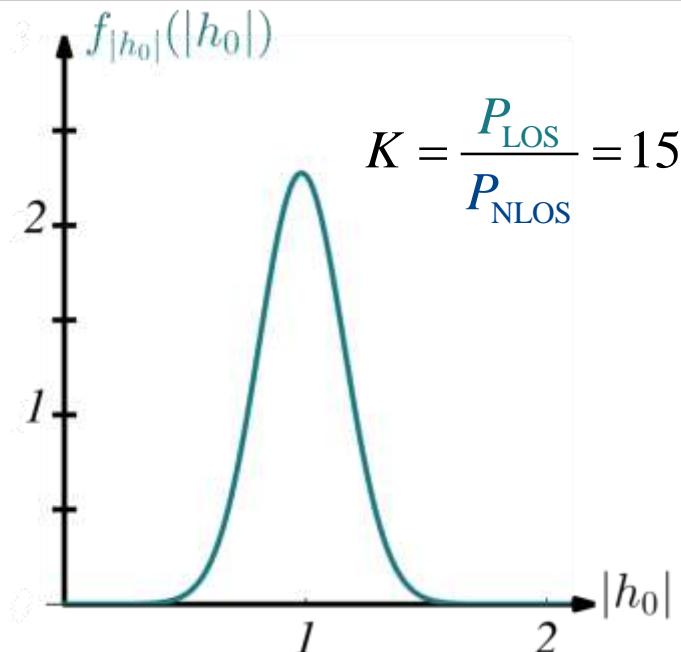
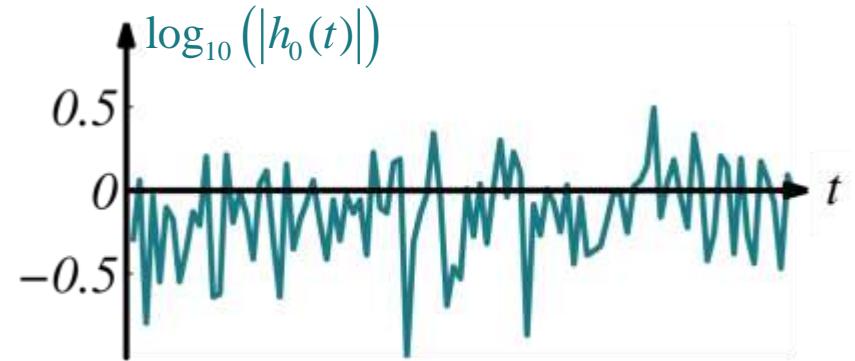


# Rice Fading Channel Model (11)

Probability density function (PDF)

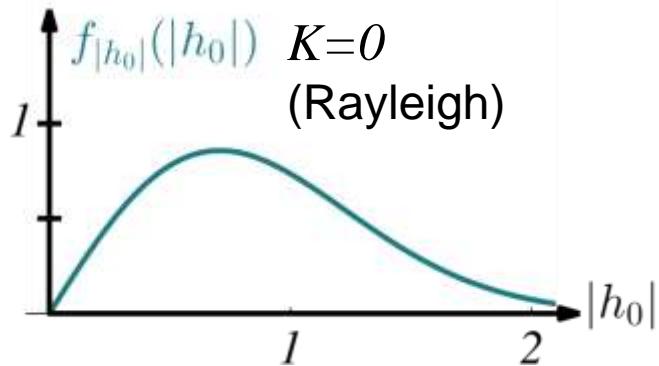


Time domain signal

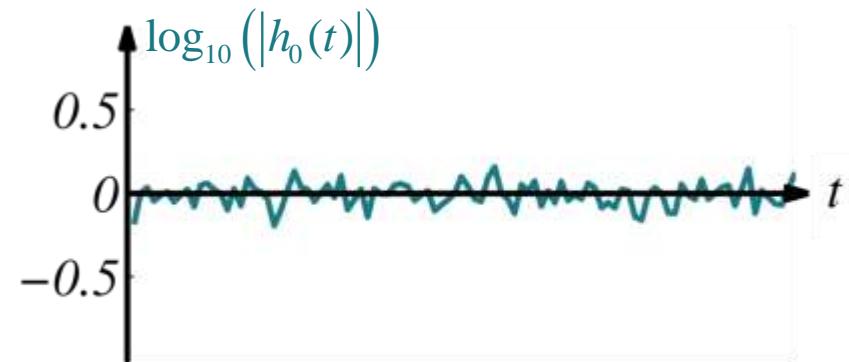
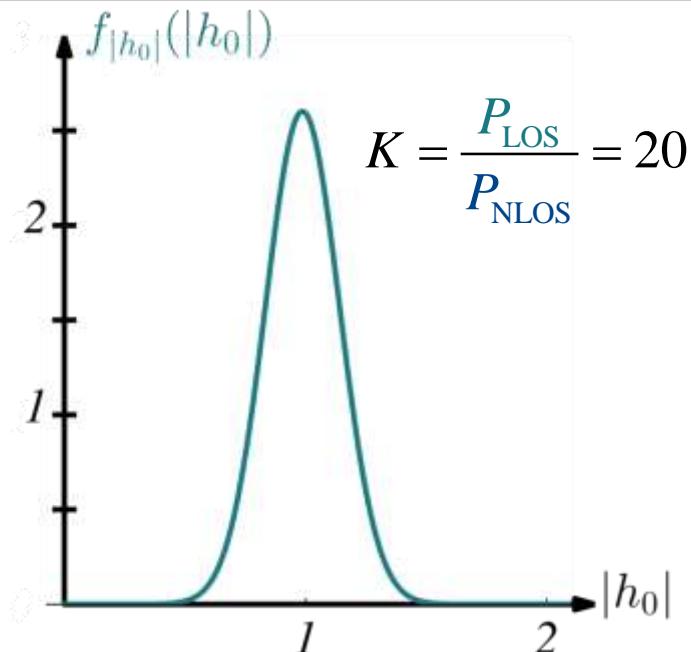
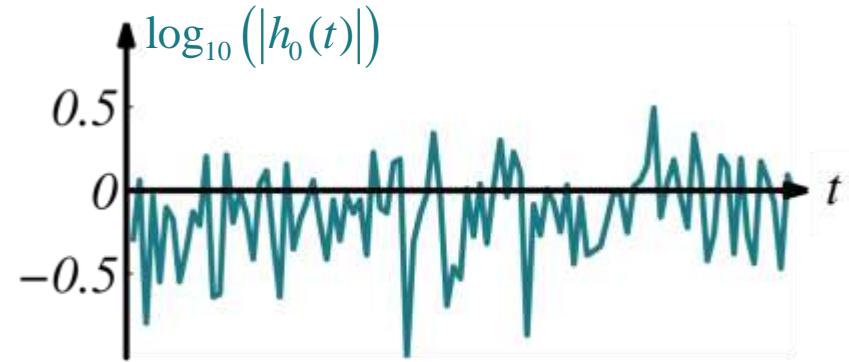


# Rice Fading Channel Model (12)

Probability density function (PDF)

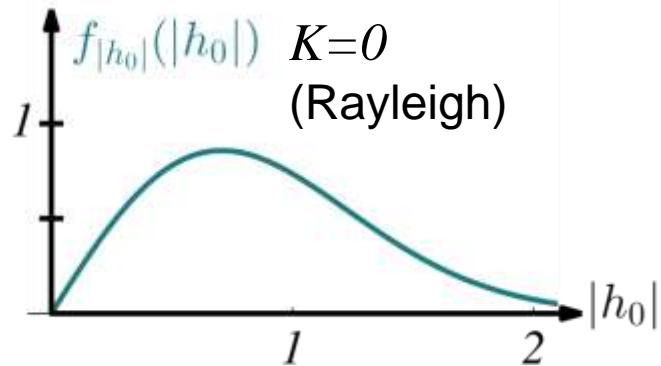


Time domain signal

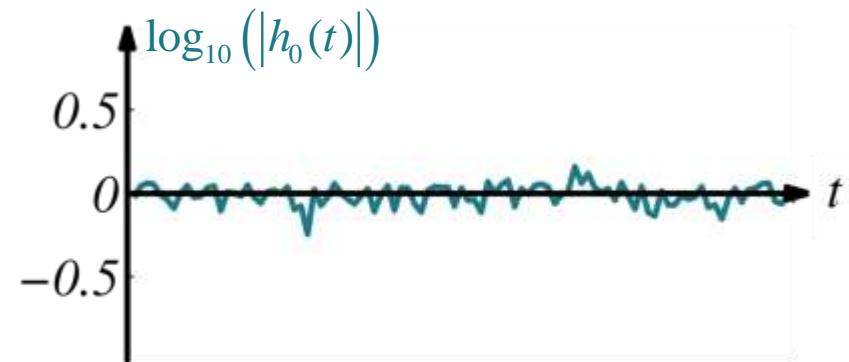
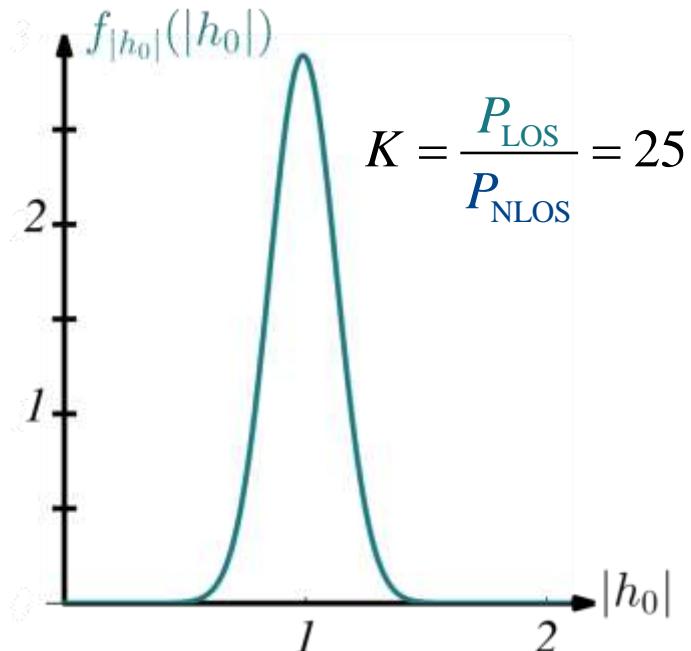
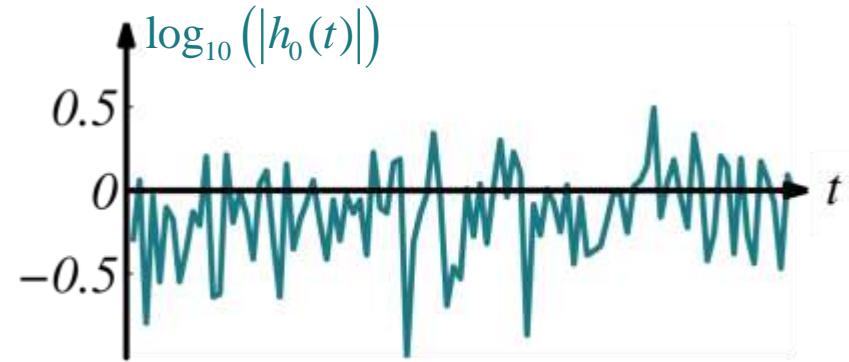


# Rice Fading Channel Model (13)

Probability density function (PDF)

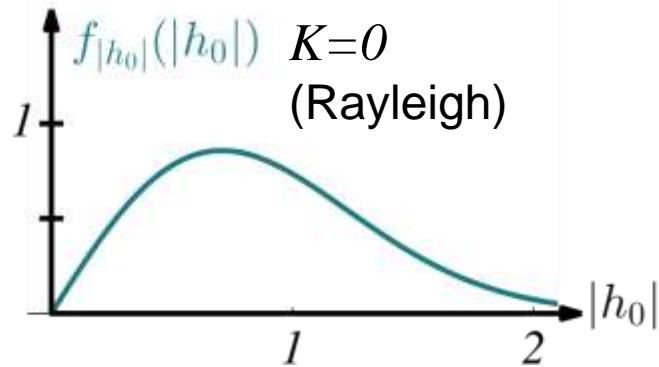


Time domain signal

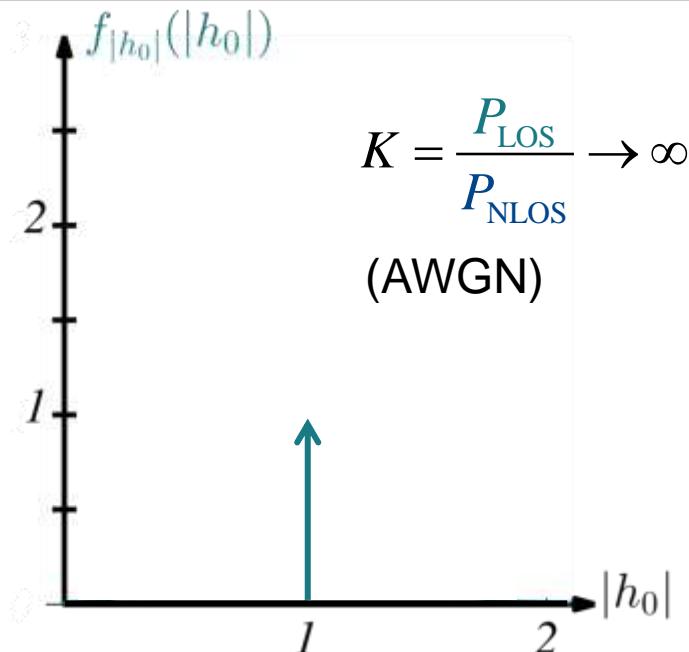
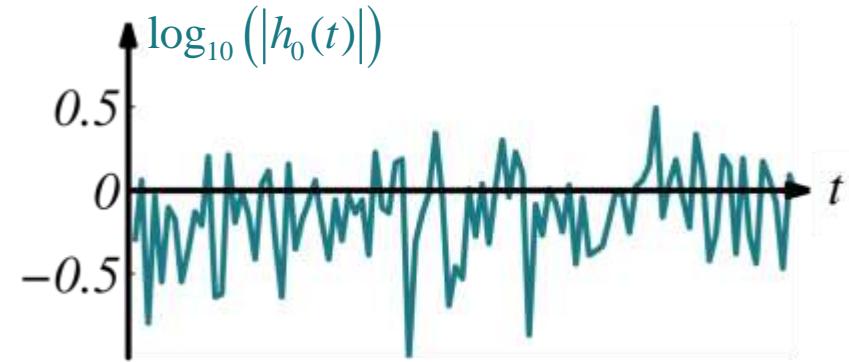


# Rice Fading Channel Model (14)

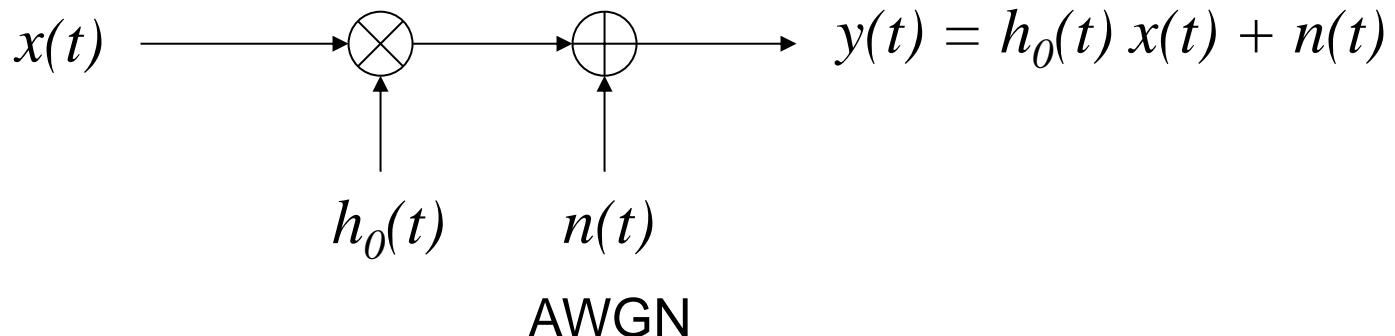
Probability density function (PDF)



Time domain signal



# AWGN, Rayleigh, Rice Channel Models



**AWGN**

$$h_0(t) = I \quad K \rightarrow \infty$$

**Rice**

$$h_0(t) = \sqrt{\frac{K}{1+K}} h_{\text{LOS}}(t) + \sqrt{\frac{1}{1+K}} h_{\text{NLOS}}(t)$$

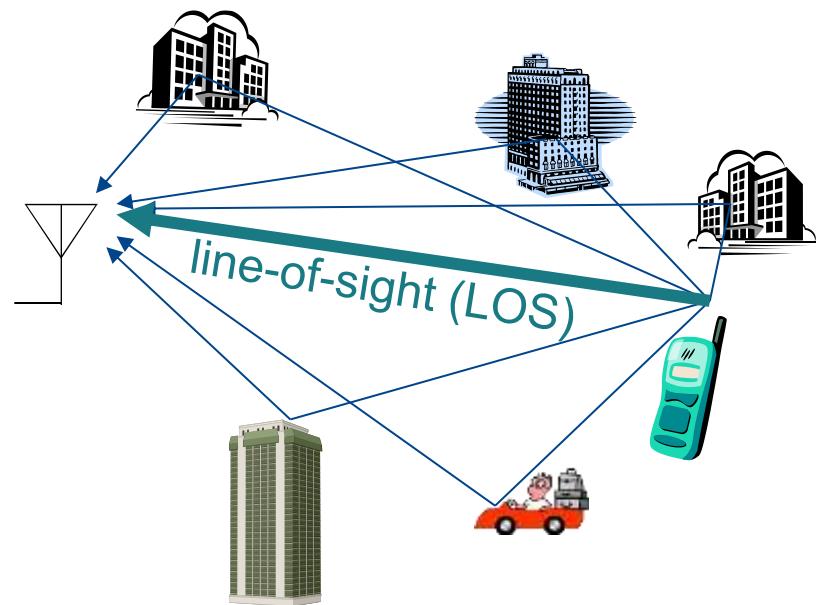
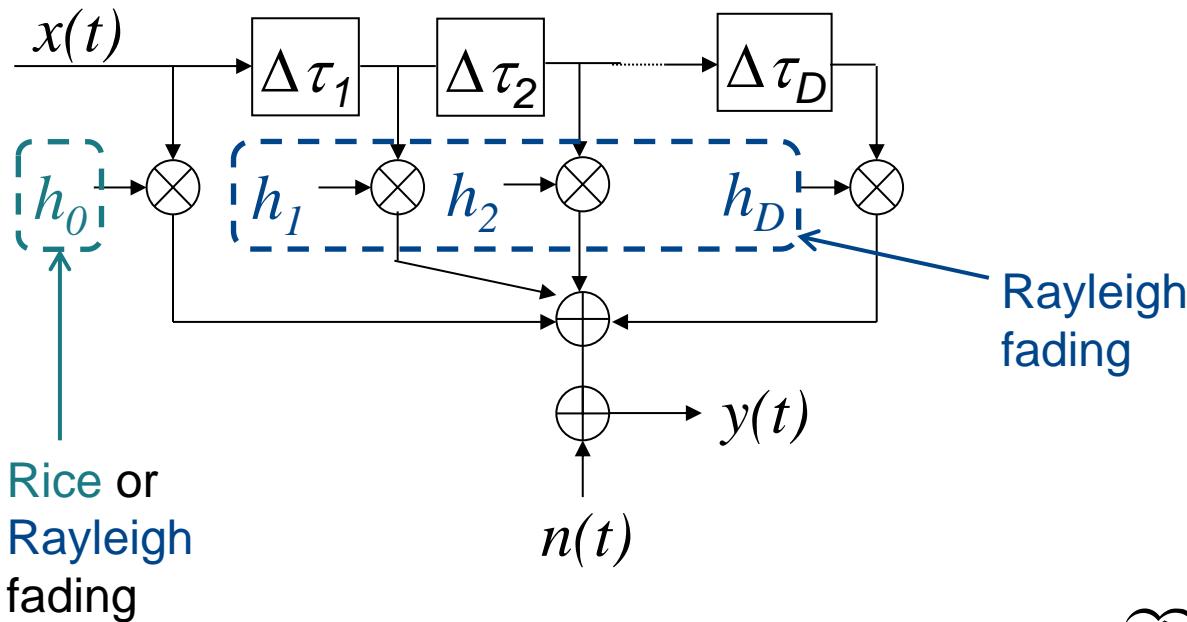
$$K=0$$

**Rayleigh**

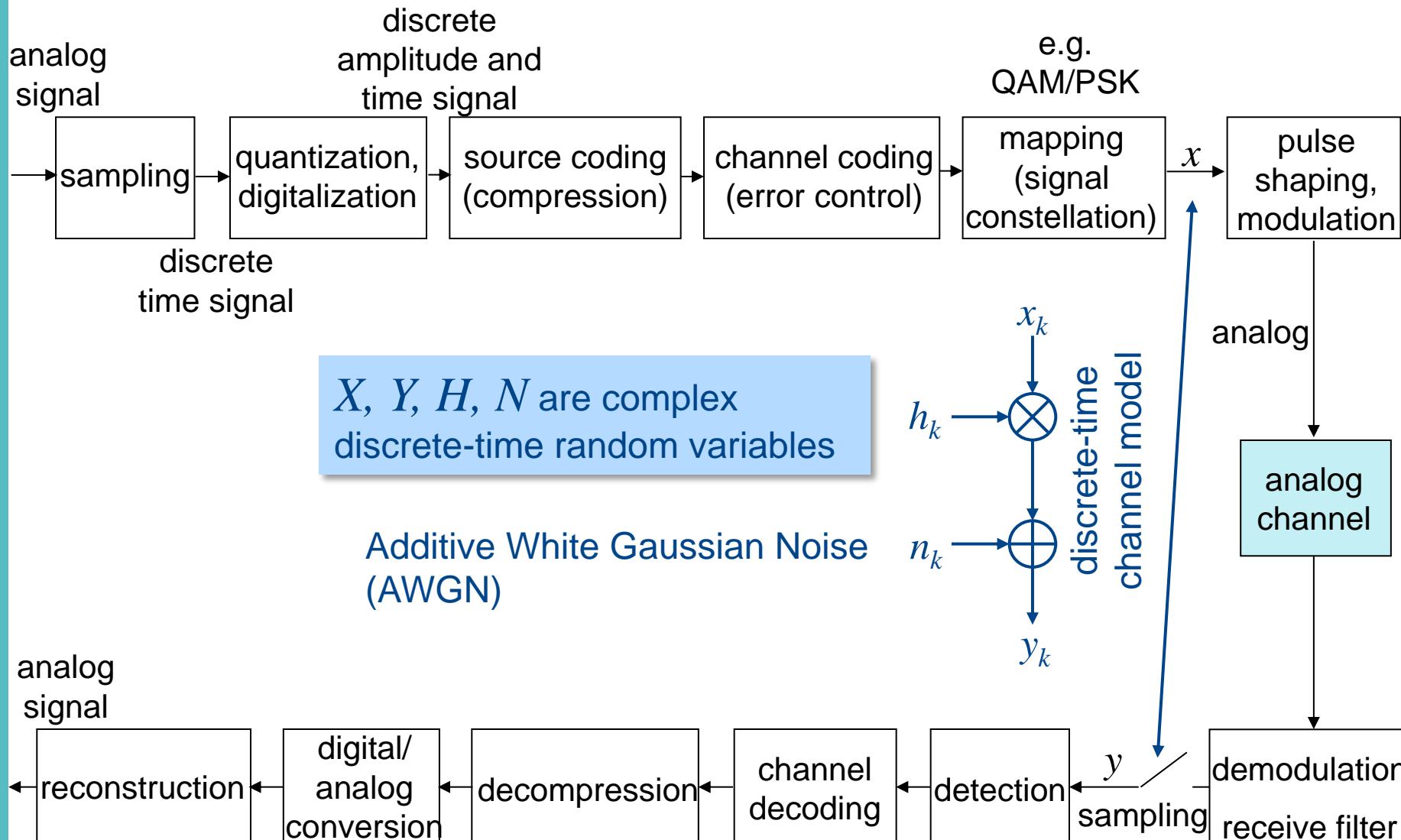
$$h_0(t) = h_{\text{NLOS}}(t)$$

$$K = \frac{P_{\text{LOS}}}{P_{\text{NLOS}}}$$

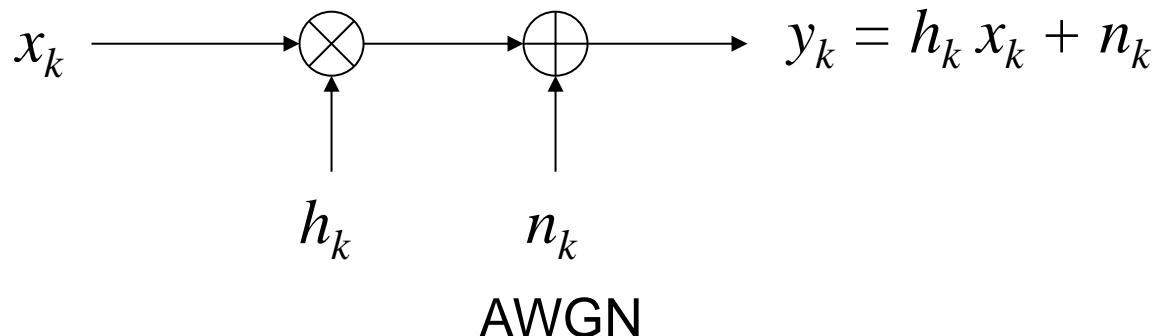
# Frequency-Selective Channel Model



# Digital Communications System

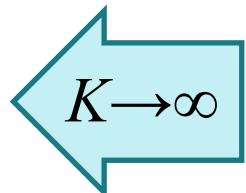


# Discrete-Time Channel Models



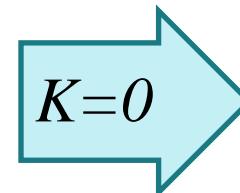
**AWGN**

$$h_k = 1$$



**Rice**

$$h_k = \sqrt{\frac{K}{1+K}} h_{LOS,k} + \sqrt{\frac{1}{1+K}} h_{NLOS,k}$$

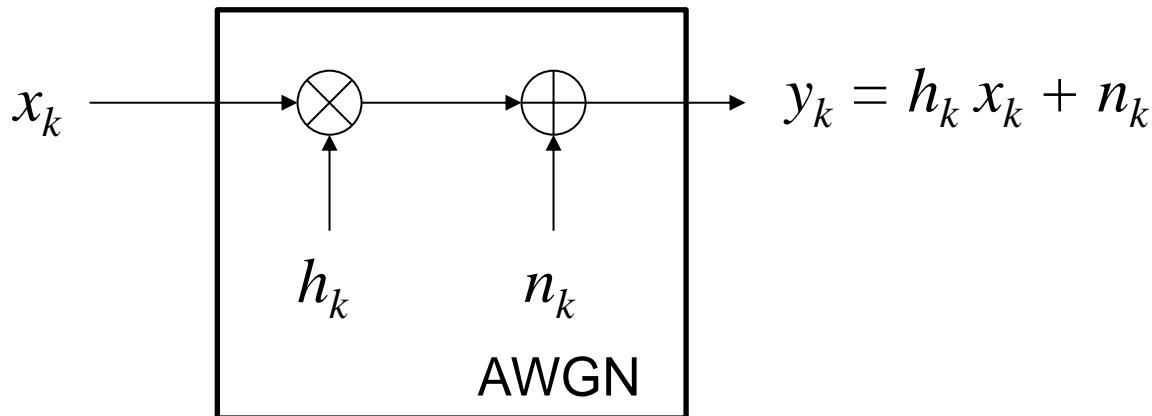


**Rayleigh**

$$h_k = h_{NLOS,k}$$

$$K = \frac{P_{LOS}}{P_{NLOS}}$$

# Discrete-Time Channel Models



The channel model is characterized by its transition probability density function  $f_{Y|X}(\mathbf{y}|\mathbf{x})$ .

For AWGN ( $h_k = 1$ ):

$$f_{Y|X}(y_k | x_k) = \frac{1}{\sqrt{2\pi}\sigma_N} e^{-\frac{1}{2\sigma_N^2}|y_k - x_k|^2}$$

# Discrete Memoryless Channels (DMC)

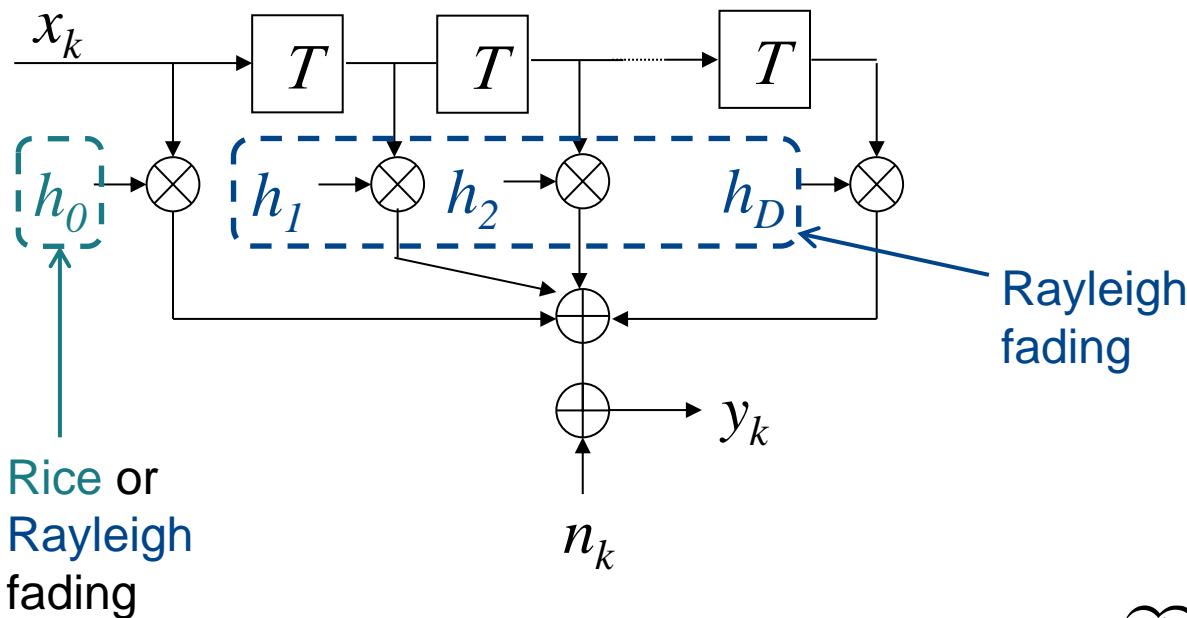


Memoryless means, that the transition probability density functions of symbol sequences are given by the product of the transition probability density functions of the individual symbols:

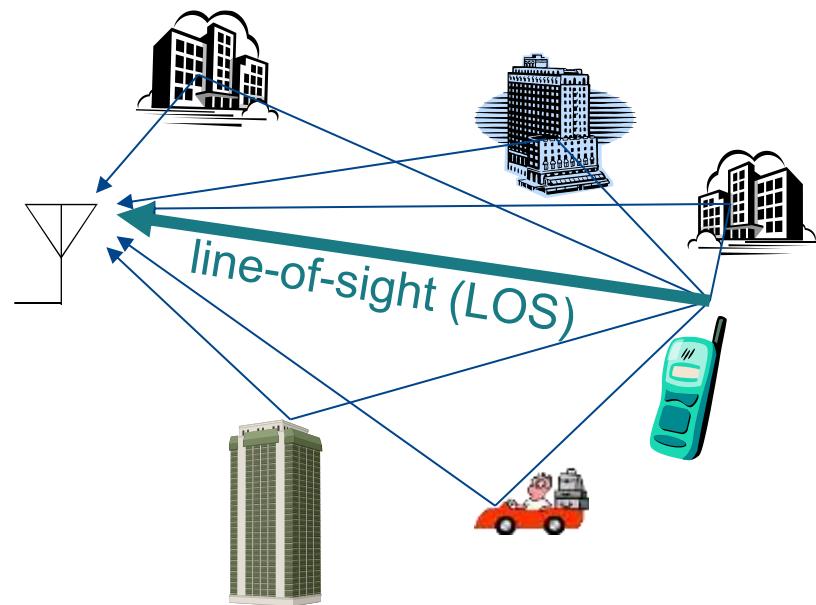
$$f_{\mathbf{Y}|\mathbf{X}}(y_0, \dots, y_{N-1} | x_0, \dots, x_{N-1}) = \prod_{k=0}^{N-1} f_{Y|X}(y_k | x_k)$$

Example: Frequency-flat additive white Gaussian noise (AWGN) channel

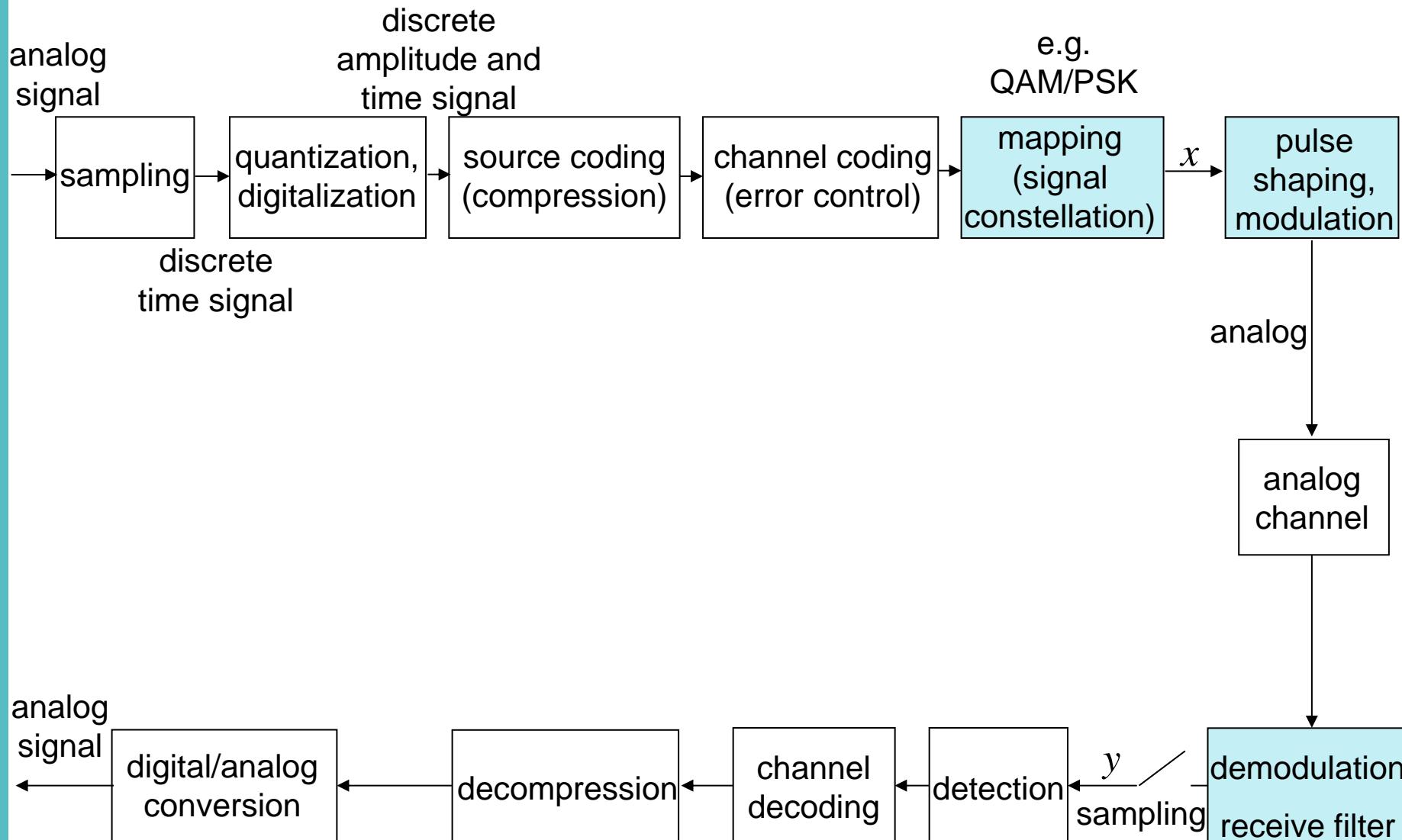
# Discrete-Time Frequency-Selective Channel Model



$$\text{Normalization: } E \left\{ \sum_{k=0}^D |h_k|^2 \right\} = 1$$



# Digital Communications System



# Bandpass Transmission Via Carrier Modulation

For many applications, baseband transmission is not suitable but the spectrum of the transmit signal has to be shifted to a higher frequency range.

This might be necessary in order to adapt the transmit signal to the pass-band of the transmission medium. E.g. radio transmission is not possible in the baseband but requires higher frequencies.

Another reason for bandpass transmission is that different frequencies are allocated for certain services. E.g. in Germany, the transmission range around 900 MHz is allocated to second generation mobile communications services (GSM) whereas the transmission range around 2 GHz is allocated to third generation mobile communications services (UMTS).

Furthermore, adapting the used frequency range allows to support multiple users at the same time at different frequencies (Frequency Division Multiple Access, FDMA). Bandpass transmission is realized by carrier modulation. I.e. the information carrying baseband signal modulates a parameter (amplitude, frequency, phase) of a sine wave. Depending on the transmission medium, we have an electromagnetic, acoustical or optical wave. The modulated parameter of the sine wave determines the name of the modulation scheme: Amplitude modulation, frequency modulation or phase modulation.

In digital communications, the modulating baseband signal is generated from a digital signal as described in the previous sections of the course and we call it a digital modulation method.

# Carrier Frequencies for Typical Communications Systems

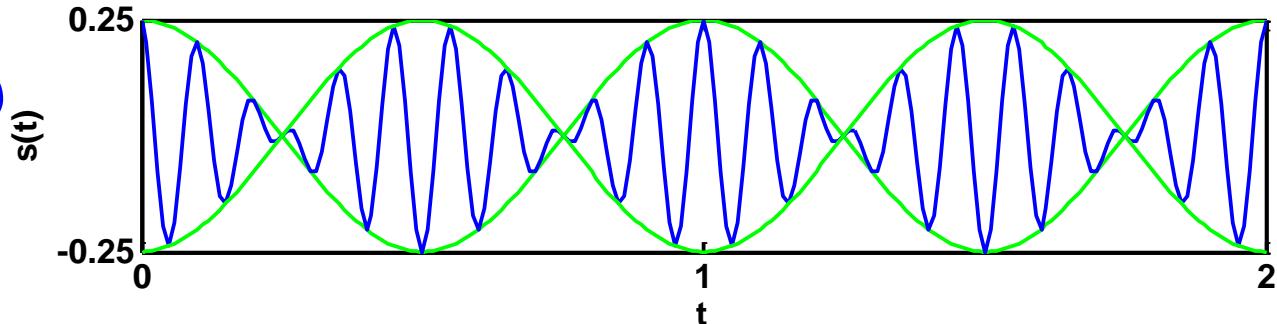
System	Typical carrier frequency $f_c$
data modem	1.8 kHz
medium wave radio	525-1605 kHz
FM radio	87-108 MHz
cellular radio	900 MHz, 1800 MHz, 1900MHz
directional radio	4 GHz, 6 GHz, 11 GHz
satellite radio	4/6 GHz, 11/14 GHz, 20/30 GHz
optical fiber	$\lambda=1.5 \mu m \Rightarrow f_c=200 \text{ THz}=2 \cdot 10^{14} \text{ Hz}$

# Carrier Modulation

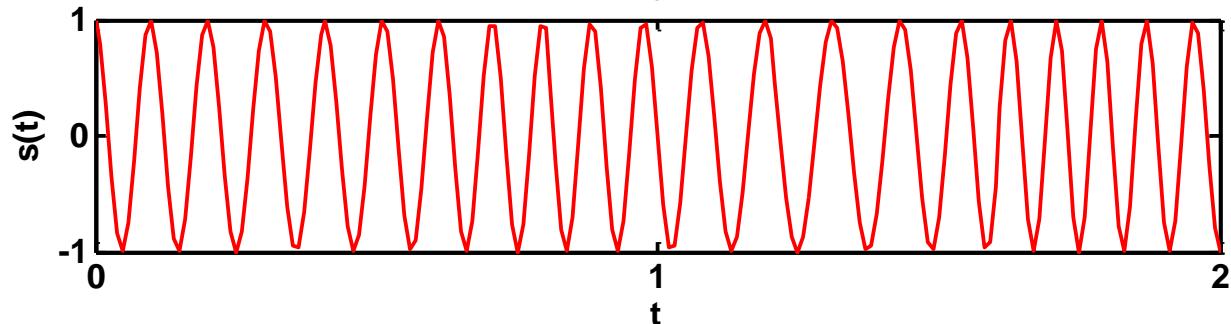
Transmit signal:  $s(t) = a(t) \cos\{2\pi[f_c + f(t)]t + \varphi(t)\}$   $f_c$ : carrier frequency

Example: Modulating info signal:  $x(t) = \frac{1}{4} \cos(2\pi f_n t)$

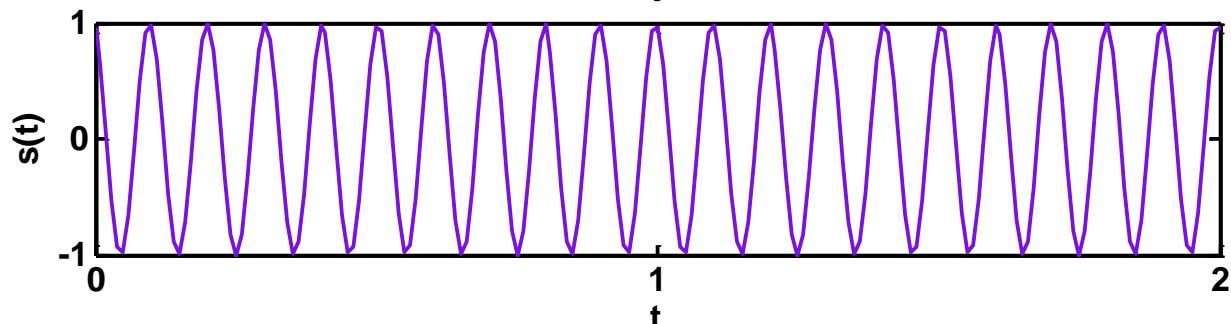
Amplitude Modulation:  
 $a(t)$  is a function of  $x(t)$



Frequency Modulation:  
 $f(t)$  is a function of  $x(t)$



Phase Modulation:  
 $\varphi(t)$  is a function of  $x(t)$



# Linear Digital Modulation Methods (1)

Linear digital modulation methods generate the transmit signal in the following steps: A group of  $b$  bits is mapped to a complex signal constellation point  $d_n$  according to the mapping rule of the modulation scheme. The  $\log_2 M = 2^b$  possible discrete values (signal space constellation points) of the symbols  $d_n$  constitute the signal space constellation which determines class and name of the modulation method. The number  $M$  of signal constellation points is called the *cardinality* of the modulation scheme. It is part of the name of a digital modulation method (e.g.  $M$ -PSK,  $M$ -QAM,  $M$ -ASK).

In phase modulation ( $M$ -ary phase shift keying,  $M$ -PSK), the information is contained in the phase of the symbols  $d_n$ , i.e.  $|d_n| = \text{const}$  for  $n=0, 1, \dots, M-1$ . Consequently, all symbols of a PSK constellation are located on a circle in the complex plane.

The mapping of bits to constellation points is often done according to a Gray mapping, i.e. adjacent constellation points (constellation points with minimum Euclidean distance) differ in only one bit. Since the most likely symbol error at a reasonably high SNR is to confuse two adjacent symbols, we have only one bit error for the most likely symbol error. Symbol errors which result in more than one bit error have lower probability. Hence, Gray mapping minimizes the average bit error probability.

In amplitude shift keying ( $M$ -ASK), all constellation points are located on a straight line, usually on the real axis. The symbols can have different magnitude  $|d_n|$ .

Quadrature amplitude modulation ( $M$ -QAM) is a more general modulation method where the symbols  $d_n$  can have different amplitude and different phase. Since QAM can be viewed as a combination of PSK and ASK, QAM schemes are sometimes called hybrid modulation methods.

# Linear Digital Modulation Methods (2)

Often, the constellation points are normalized such that  $E\{|d_n|^2\}=1$ .

The real part  $d_{n,I}$  of  $d_n$  is called the *inphase* component, the imaginary part  $d_{n,Q}$  of  $d_n$  is called the *quadrature* component. Both components are referred to as the *quadrature components*.

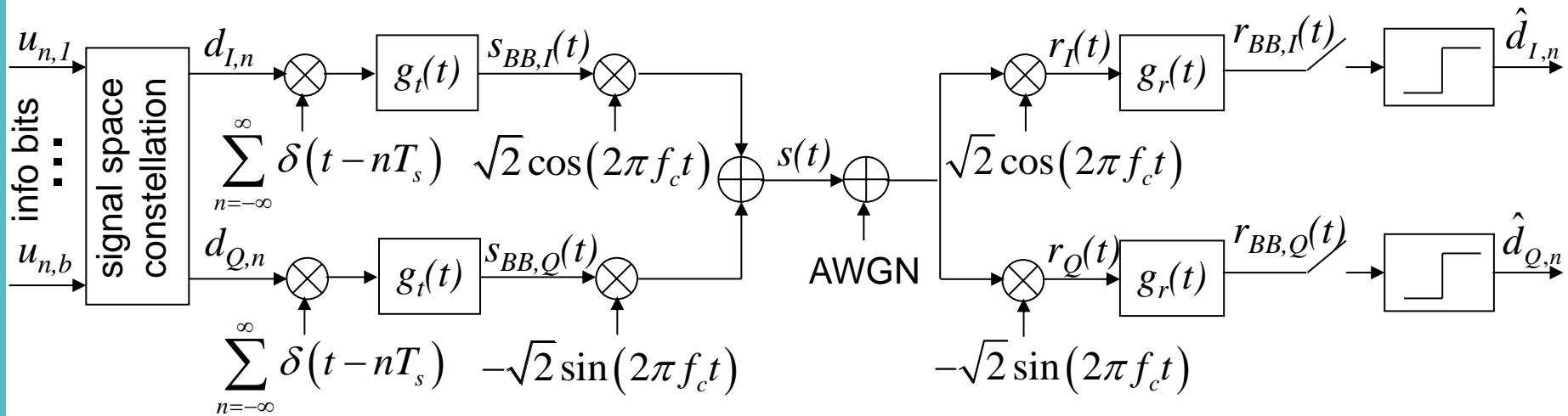
Both inphase and quadrature component are passed through a pulse shaping filter similar as in baseband transmission. The outputs of the pulse shaping filters form the complex equivalent baseband signal  $s_{BB}(t)$  which is also called the (scaled) complex envelope of the transmit signal. The scaled magnitude  $\sqrt{2}|s_{BB}(t)|$  of the equivalent baseband signal is the real envelope of the transmit signal.

The radio frequency (RF) transmit signal  $s(t)$  is obtained by multiplying the inphase component with a cosine carrier, the quadrature component with a sine carrier and adding both quadrature components. The RF transmit signal is always real.

The transmit signal can be completely described by the complex equivalent baseband signal. Therefore, theoretical analysis can be done based on the equivalent baseband model which significantly simplifies the mathematical treatment.

At the receiver, inphase component and quadrature component are downconverted by multiplication with a carrier cosine and sine, respectively. Since cosine and sine are orthogonal functions, the quadrature components can be completely separated at the receiver. In each of the quadrature components, detection is performed using a matched filter as it is done in baseband transmission yielding an estimate on the transmitted symbol  $d_n$ . Since the matched filter has lowpass characteristic, the spectral components around twice the carrier frequency  $f_c$ , which occur due to downconversion, are eliminated by the matched filter.

# Linear Digital Modulation Methods



Transmit signal:  $s(t) = s_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - s_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t)$

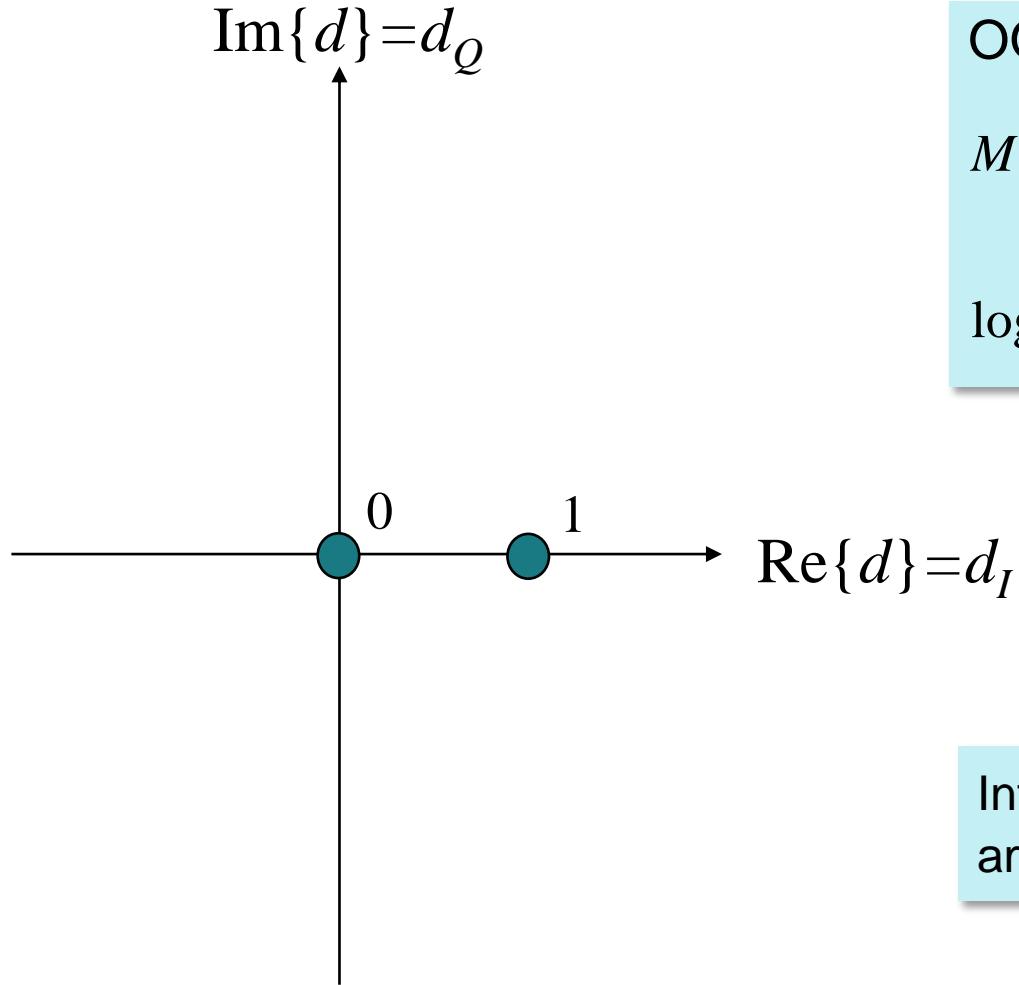
Inphase component:  $s_{BB,I}(t) = \sum_{n=-\infty}^{\infty} d_{I,n} g_t(t - nT_s)$

$$\begin{aligned}
 r_I(t) &= [s_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - s_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t)] \cdot \sqrt{2} \cdot \cos(2\pi f_c t) \\
 &= s_{BB,I}(t)[1 + \cos(4\pi f_c t)] - s_{BB,Q}(t)\sin(4\pi f_c t)
 \end{aligned}$$

Quadrature component:  $s_{BB,Q}(t) = \sum_{n=-\infty}^{\infty} d_{Q,n} g_t(t - nT_s)$

$$\begin{aligned}
 r_Q(t) &= [s_{BB,I}(t)\sqrt{2}\cos(2\pi f_c t) - s_{BB,Q}(t)\sqrt{2}\sin(2\pi f_c t)] \cdot (-\sqrt{2}) \cdot \sin(2\pi f_c t) \\
 &= -s_{BB,I}(t)[\sin(4\pi f_c t)] + s_{BB,Q}(t)[1 - \cos(4\pi f_c t)]
 \end{aligned}$$

# Signal Space Constellation: On-Off Keying (OOK)



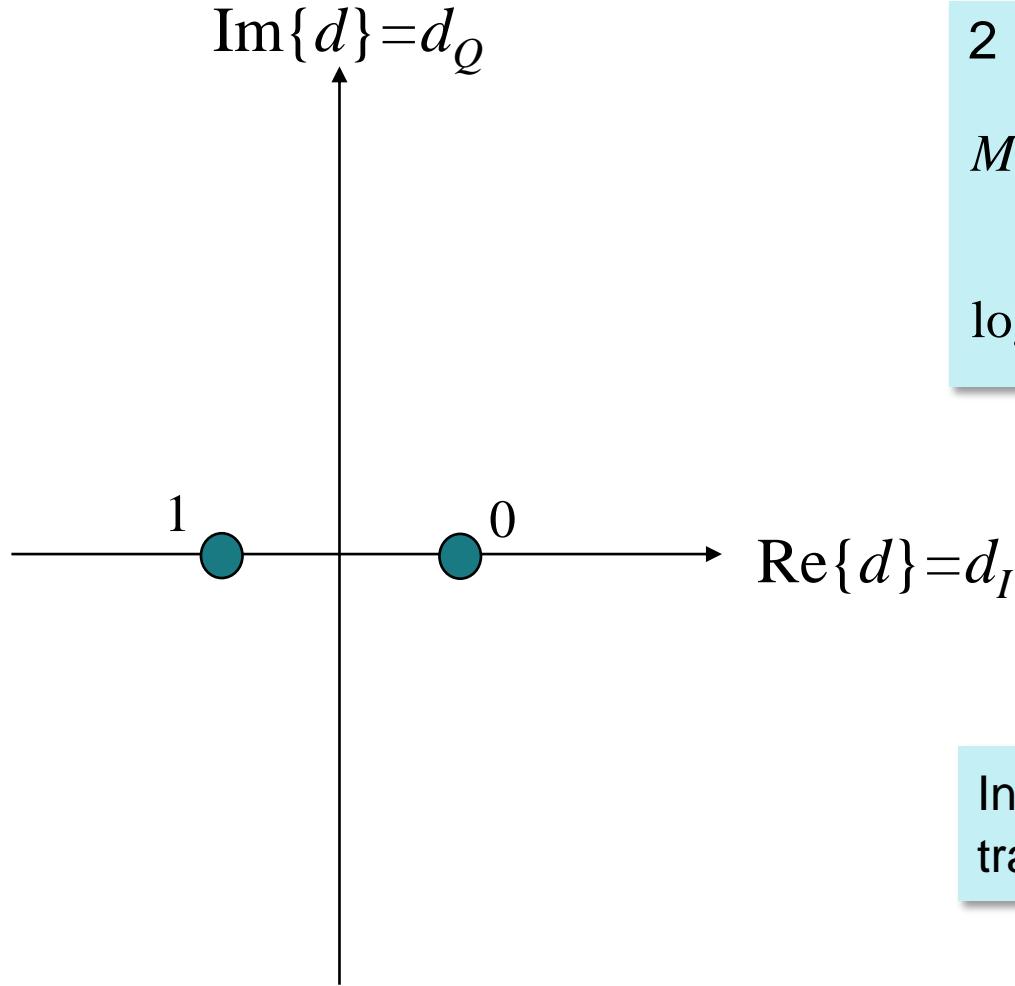
OOK

$M=2$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=1$  bit per transmit symbol  $d$

Information is contained in  
amplitude of transmit symbols  $d$ .

# Signal Space Constellation: Binary Phase Shift Keying (BPSK)



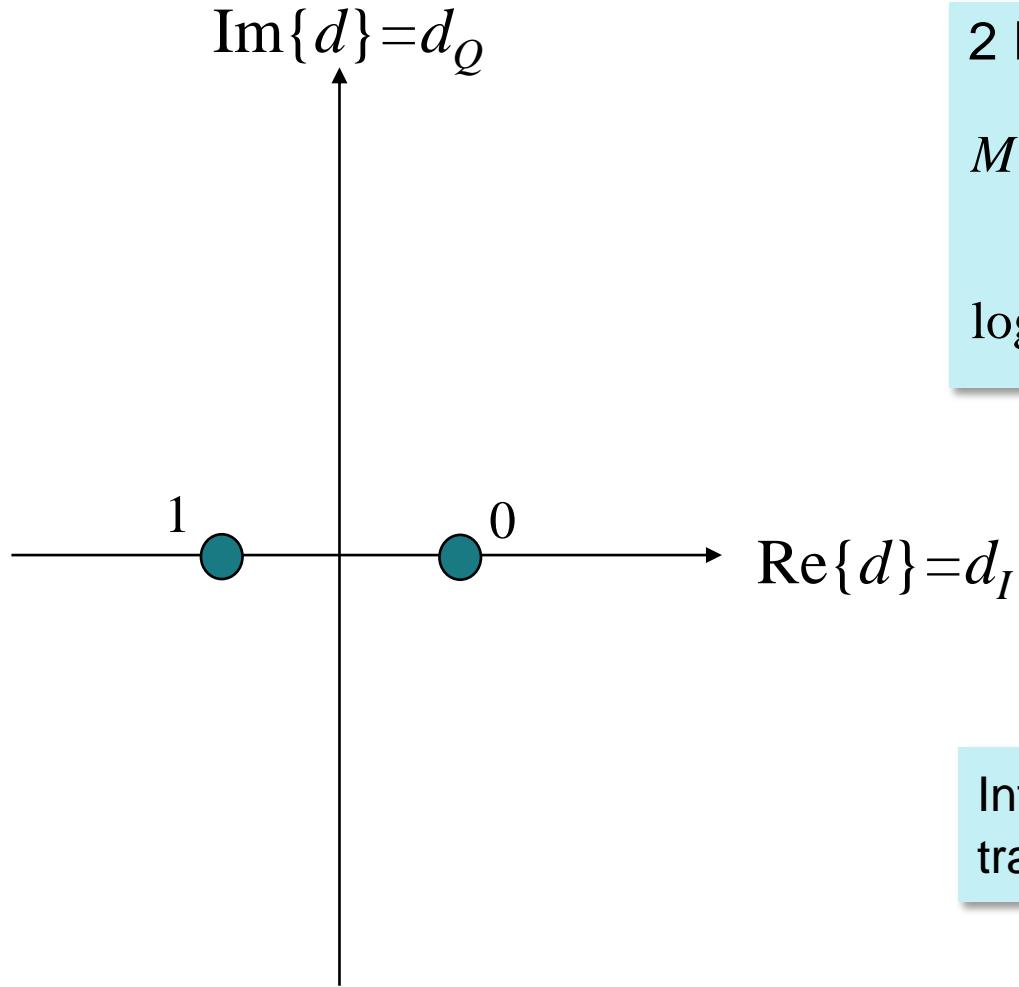
2 PSK / BPSK

$M=2$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=1$  bit per transmit symbol  $d$

Information is contained in phase of  
transmit symbols  $d$ .

# Signal Space Constellation: Amplitude Shift Keying (ASK) (1)



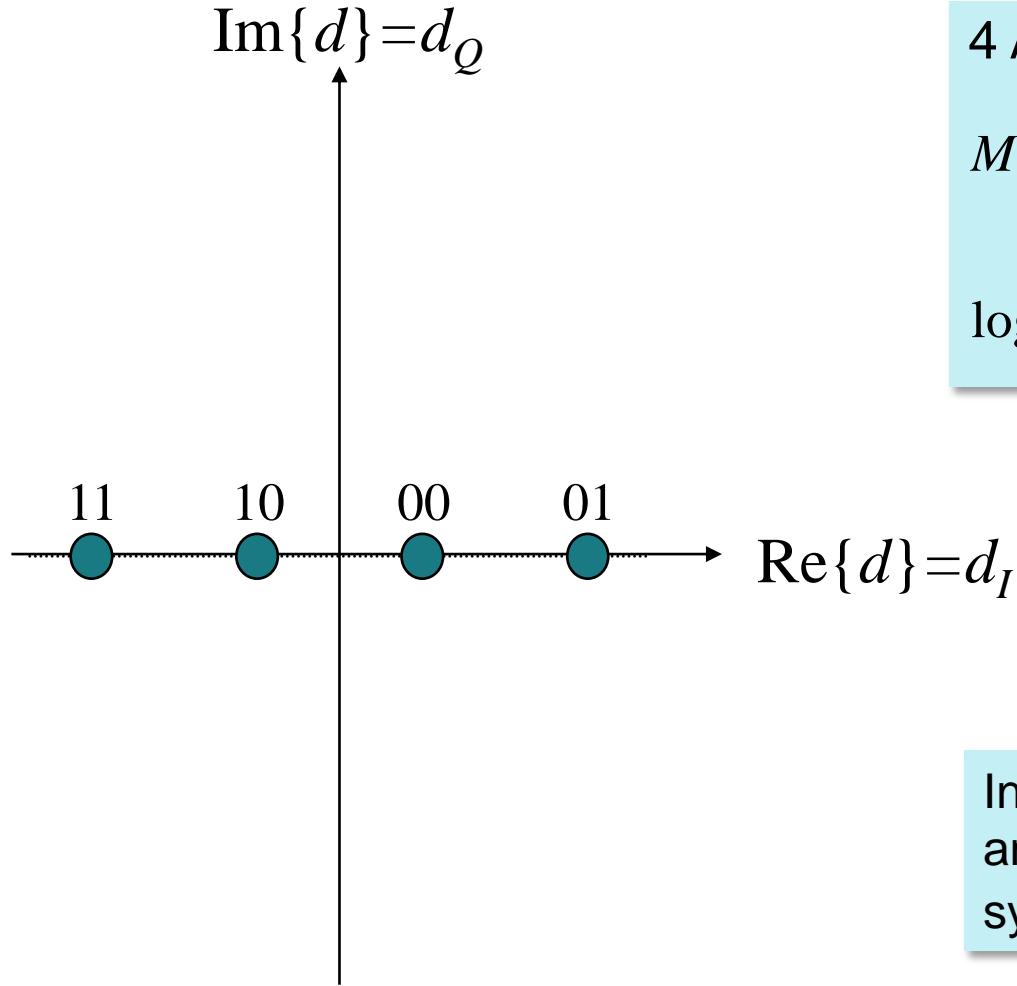
2 PSK / BPSK / 2 ASK

$M=2$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=1$  bit per transmit symbol  $d$

Information is contained in phase of  
transmit symbols  $d$ .

# Signal Space Constellation: Amplitude Shift Keying (ASK) (2)



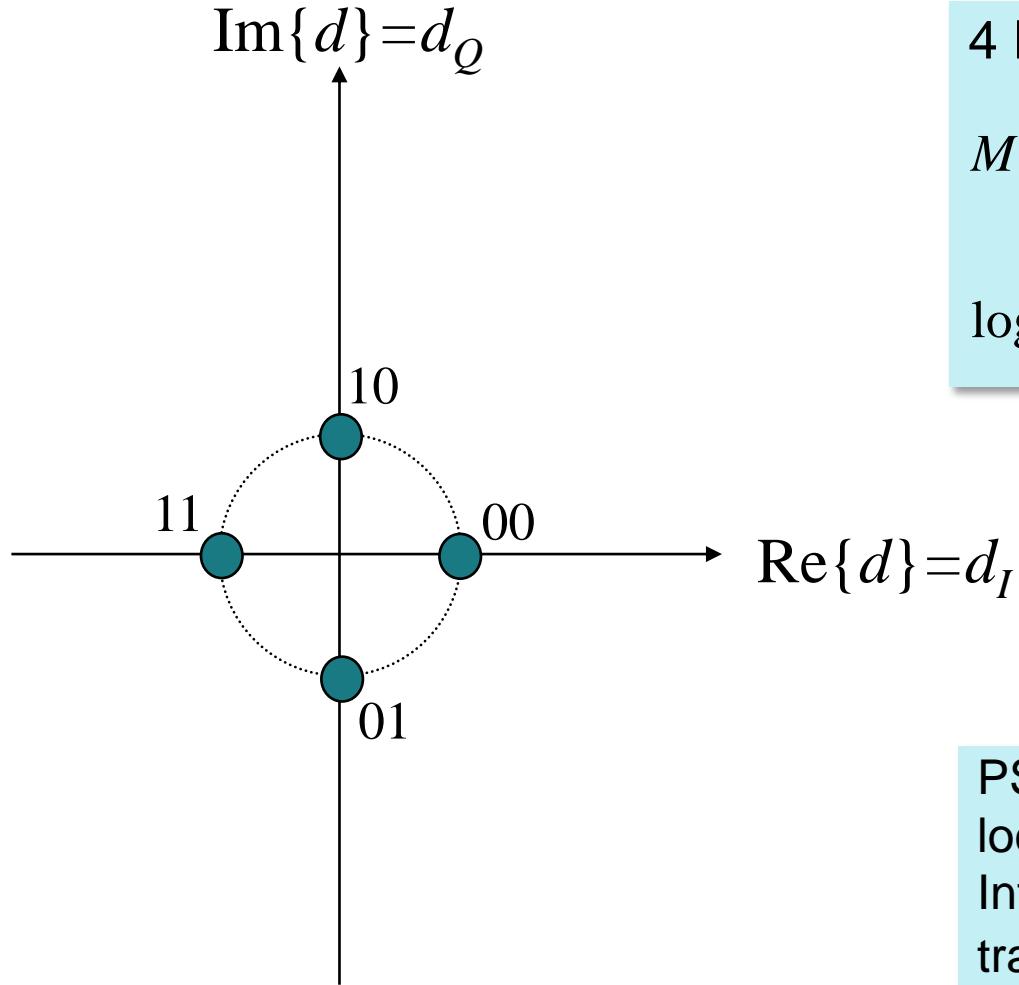
4 ASK

$M=4$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=2$  bit per transmit symbol  $d$

Information is contained in  
amplitude and phase of transmit  
symbols  $d$ .

# Signal Space Constellation: Quadrature Phase Shift Keying (QPSK) (1)



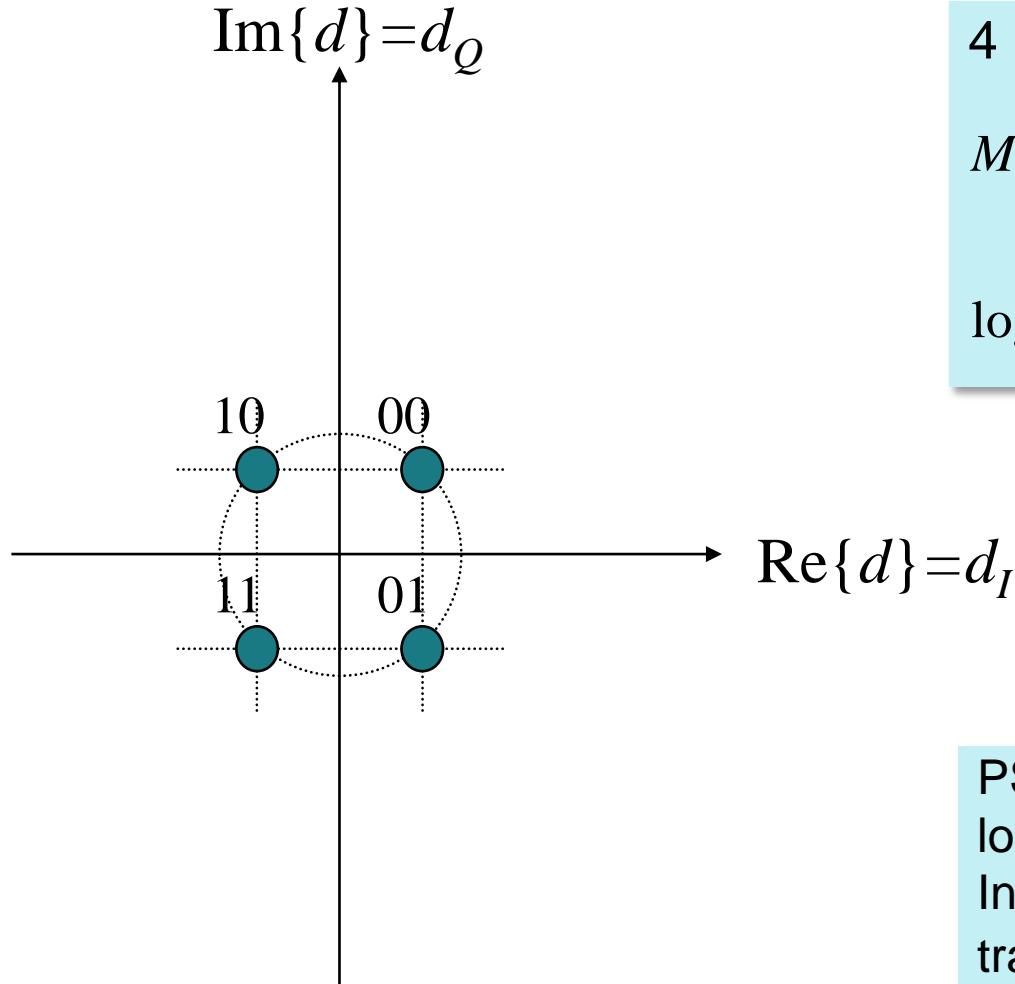
4 PSK / QPSK

$M=4$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=2$  bit per transmit symbol  $d$

PSK: Constellation points are located on a circle.  
Information is contained in phase of transmit symbols  $d$ .

# Signal Space Constellation: Quadrature Phase Shift Keying (QPSK) (2)



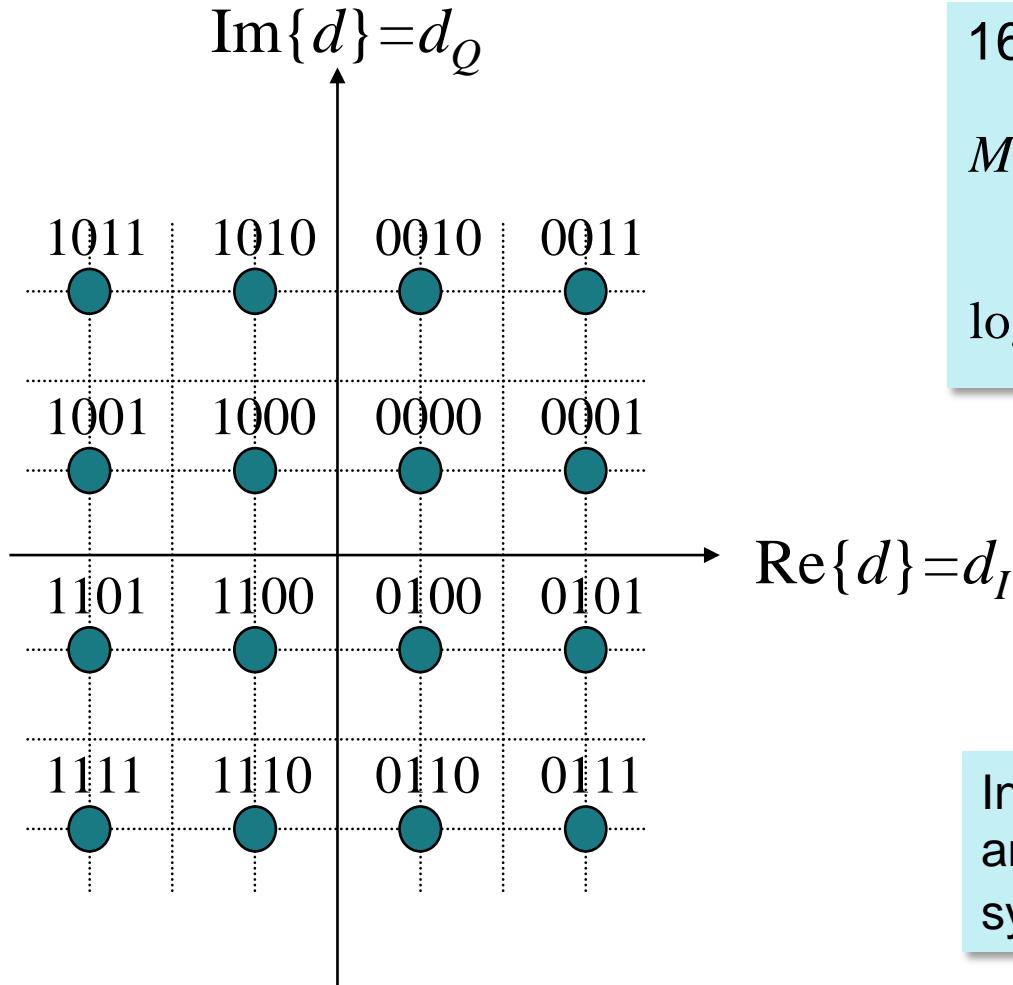
4 PSK / QPSK / 4 QAM

$M=4$  constellation points  
(possible transmit symbols  $d$ )

$\log_2 M=2$  bit per transmit symbol  $d$

PSK: Constellation points are located on a circle.  
Information is contained in phase of transmit symbols  $d$ .

# Signal Space Constellation: Quadrature Amplitude Modulation (QAM)



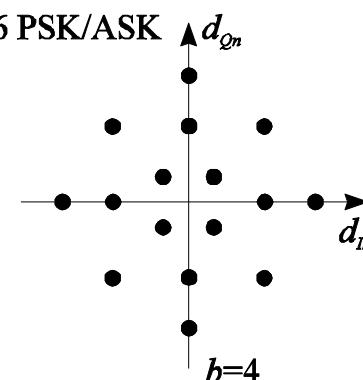
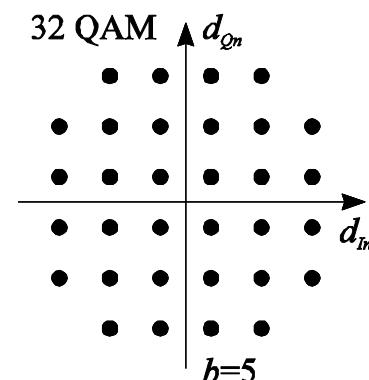
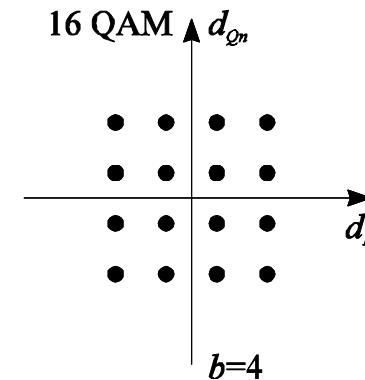
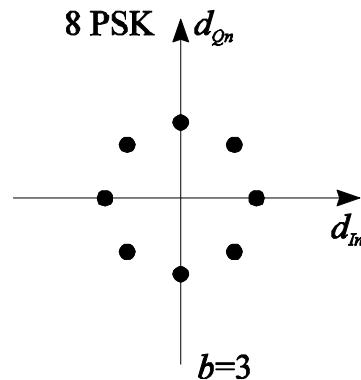
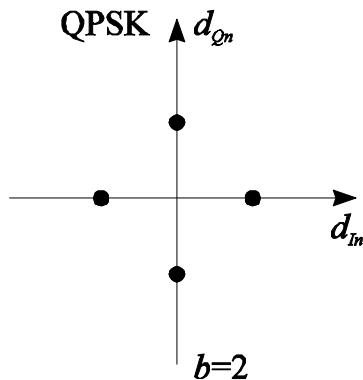
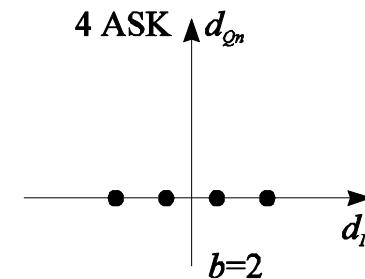
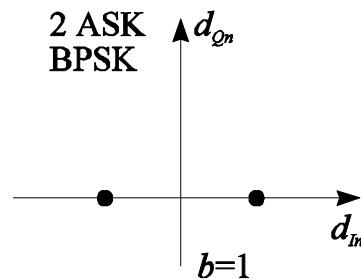
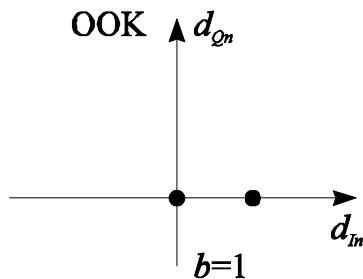
16 QAM

$M=16$  constellation points  
(possible transmit symbols  $d$ )

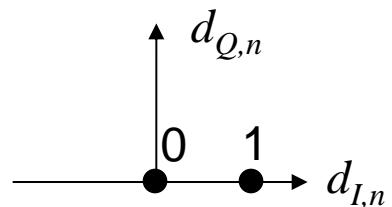
$\log_2 M=4$  bit per transmit symbol  $d$

Information is contained in  
amplitude and phase of transmit  
symbols  $d$ .

# Signal Space Constellations



# On-Off Keying (OOK) with Rectangular Pulses



$M = 2$  constellation points

$b = \log_2 M = 1$  bit per transmit symbol

Bit sequence  $u_n$ :

1

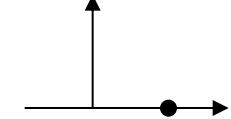
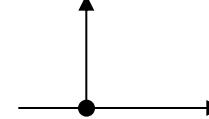
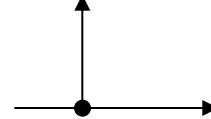
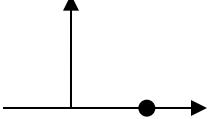
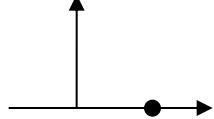
1

0

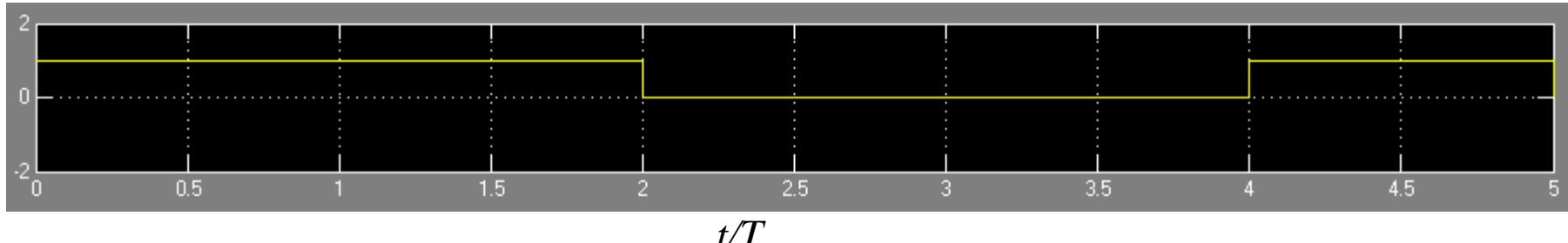
0

1

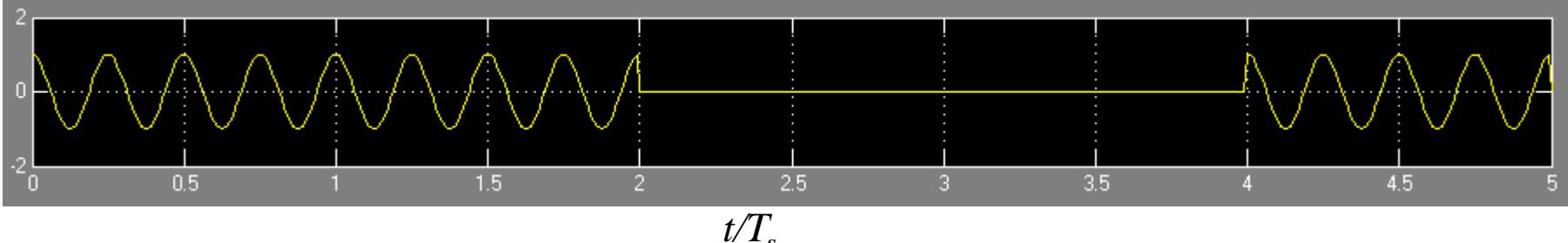
Symbol sequence  $d_n$ :



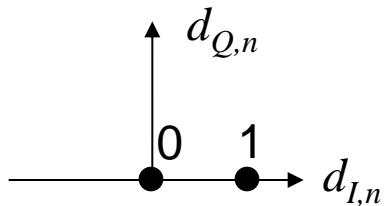
Equivalent baseband signal  $\sqrt{2}s_{BB}(t)$ :



Transmit signal  $s(t)$ :



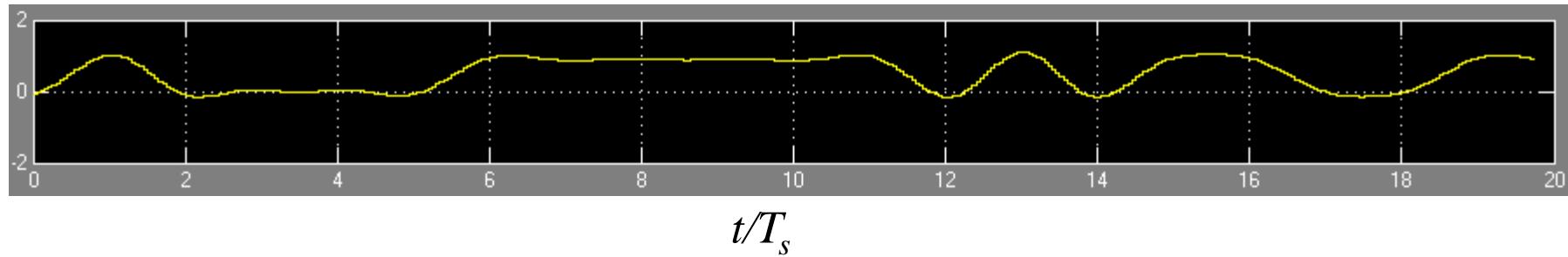
# On-Off Keying (OOK) with Square-Root Raised Cosine Pulses



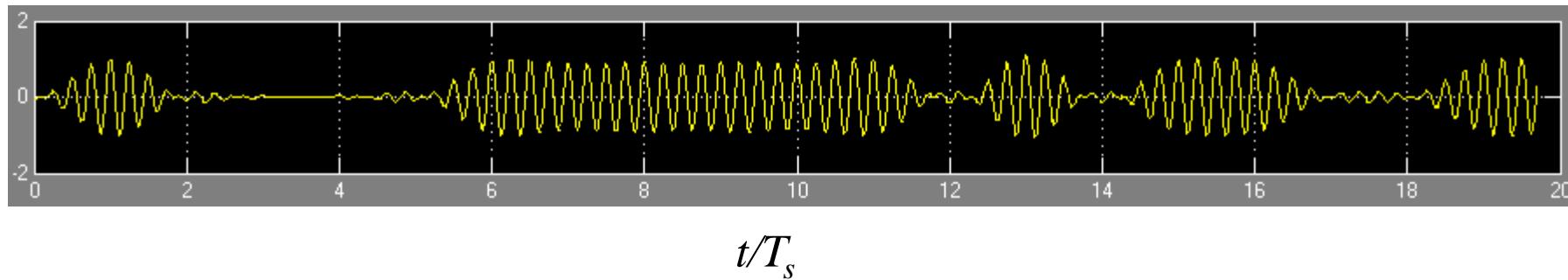
$M = 2$  constellation points

$b = \log_2 M = 1$  bit per transmit symbol

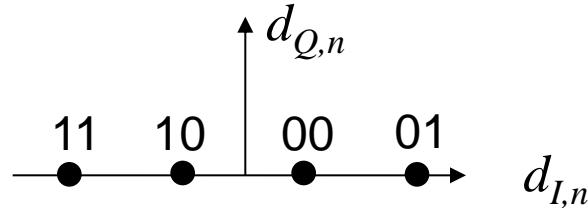
Equivalent baseband signal  $\sqrt{2}s_{BB}(t)$ :



Transmit signal  $s(t)$ :



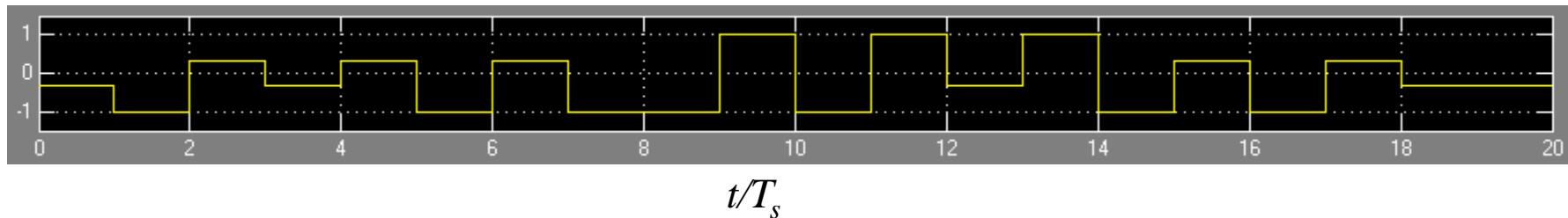
# 4-ary Amplitude Shift Keying (4-ASK) with NRZ Rectangular Pulses



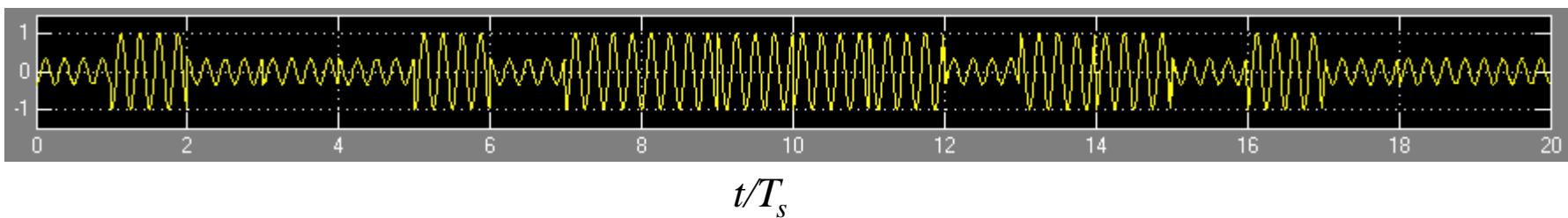
Bit sequence  $u_n$ :

10 11 00 10 00 11 00 11 11 01 11 01 10 01 11 00 11 00 10 10

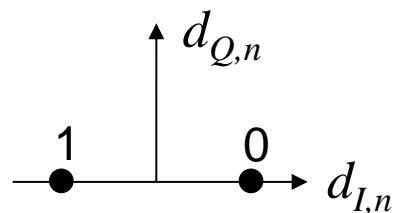
Equivalent baseband signal  $\sqrt{2}s_{BB}(t)$ :



Transmit signal  $s(t)$ :



# Binary Phase Shift Keying (BPSK) with Rectangular Pulses



$M = 2$  constellation points

$b = \log_2 M = 1$  bit per transmit symbol

Bit sequence  $u_n$ :

0

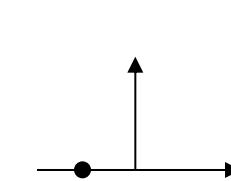
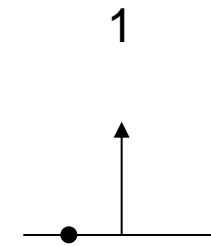
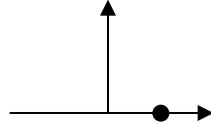
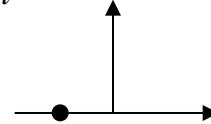
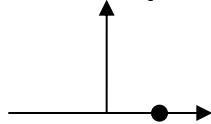
1

0

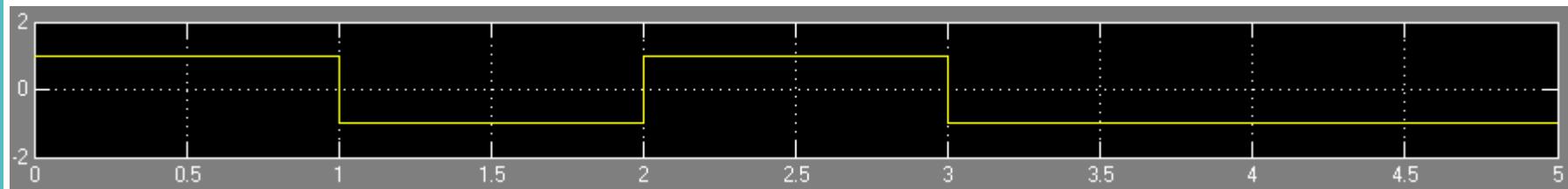
1

1

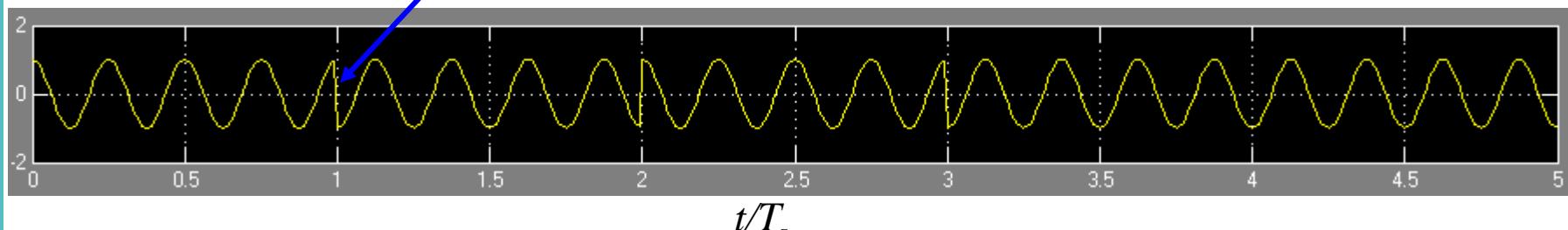
Symbol sequence  $d_n$ :



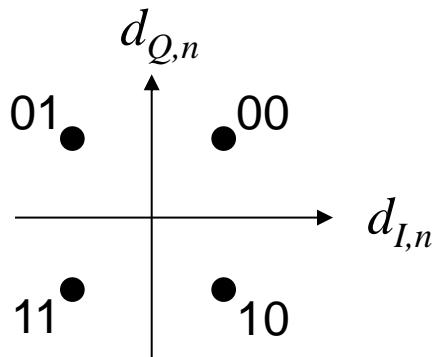
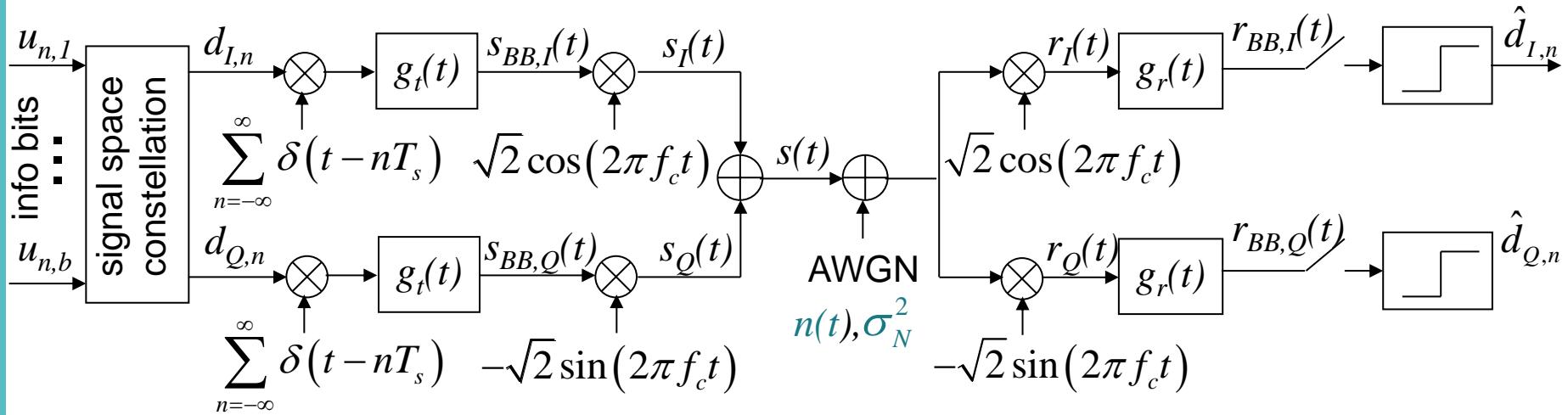
Equivalent baseband signal  $\sqrt{2}s_{BB}(t)$ :



Transmit signal  $s(t)$ : “phase shift keying”  $\Rightarrow$  info is in the carrier phase

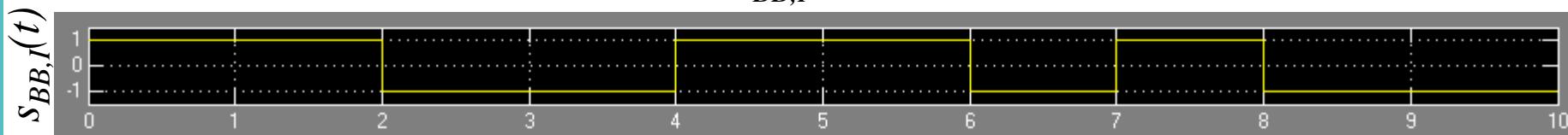


# Quadrature Phase Shift Keying (QPSK)

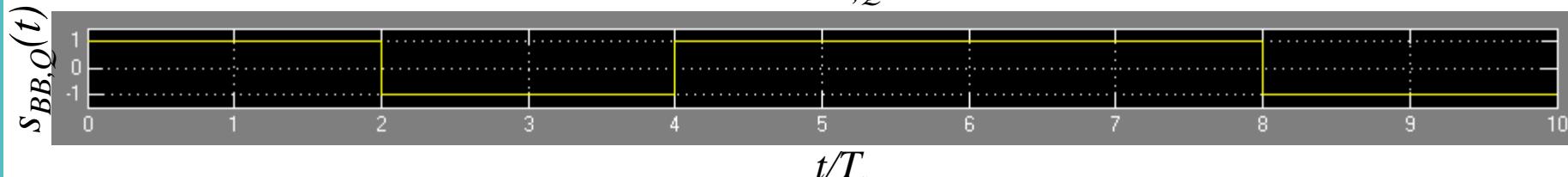


# Quadrature Phase Shift Keying (QPSK) with Rectangular NRZ Pulses

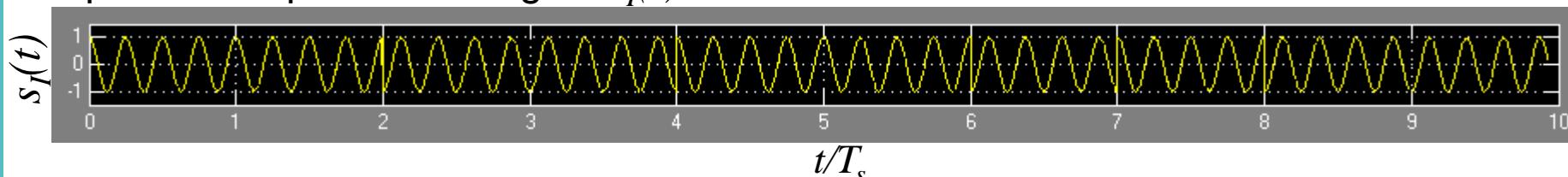
Inphase component baseband signal  $s_{BB,I}(t)$ :



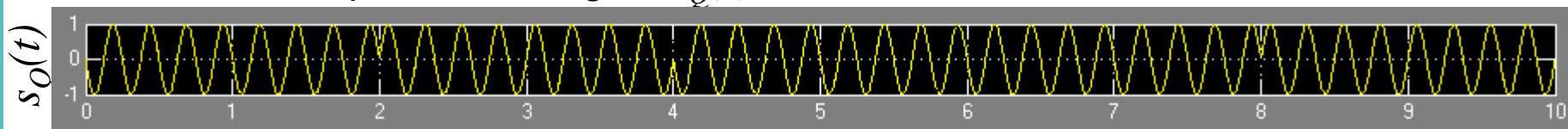
Quadrature component baseband signal  $s_{BB,Q}(t)$ :



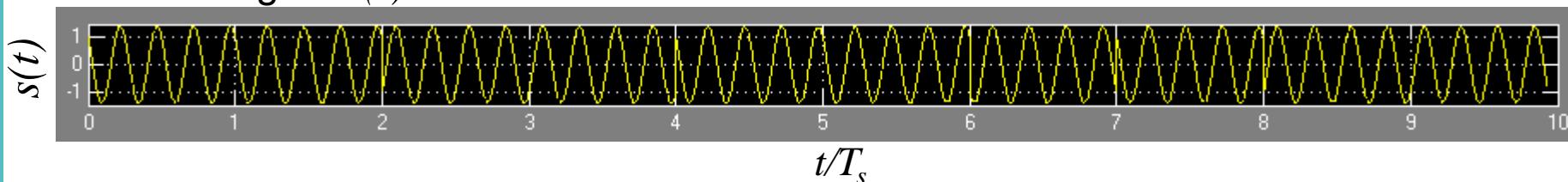
Inphase component RF signal  $s_I(t)$ :



Quadrature component RF signal  $s_Q(t)$ :

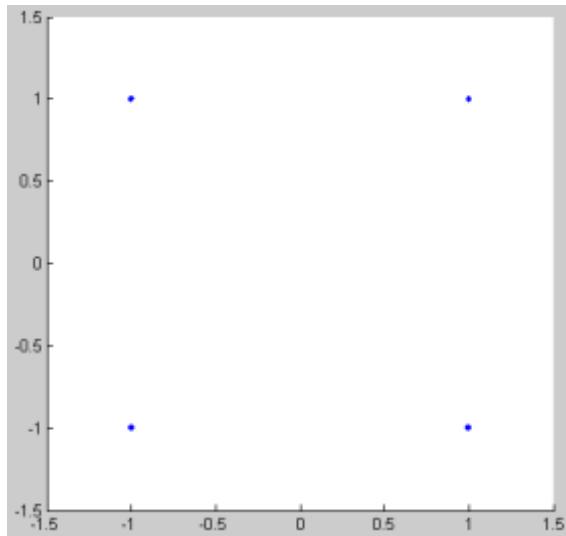


Transmit signal  $s(t)$ :

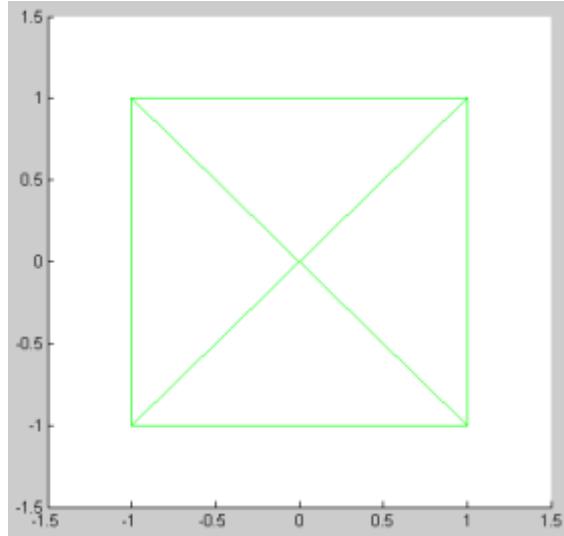


# Quadrature Phase Shift Keying (QPSK) with Rectangular NRZ Pulses

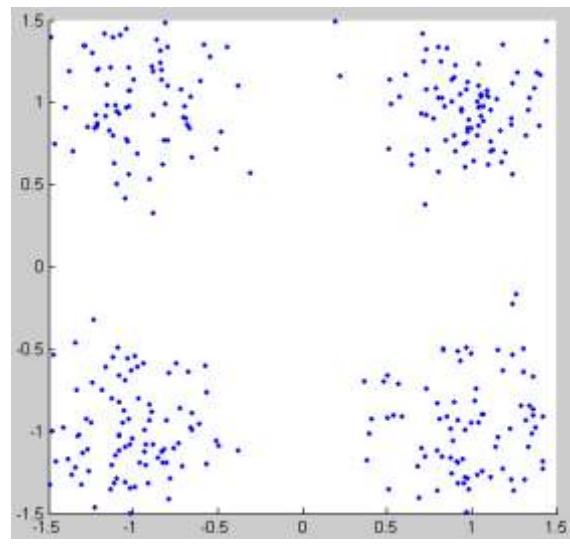
Signal space constellation



Baseband signal  
 $s_{BB}(t) = s_{BB,I}(t) + js_{BB,Q}(t)$   
in signal space

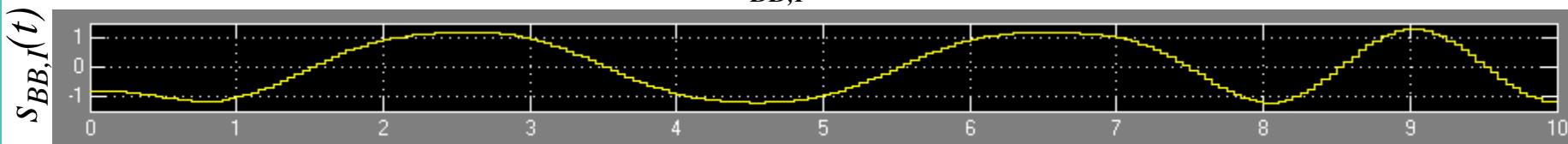


Noisy received points in  
signal space after sampling

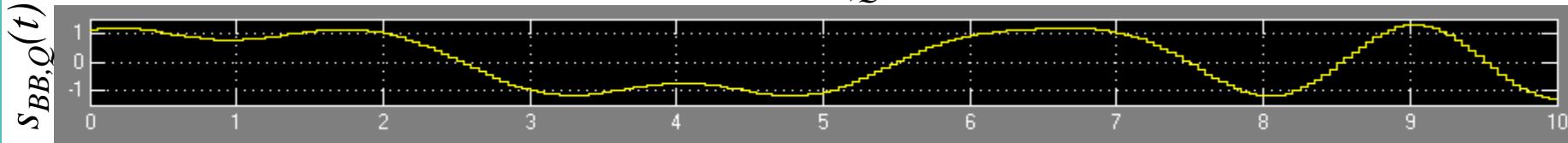


# Quadrature Phase Shift Keying (QPSK) with Square-Root Raised-Cosine Pulses – Signals at Transmitter

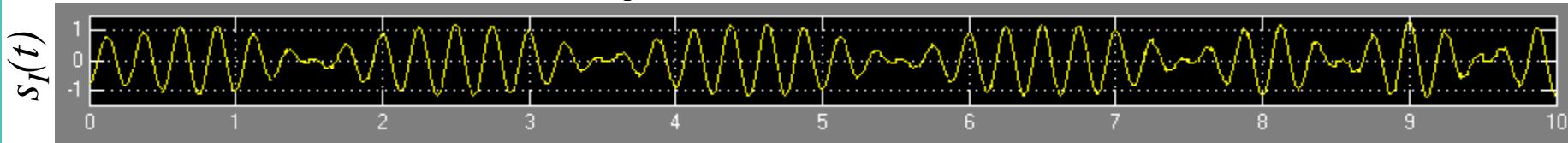
Inphase component baseband signal  $s_{BB,I}(t)$ :



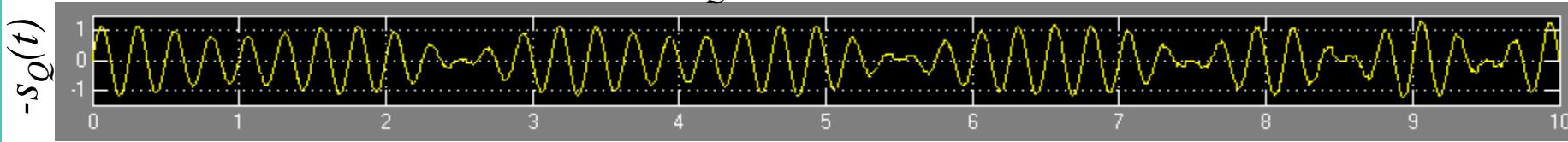
Quadrature component baseband signal  $s_{BB,Q}(t)$ :



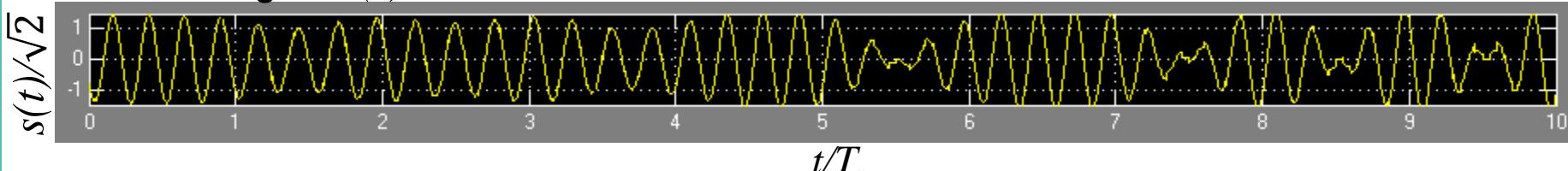
Inphase component RF signal  $s_I(t)$ :



Quadrature component RF signal  $-s_Q(t)$ :

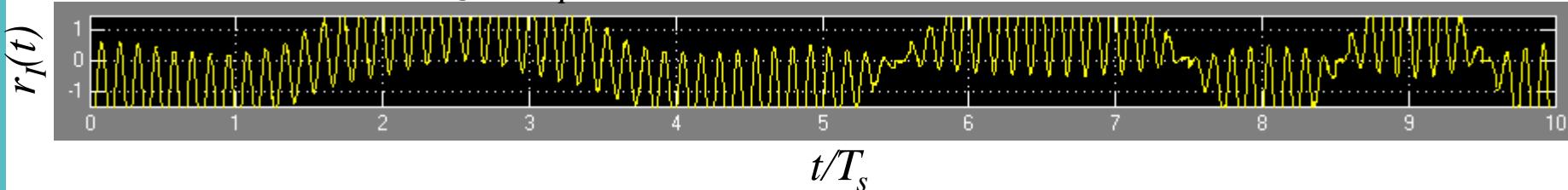


Transmit signal  $s(t)/\sqrt{2}$ :

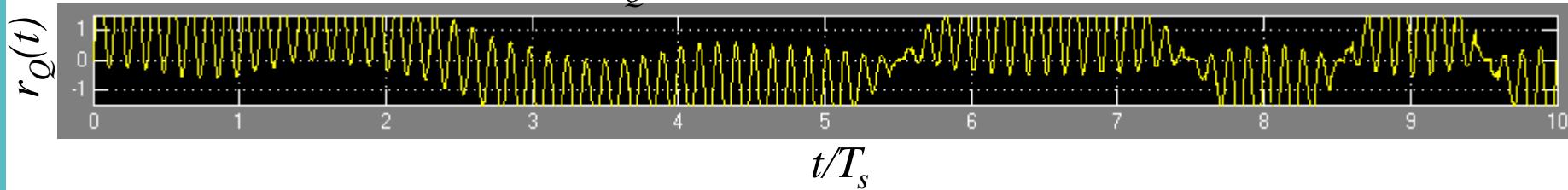


# Quadrature Phase Shift Keying (QPSK) with Square-Root Raised-Cosine Pulses – Signals at Receiver

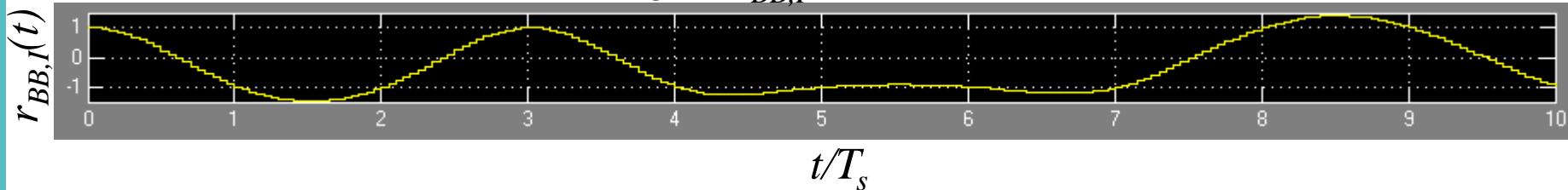
Inphase component signal  $r_I(t)$  at matched filter input:



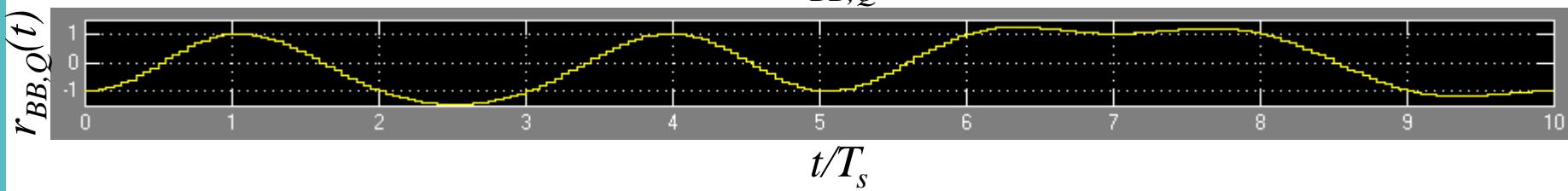
Quadrature component signal  $r_Q(t)$  at matched filter input:



Inphase component baseband signal  $r_{BB,I}(t)$  at matched filter output:



Quadrature component baseband signal  $r_{BB,Q}(t)$  at matched filter output:

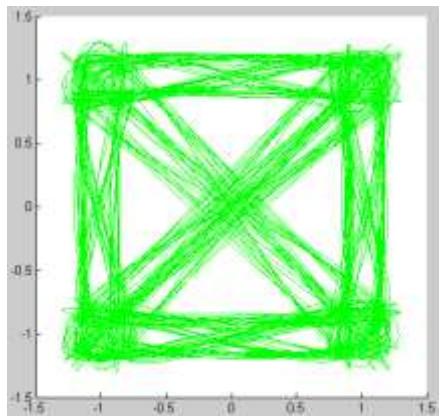


# Quadrature Phase Shift Keying (QPSK) with Square-Root Raised-Cosine Pulses

Baseband signal

$$s_{BB}(t) = s_{BB,I}(t) + j s_{BB,Q}(t)$$

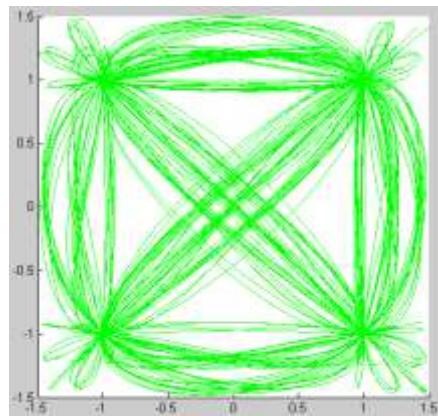
at transmitter  
in signal space:



Baseband signal

$$r_{BB}(t) = r_{BB,I}(t) + j r_{BB,Q}(t)$$

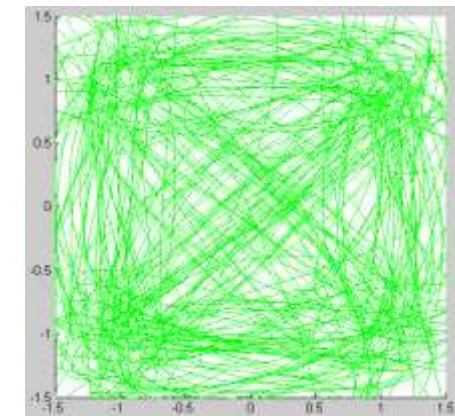
at matched filter output for  
noiseless transmission:



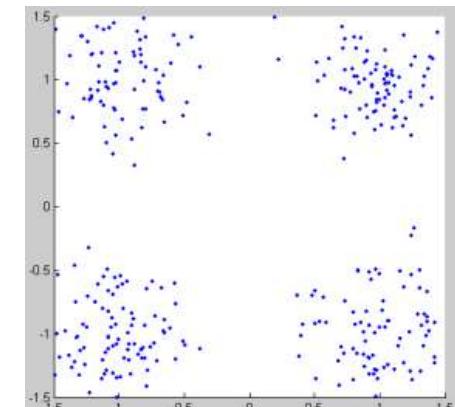
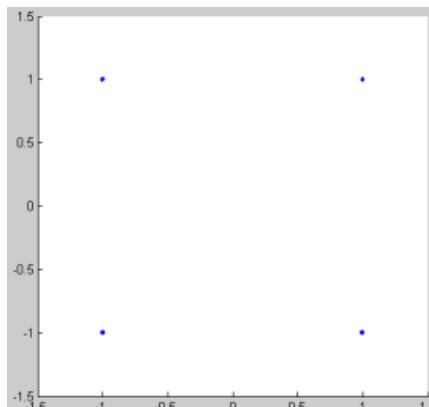
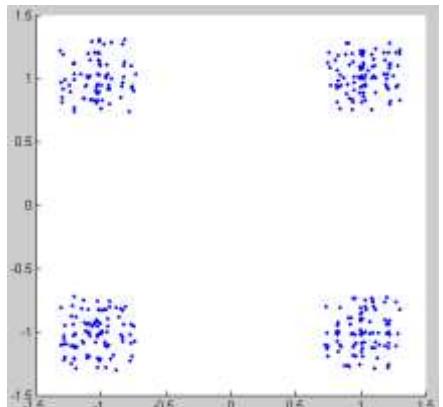
Baseband signal

$$r_{BB}(t) = r_{BB,I}(t) + j r_{BB,Q}(t)$$

at matched filter output  
for noisy transmission:



sampling at symbol rate  $1/T_s$ :



# Baseband Signal in Signal Space, Offset-QPSK

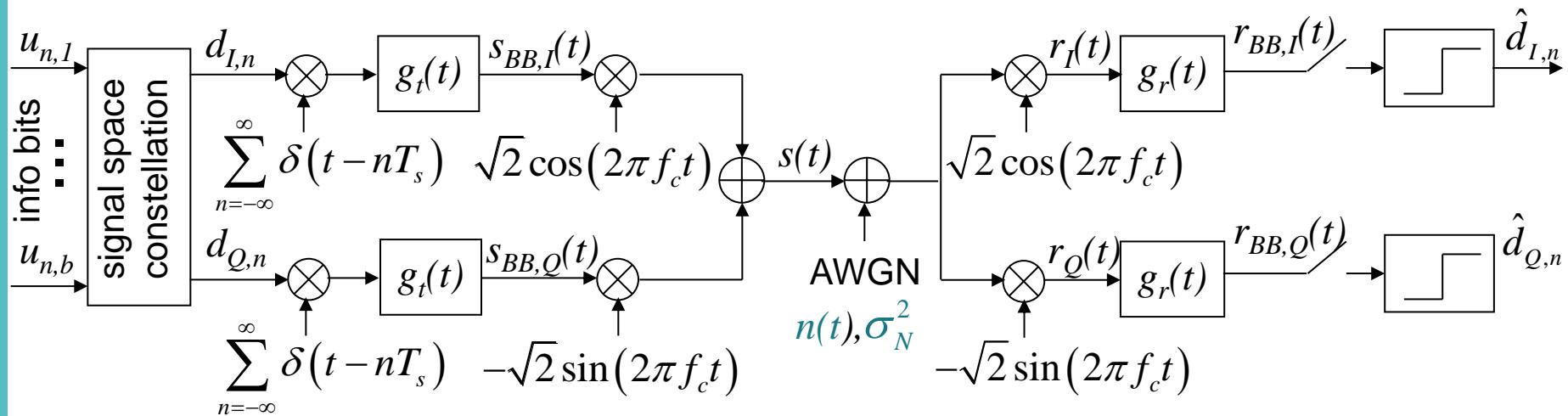
The complex equivalent baseband signal after pulse shaping can be described by a trajectory in the signal space. If the first Nyquist condition is met and no noise is present, all lines of the trajectory cross in the signal space constellation points at the sampling time and the trajectory describes the transitions between the constellation points. In this case, sampling the trajectory at the optimum sampling time yields the signal space constellation.

If the 1. Nyquist condition is not met or noise is present, sampling yields clouds around the signal space constellation points. The larger the variance of the clouds the higher is the probability of detection errors.

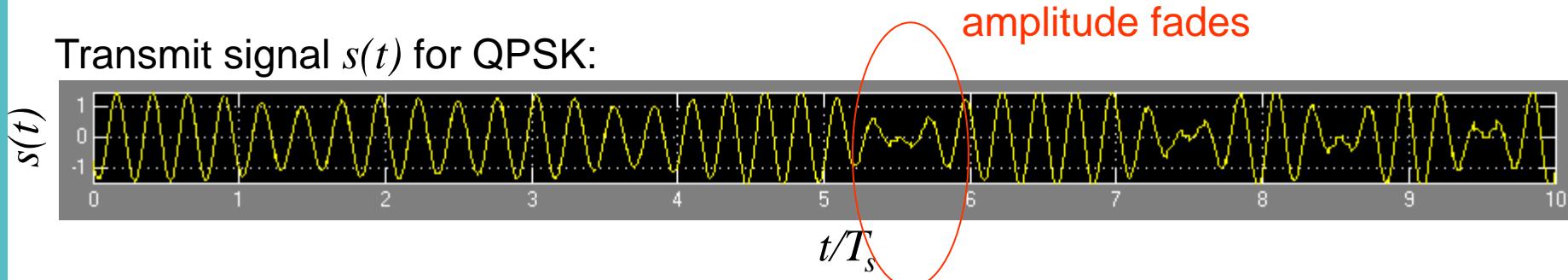
The magnitude of the trajectory, i.e. of the complex equivalent baseband signal, is the envelope of the transmit signal. In QPSK, we observe diagonal transitions where the trajectory goes almost through the origin. This means deep fades in the envelope of the transmit signal. Consequently, the transmitted signal becomes sensitive to nonlinear distortions as imposed by amplifiers. Therefore, many applications require an almost constant envelope of the transmit signal.

Since diagonal transitions happen when the data in both quadrature components changes simultaneously, a simple method to avoid deep fades is to shift the quadrature component by  $T_s/2$  relative to the inphase component. Doing so, inphase and quadrature component cannot change simultaneously anymore and diagonal transitions of the trajectory are avoided. This method is called *Offset-QPSK*.

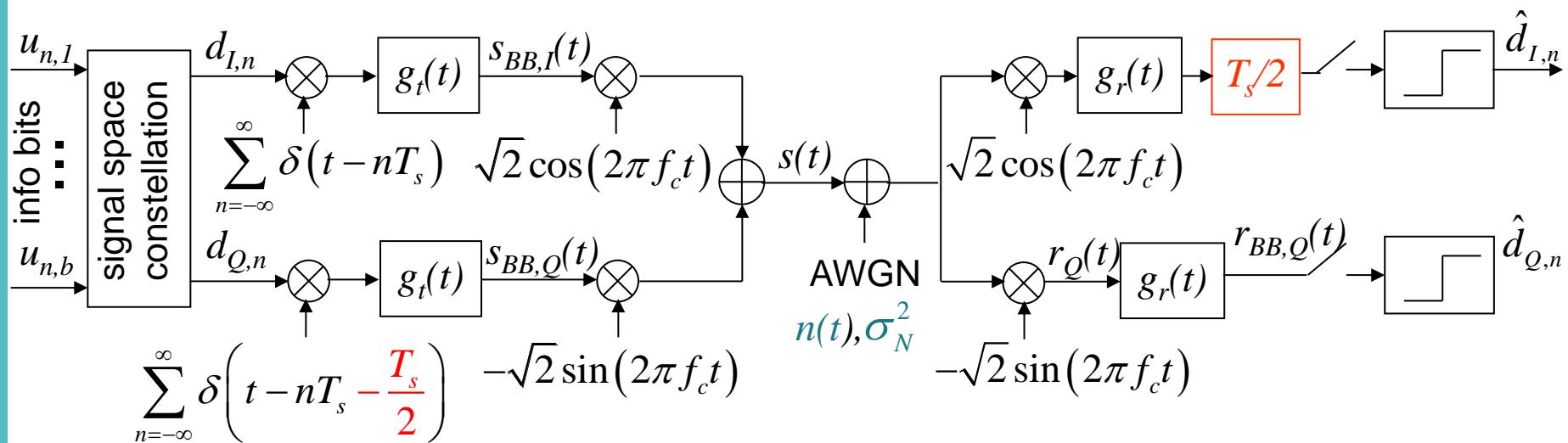
# Quadrature Phase Shift Keying (QPSK)



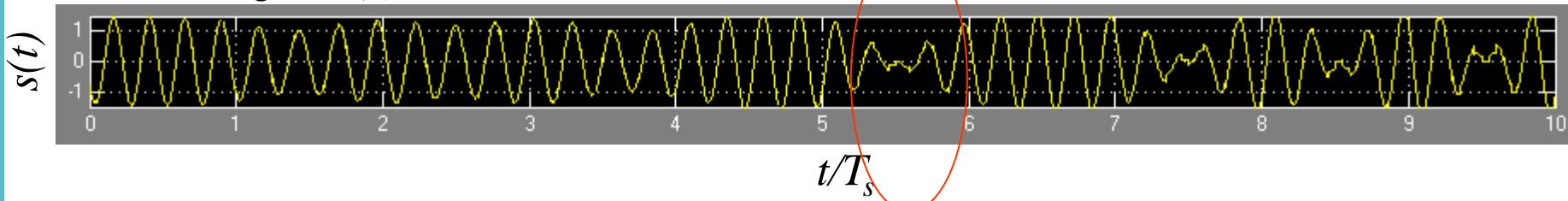
Transmit signal  $s(t)$  for QPSK:



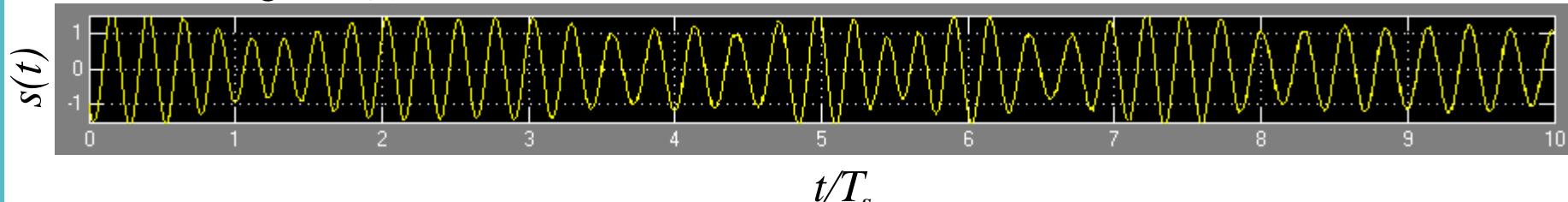
# Offset-QPSK



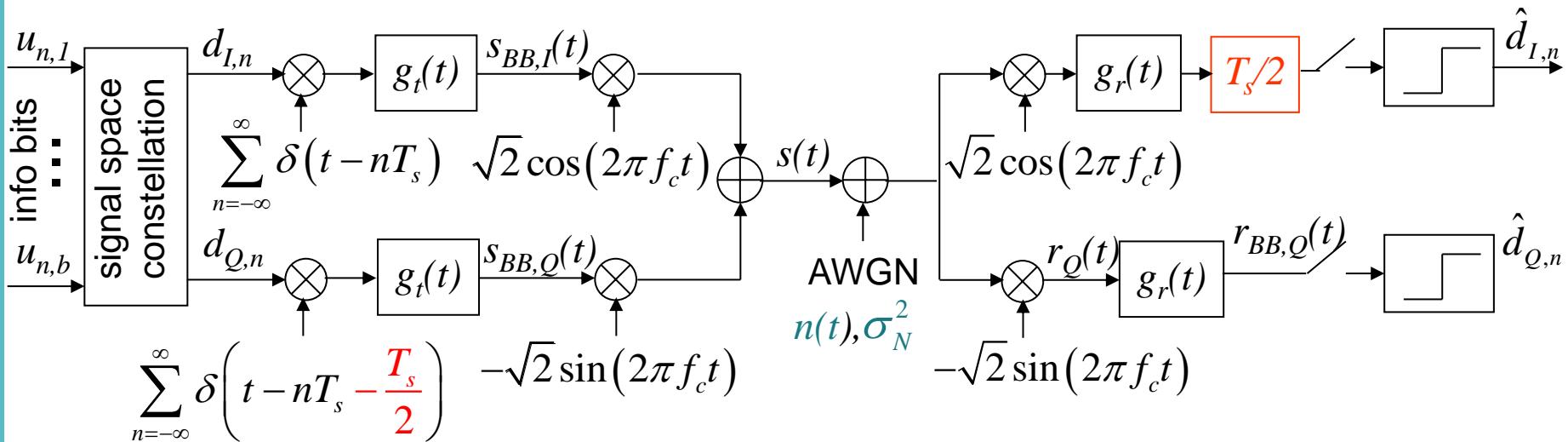
Transmit signal  $s(t)$  for QPSK:



Transmit signal  $s(t)$  for Offset-QPSK:

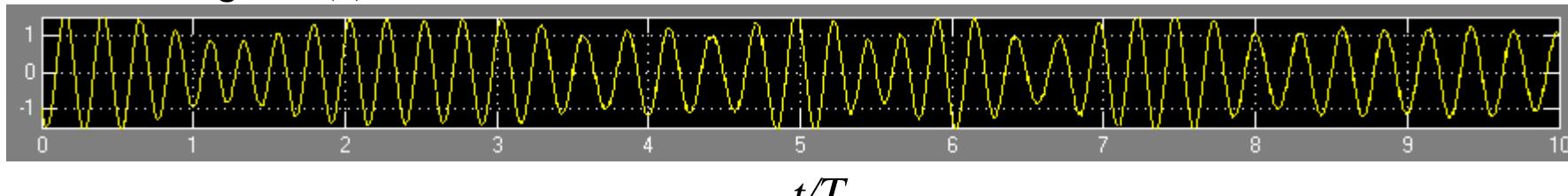


# Offset-QPSK



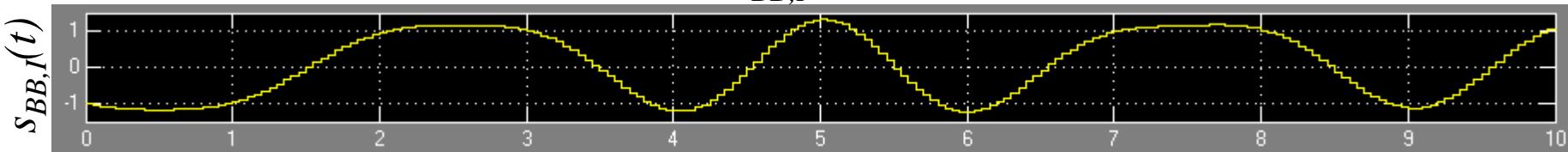
The quadrature component is delayed by  $T_s/2$  such that inphase and quadrature component cannot change at the same time. This avoids diagonal transitions of the baseband signal in the signal space and hence avoids deep fades in the transmit signal amplitude.

Transmit signal  $s(t)$  for Offset-QPSK:

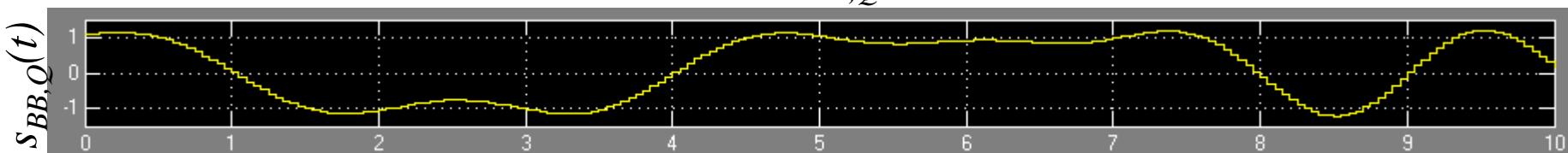


# Offset-QPSK with Square-Root Raised Cosine Pulses

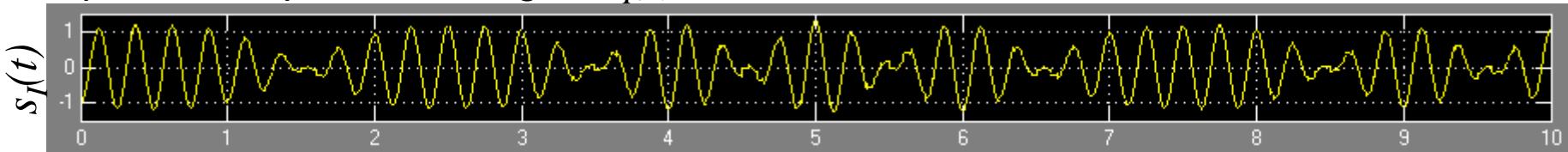
Inphase component baseband signal  $s_{BB,I}(t)$ :



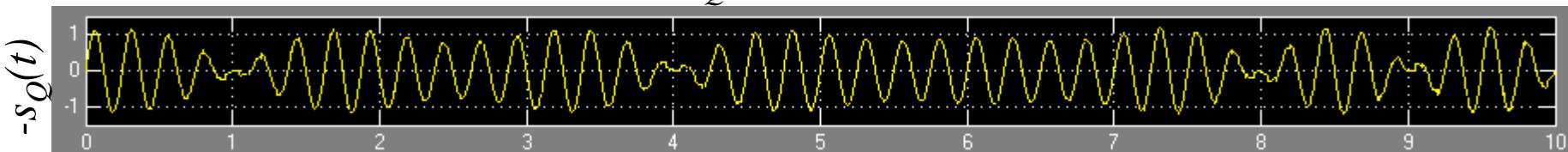
Quadrature component baseband signal  $s_{BB,Q}(t)$ :



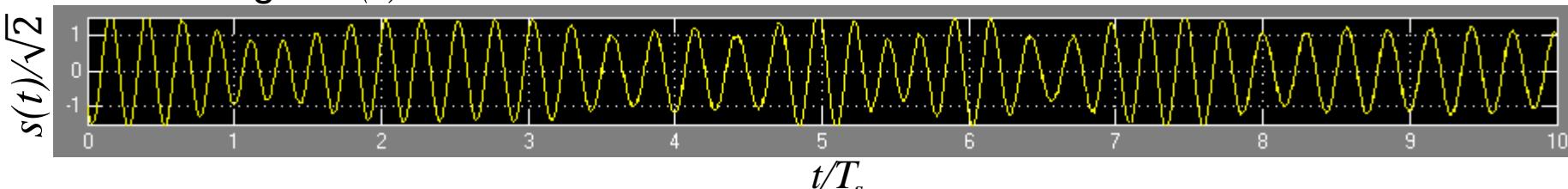
Inphase component RF signal  $s_I(t)$ :



Quadrature component RF signal  $-s_Q(t)$ :

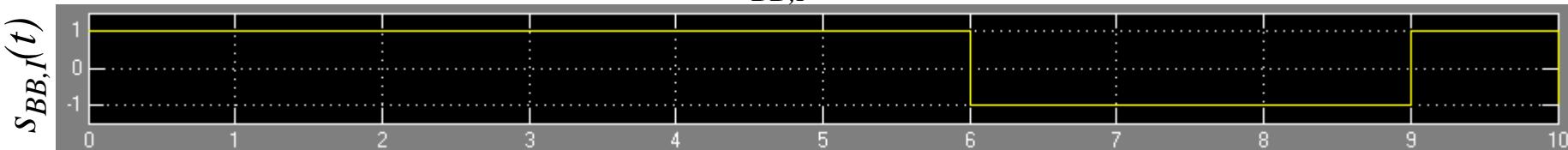


Transmit signal  $s(t)/\sqrt{2}$ :

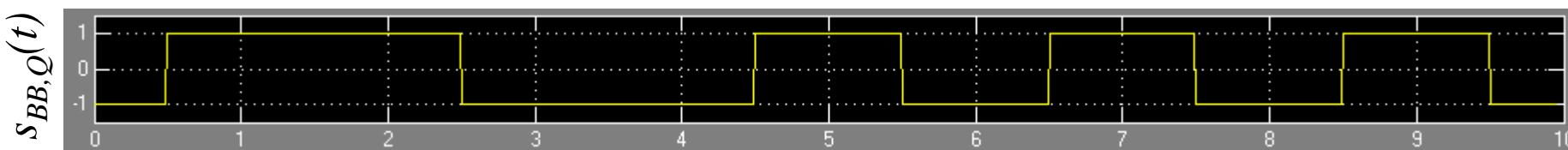


# Offset-QPSK with Rectangular NRZ Pulses

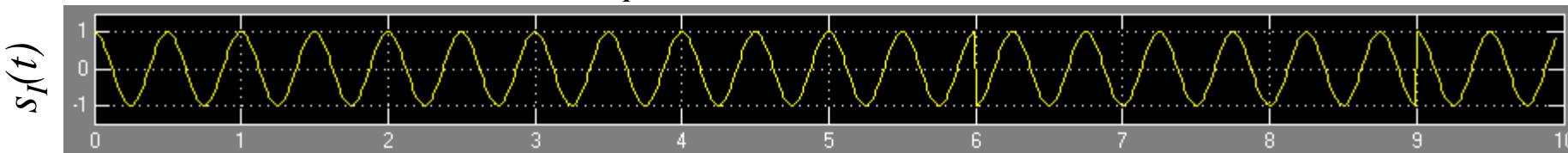
Inphase component baseband signal  $s_{BB,I}(t)$ :



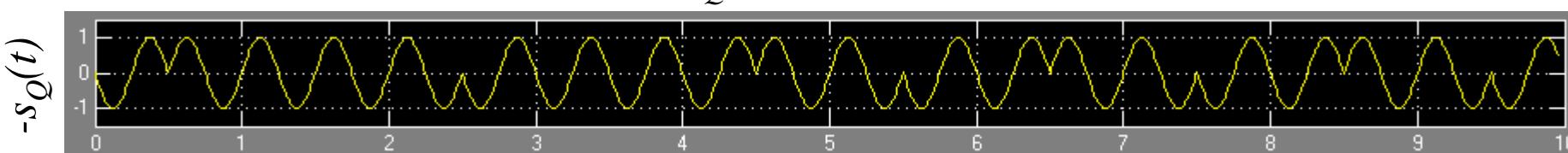
Quadrature component baseband signal  $s_{BB,Q}(t)$ :



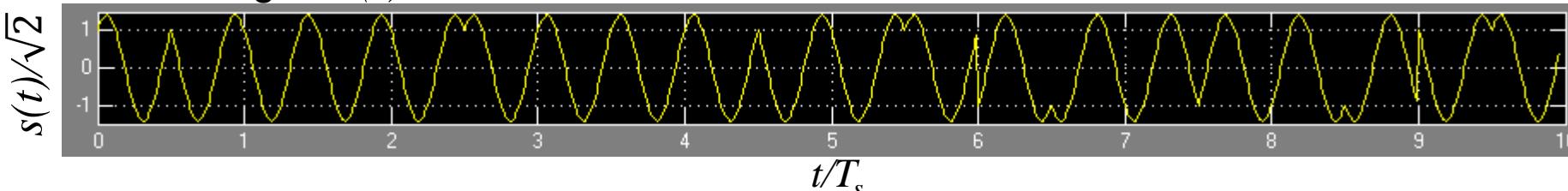
Inphase component RF signal  $s_I(t)$ :



Quadrature component RF signal  $-s_Q(t)$ :



Transmit signal  $s(t)/\sqrt{2}$ :

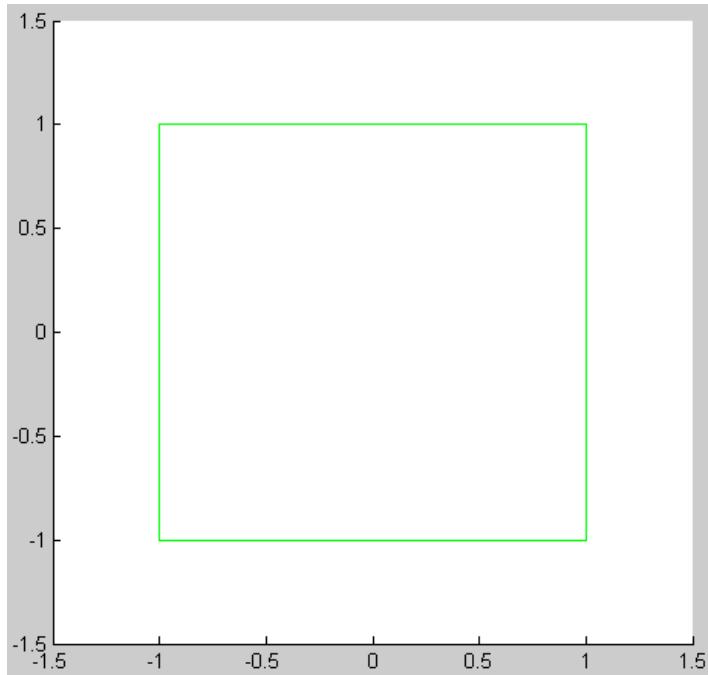


# Offset-QPSK

Baseband signal

$$s_{BB}(t) = s_{BB,I}(t) + j s_{BB,Q}(t)$$

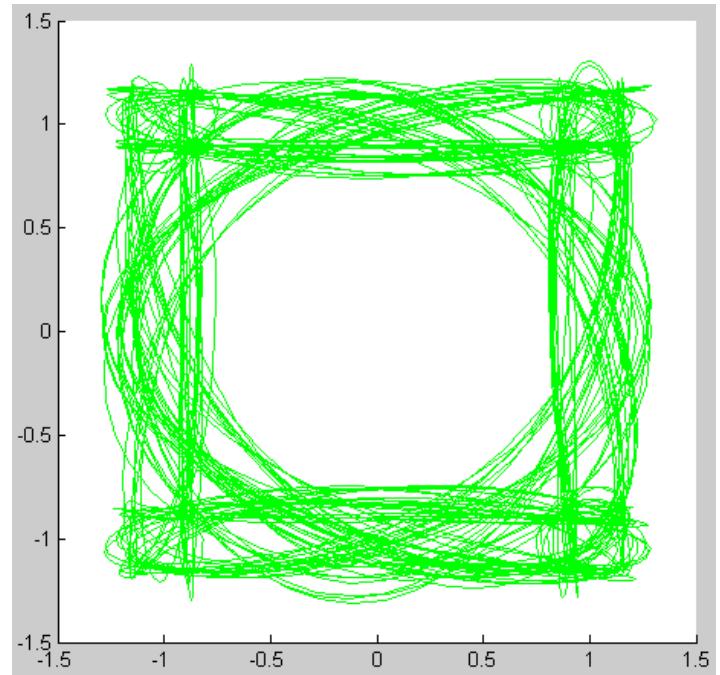
at transmitter in signal space for NRZ rectangular pulses:



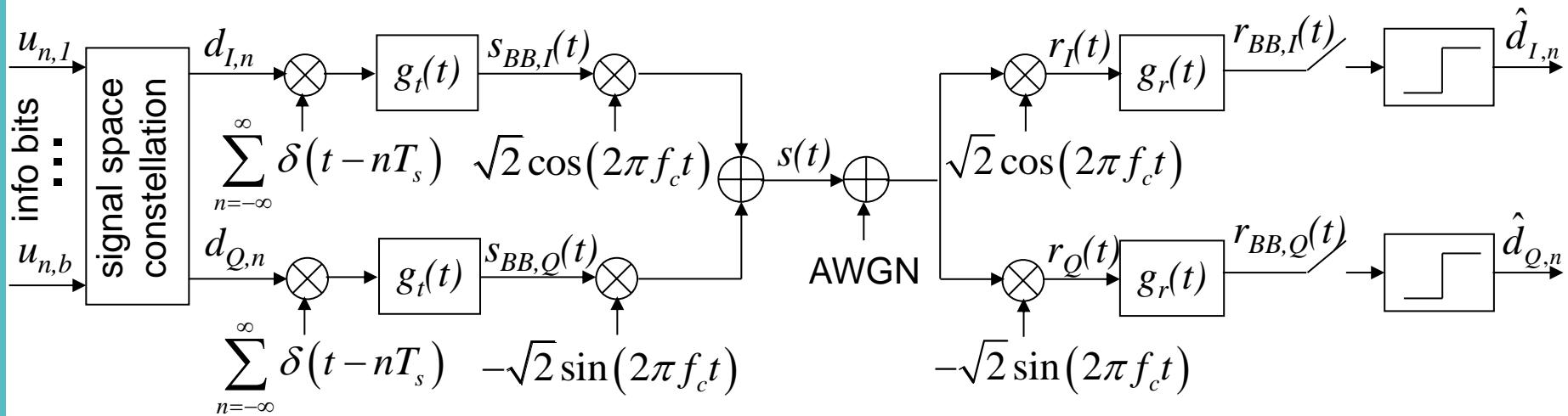
Baseband signal

$$s_{BB}(t) = s_{BB,I}(t) + j s_{BB,Q}(t)$$

at transmitter in signal space for square-root raised-cosine pulses:



# Equivalent Baseband Signal



$$\text{Transmit signal: } s(t) = s_{BB,I}(t)\sqrt{2} \cos(2\pi f_c t) - s_{BB,Q}(t)\sqrt{2} \sin(2\pi f_c t)$$

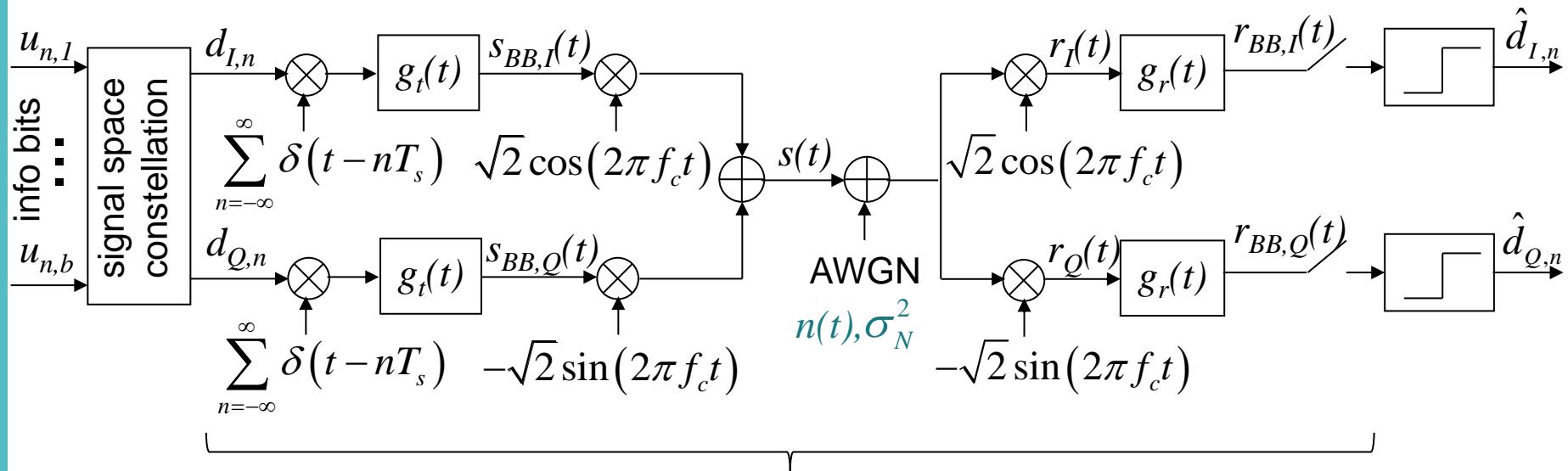
Equivalent baseband signal:

$$s_{BB}(t) = s_{BB,I}(t) + j s_{BB,Q}(t) = \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s) \quad d_n = d_{I,n} + j d_{Q,n}$$

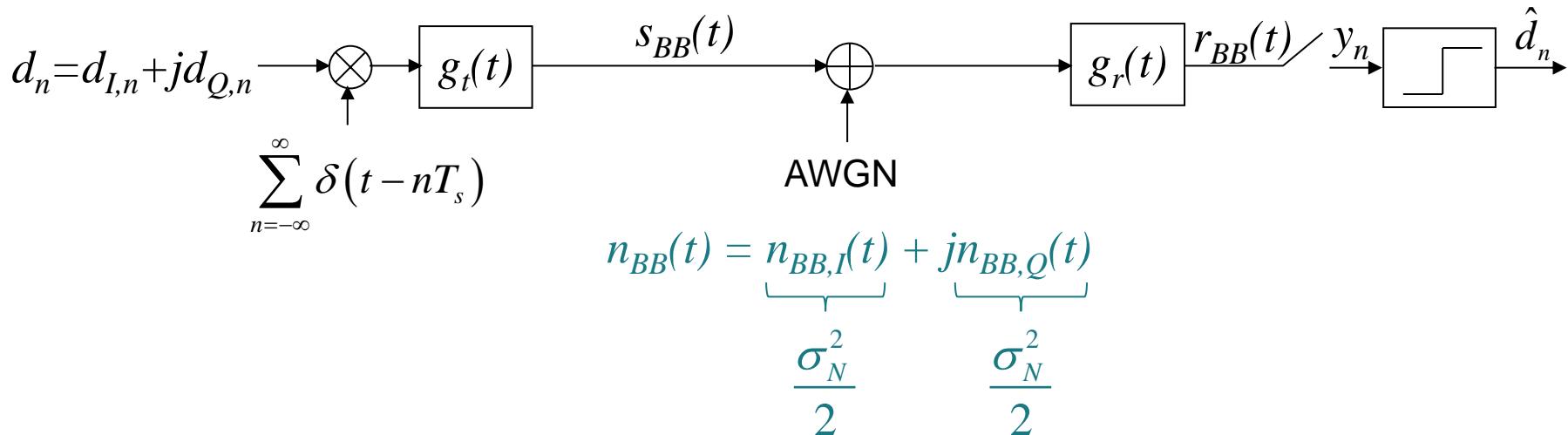
$$s(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\} = \operatorname{Re} \left\{ s_+(t) \right\}$$

$s_+(t)$  is the analytical signal, i.e.  $s_+(t) \circ \bullet S_+(f) = \begin{cases} 0 & \text{for } f < 0 \\ 2S(f) & \text{for } f \geq 0 \end{cases}$

# Equivalent Baseband System

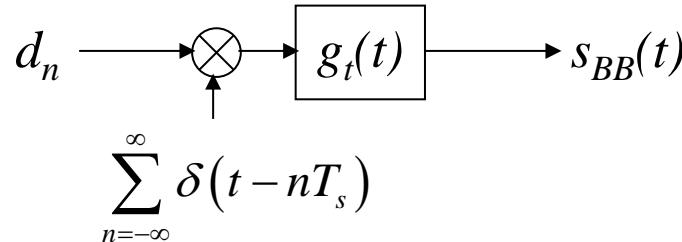


Equivalent baseband system:



# Energy of Linear Digital Modulated Signals

Equivalent baseband model:



$$\text{Transmit signal: } s(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\}$$

Energy of transmit signal:

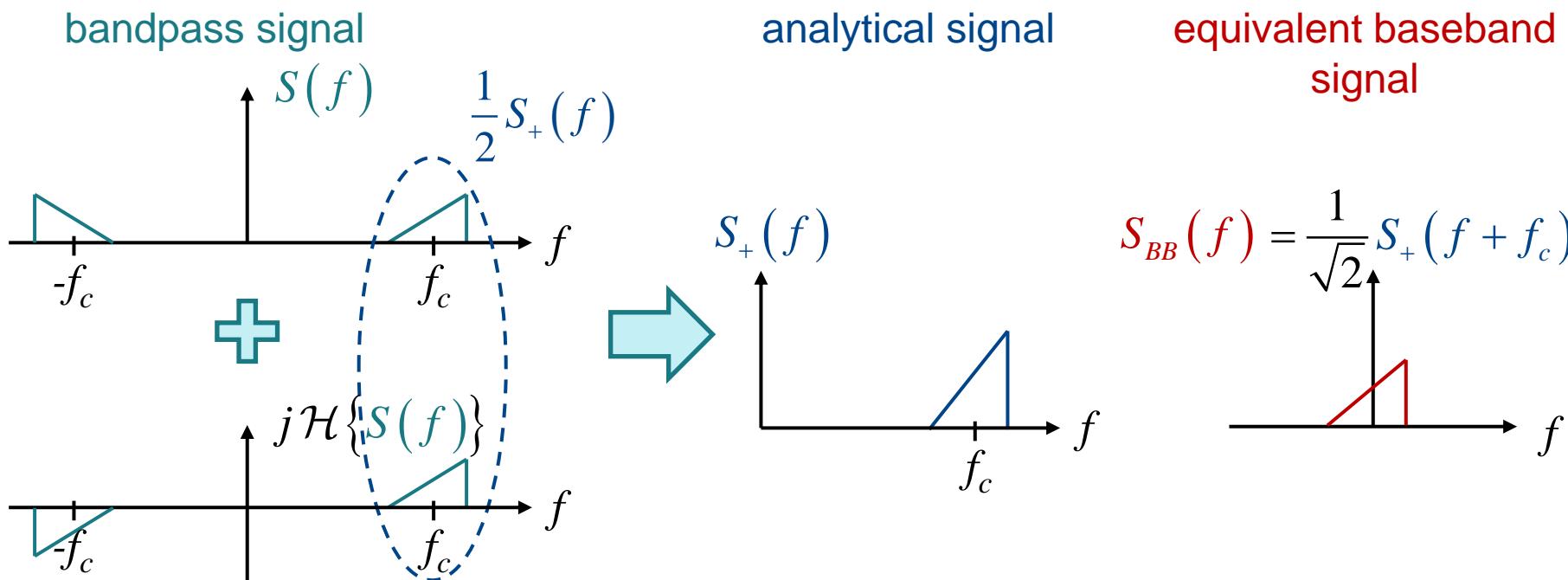
$$E = \int_{-\infty}^{\infty} s^2(t) dt = \int_{-\infty}^{\infty} \left( \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\} \right)^2 dt = \int_{-\infty}^{\infty} |s_{BB}(t)|^2 dt$$

Average energy per symbol:

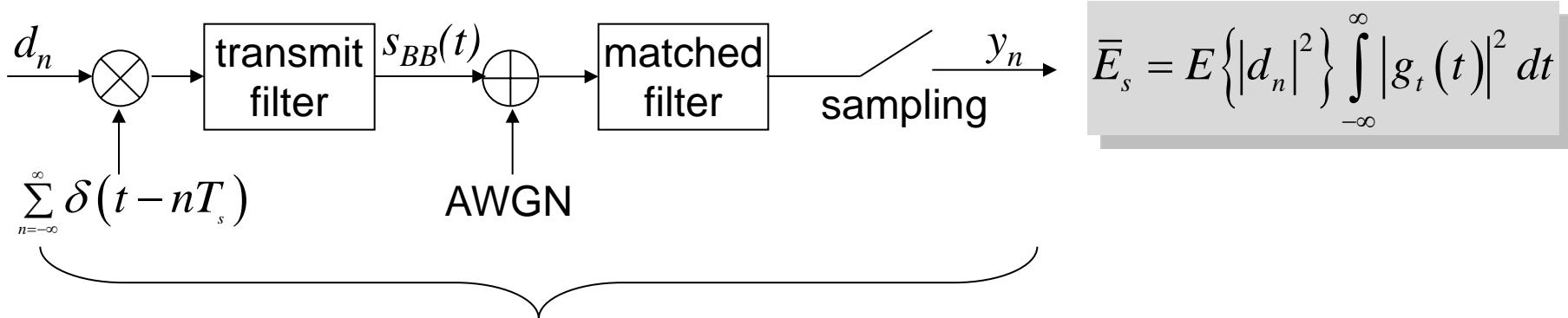
$$\bar{E}_s = E \left\{ |d_n|^2 \right\} \int_{-\infty}^{\infty} |g_t(t)|^2 dt$$

# Derivation of Energy of Linear Digital Modulated Signals

$$\begin{aligned}
 E &= \int_{-\infty}^{\infty} s^2(t) dt = \int_{-\infty}^{\infty} |S(f)|^2 df = 2 \int_0^{\infty} |S(f)|^2 df = 2 \int_0^{\infty} \left| \frac{1}{2} S_+(f) \right|^2 df = 2 \int_{-\infty}^{\infty} \left| \frac{1}{2} \cdot \sqrt{2} S_{BB}(f) \right|^2 df \\
 &\quad \text{Parseval's theorem} \quad s(t) \text{ real} \\
 &\quad \qquad \qquad \rightarrow |S(f)| = |S(-f)| \\
 &= \int_{-\infty}^{\infty} |S_{BB}(f)|^2 df = \int_{-\infty}^{\infty} |s_{BB}(t)|^2 dt
 \end{aligned}$$



# Equivalent Baseband Discrete Time AWGN Channel Model



Equivalent baseband discrete time channel model:

$$E \left\{ |d_n|^2 \right\} = \frac{\bar{E}_s}{T_s}$$

$$n_{BB,n} = n_{BB,I,n} + jn_{BB,Q,n},$$

$$\sigma_{N_{BB,I}}^2 = \sigma_{N_{BB,Q}}^2 = \frac{\sigma_N^2}{2} = \frac{N_0}{2T_s}$$

Equivalent channel model:

$$E \left\{ |d_n|^2 \right\} = 1$$

$$n_{BB,n} = n_{BB,I,n} + jn_{BB,Q,n},$$

$$\sigma_{BB,I}^2 = \sigma_{BB,Q}^2 = \frac{N_0}{2T_s} \frac{T_s}{\bar{E}_s} = \frac{N_0}{2\bar{E}_s}$$

# Power Spectral Density of Linear Digital Modulated Signals (1)

In baseband transmission, the average power spectral density

$$\bar{S}_{ss}(f) = \frac{1}{T_s} E \left\{ |d_n|^2 \right\} |G_t(f)|^2$$

is determined by the pulse shaping filter  $G_t(f)$ . The only difference in structure between a real baseband signal and the equivalent baseband signal  $s_{BB}(t)$  of a linear digital modulated bandpass signal  $s(t)$  is, that equivalent baseband signals, in particular the transmit symbols  $d_n$ , are complex. However, the general case of complex signals had already been taken into account in the derivation of the power spectral density given above. Consequently, the power spectral density of the equivalent baseband signal is given by the same equation, i.e.

$$\bar{S}_{s_{BB}s_{BB}}(f) = \frac{1}{T_s} E \left\{ |d_n|^2 \right\} |G_t(f)|^2$$

The real bandpass signal

$$s(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\}$$

of a linear digital modulation scheme is obtained by shifting the equivalent baseband signal to the passband around the carrier frequency  $f_c$ .

# Power Spectral Density of Linear Digital Modulated Signals (2)

Consequently, the power spectral density is also shifted to the passband around the carrier frequency  $f_c$ . This does not change the shape of the power spectral density. Since the bandpass signal is real, the power spectral density is an even function. This is reflected in the fact that the average power density spectrum of the equivalent baseband signal is shifted not only to the carrier frequency  $f_c$  but also to  $-f_c$  and the average power spectral density of the bandpass signal yields

$$\bar{S}_{ss}(f) = \frac{1}{2} \left[ \bar{S}_{s_{BB}s_{BB}}(f - f_c) + \bar{S}_{s_{BB}s_{BB}}(-(f + f_c)) \right].$$

The factor  $\frac{1}{2}$  makes sure that the total average transmit power of equivalent baseband signal and bandpass signal remains the same.

The bandwidth is again defined as the occupied frequency range at positive frequencies. Consequently, i.e. with square-root raised cosine pulses (roll-off factor  $\alpha$ ), the bandwidth is  $(1+\alpha)/T_s$ , whereas in baseband transmission we had  $(1+\alpha)/(2T_s)$ .

It should be noted that shape and in particular bandwidth of a linear digital modulation scheme only depends on the pulse shaping filter  $G_t(f)$  as long as the data is uncorrelated and has zero mean. I.e., BPSK, QPSK, 16-QAM all have the same average power spectral density if the same pulse shaping filter and symbol duration  $T_s$  is applied, even though they transmit a different number  $b = \log_2 M$  of bits in each symbol  $d_n$ .

From this perspective, it becomes clear that the data rate  $b/T_s$  and the bandwidth  $B$  are not the same. This is important to mention as in some communities, data rate and bandwidth are erroneously used synonymously. This may be a relict from the days when transmission was only binary.

# Power Spectral Density of Linear Digital Modulated Signals

Assumption: Uncorrelated data with zero mean:

RF transmit signal:  $s(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\}; \quad s_{BB}(t) = \sum_{n=-\infty}^{\infty} d_n g_t(t - nT_s)$

Pulse shaping filter:  $g_t(t)$    $G_t(f)$

**Average autocorrelation function of equivalent baseband signal:**

$$\bar{r}_{s_{BB}s_{BB}}(\tau) = \frac{1}{T_s} E \left\{ d_n^* d_n \right\} (g_t^*(-\tau) * g_t(\tau)) \quad \text{---} \bullet$$

**Average power spectral density of equivalent baseband signal:**

$$\bar{S}_{s_{BB}s_{BB}}(f) = \frac{1}{T_s} E \left\{ d_n^* d_n \right\} |G_t(f)|^2$$

**Average power spectral density of RF signal:**

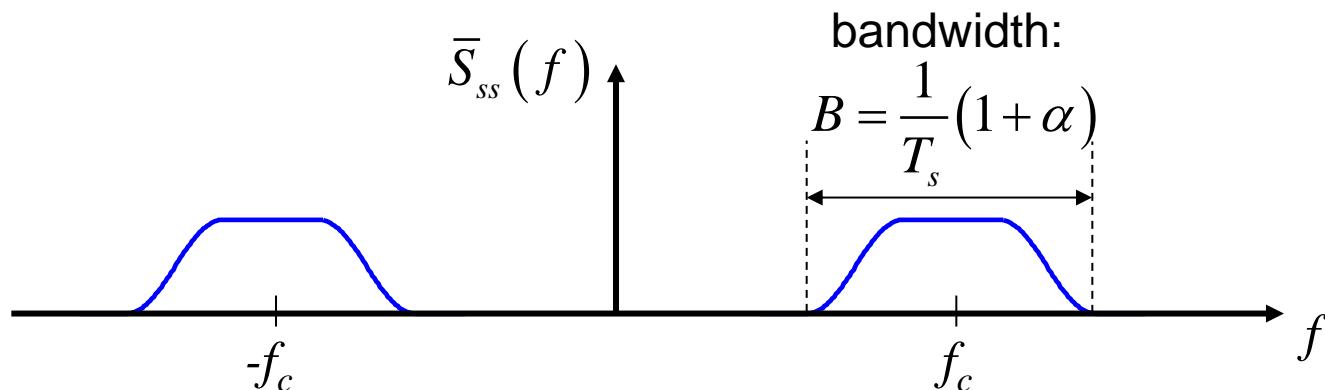
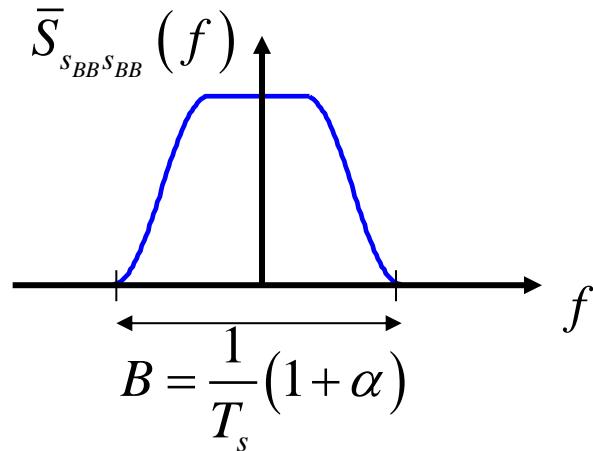
$$\bar{S}_{ss}(f) = \frac{1}{2} \left[ \bar{S}_{s_{BB}s_{BB}}(f - f_c) + \bar{S}_{s_{BB}s_{BB}}(-(f + f_c)) \right]$$

As in the case of baseband transmission, the average power spectral density is determined by the pulse shaping filter  $G_t(f)$ . It can be determined via the power spectral density of the equivalent baseband signal.

# Power Spectral Density of Linear Digital Modulated Signals - Example

$$\bar{S}_{ss}(f) = \frac{1}{2} \left[ \bar{S}_{s_{BB}s_{BB}}(f - f_c) + \bar{S}_{s_{BB}s_{BB}}(-(f + f_c)) \right]$$

Square-root-raised-cosine pulse with roll-off factor  $\alpha=0.5$ :



# Bandwidth - Efficiency

$$\eta = \frac{\text{data rate}}{\text{required bandwidth}}$$

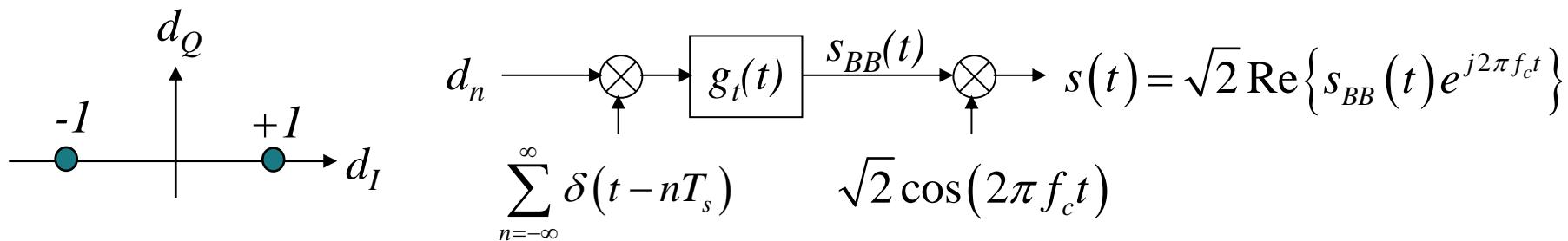
Example:

- $M$ -ary transmission
- (square-root) raised-cosine pulses with roll-off factor  $\alpha$

$$\eta = \frac{\frac{1}{T_s} \log_2 M}{1 + \alpha} = \frac{\log_2 M}{1 + \alpha} \frac{\text{bit}}{\text{s} \cdot \text{Hz}}$$

	$M$	$\eta$ in bit/s/Hz
BPSK	2	$1/(1+\alpha)$
QPSK	4	$2/(1+\alpha)$
16-QAM	16	$4/(1+\alpha)$

# Elementary Signals – Example: BPSK



Two elementary transmit signals:

$$d_0 = +1 \Rightarrow s_{+1}(t) = (+1) \cdot g_t(t) \cdot \sqrt{2} \cos(2\pi f_c t)$$

$$d_0 = -1 \Rightarrow s_{-1}(t) = (-1) \cdot g_t(t) \cdot \sqrt{2} \cos(2\pi f_c t)$$

How well are the elementary signals discriminable ?  
 → correlation coefficient

In general for  $M$ -ary modulation:

$$\begin{aligned}
 s_\mu(t) &= \sqrt{2} \operatorname{Re} \left\{ s_{BB,\mu}(t) e^{j2\pi f_c t} \right\} \\
 &= \sqrt{2} \operatorname{Re} \left\{ d_{\mu,0}(t) g_t(t) e^{j2\pi f_c t} \right\} ; \quad \mu = 0, \dots, M-1
 \end{aligned}$$

# Correlation Coefficient

$$\rho_{\mu\nu} = \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_{\mu}^2(t) dt \cdot \int_{-\infty}^{\infty} s_{\nu}^2(t) dt}}$$

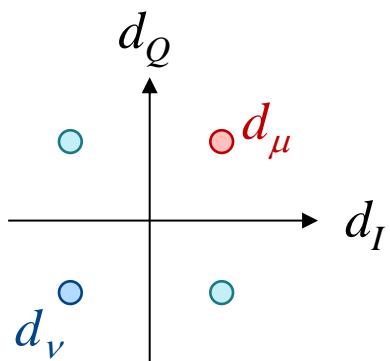
$\rho_{\mu\nu} = +1 \rightarrow$  fully correlated signals

$\rho_{\mu\nu} = 0 \rightarrow$  orthogonal signals

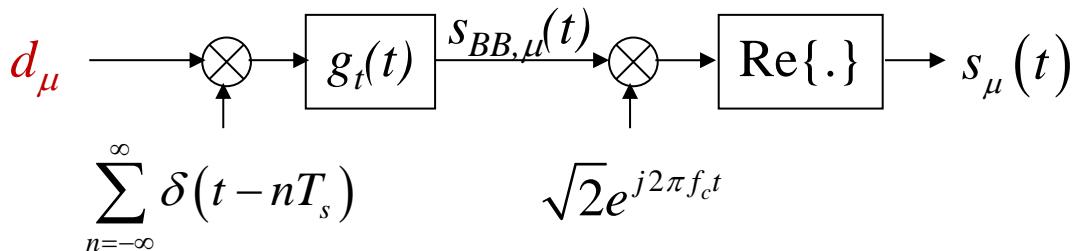
$\rho_{\mu\nu} = -1 \rightarrow$  antipodal signals

# Correlation Coefficient of Elementary Signals

$$\rho_{\mu\nu} = \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_{\mu}^2(t) dt \cdot \int_{-\infty}^{\infty} s_{\nu}^2(t) dt}} = \frac{\operatorname{Re}\{d_{\mu} d_{\nu}^*\}}{|d_{\mu}| \cdot |d_{\nu}|}$$



Elementary signal  $s_{\mu}(t)$  for a specific constellation point  $d_{\mu}$ :



# Derivation of Correlation Coefficient of Modulated Elementary Signals (1)

$$\begin{aligned}
 \rho_{\mu\nu} &= \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}(t) dt}{\sqrt{\left[ \int_{-\infty}^{\infty} s_{\mu}^2(t) dt \right] \cdot \left[ \int_{-\infty}^{\infty} s_{\nu}^2(t) dt \right]}} = \frac{\int_{-\infty}^{\infty} \sqrt{2} \operatorname{Re}\{s_{BB,\mu}(t) e^{j2\pi f_c t}\} \sqrt{2} \operatorname{Re}\{s_{BB,\nu}(t) e^{j2\pi f_c t}\} dt}{\sqrt{E_{\mu} E_{\nu}}} \\
 &= \frac{2}{\sqrt{E_{\mu} E_{\nu}}} \int_{-\infty}^{\infty} \frac{1}{2} \left[ s_{BB,\mu}(t) e^{j2\pi f_c t} + s_{BB,\mu}^*(t) e^{-j2\pi f_c t} \right] \cdot \frac{1}{2} \left[ s_{BB,\nu}(t) e^{j2\pi f_c t} + s_{BB,\nu}^*(t) e^{-j2\pi f_c t} \right] dt \\
 &= \frac{1}{2\sqrt{E_{\mu} E_{\nu}}} \left[ \underbrace{\int_{-\infty}^{\infty} s_{BB,\mu}(t) s_{BB,\nu}(t) e^{j4\pi f_c t} dt}_{=0} + \underbrace{\int_{-\infty}^{\infty} s_{BB,\mu}^*(t) s_{BB,\nu}^*(t) e^{-j4\pi f_c t} dt}_{=0} \right. \\
 &\quad \left. + \int_{-\infty}^{\infty} s_{BB,\mu}(t) s_{BB,\nu}^*(t) + s_{BB,\mu}^*(t) s_{BB,\nu}(t) dt \right] = \frac{1}{\sqrt{E_{\mu} E_{\nu}}} \cdot \operatorname{Re} \left\{ \int_{-\infty}^{\infty} s_{BB,\mu}(t) s_{BB,\nu}^*(t) dt \right\} \\
 &\quad \leftarrow \text{assumption: } s_{BB,\mu}(t) s_{BB,\nu}(t) \approx \text{const over one period } 1/f_c \text{ of oscillation}
 \end{aligned}$$

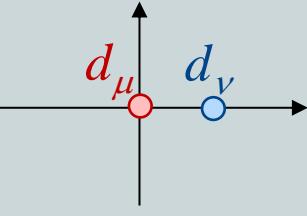
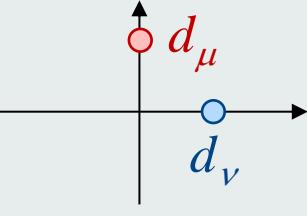
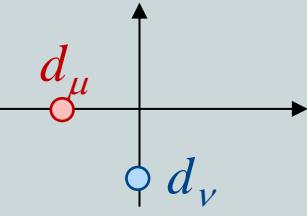
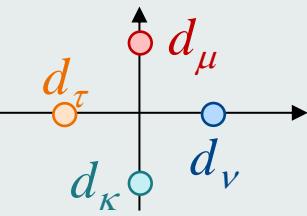
# Derivation of Correlation Coefficient of Modulated Elementary Signals (2)

$$\rho_{\mu\nu} = \frac{1}{\sqrt{E_\mu E_\nu}} \cdot \operatorname{Re} \left\{ \int_{-\infty}^{\infty} s_{BB,\mu}(t) s_{BB,\nu}^*(t) dt \right\} = \frac{\operatorname{Re} \left\{ d_\mu d_\nu^* \int_{-\infty}^{\infty} g_t(t) g_t^*(t) dt \right\}}{\sqrt{|d_\mu|^2 |d_\nu|^2 \left[ \int_{-\infty}^{\infty} |g_t(t)|^2 dt \right]^2}} = \frac{\operatorname{Re} \{ d_\mu d_\nu^* \}}{|d_\mu| \cdot |d_\nu|}$$

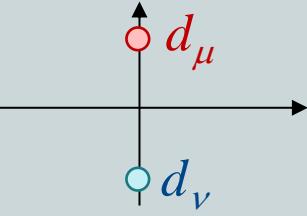
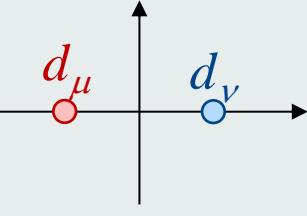
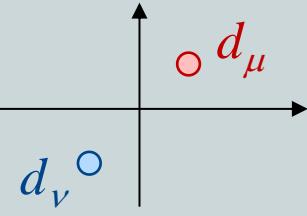
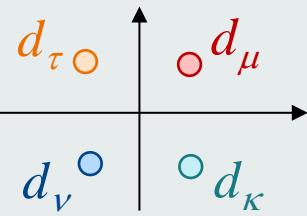
$$s_{BB,\mu}(t) = d_\mu g_t(t)$$

$$E_\mu = |d_\mu|^2 \int_{-\infty}^{\infty} |g_t(t)|^2 dt$$

# Correlation Coefficient of Elementary Signals

Constellation	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{d_\mu d_\nu^*\}}{ d_\mu  \cdot  d_\nu }$	Correlation
	$\tilde{\rho}_{\mu\nu} = \operatorname{Re}\{0 \cdot 1\} = 0$	orthogonal
	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{j \cdot 1\}}{1 \cdot 1} = 0$	orthogonal
	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{(-1) \cdot (-j)\}}{1 \cdot 1} = 0$	orthogonal
	$\rho_{\mu\nu} = \rho_{\mu\tau} = \rho_{\kappa\nu} = \rho_{\kappa\tau} = 0$ $\rho_{\mu\kappa} = \rho_{\tau\nu} = -1$	bi-orthogonal

# Correlation Coefficient of Elementary Signals

Constellation	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{d_\mu d_\nu^*\}}{ d_\mu  \cdot  d_\nu }$	Correlation
	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{j \cdot j\}}{1 \cdot 1} = -1$	antipodal
	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{(-1) \cdot 1\}}{1 \cdot 1} = -1$	antipodal
	$\rho_{\mu\nu} = \frac{\operatorname{Re}\{e^{j\frac{\pi}{4}} \cdot e^{-j\frac{5\pi}{4}}\}}{1 \cdot 1} = -1$	antipodal
	$\rho_{\mu\tau} = \rho_{\mu\kappa} = \rho_{\nu\kappa} = \rho_{\tau\nu} = 0$ $\rho_{\mu\nu} = \rho_{\kappa\tau} = -1$	bi-orthogonal

# Correlation of Elementary Signals (1)

A linear digital modulation scheme is defined by a signal space constellation of  $M$  constellation points  $d$ . A sequence of  $b = \log_2 M$  information bits is uniquely mapped to a particular constellation point  $d_\mu$ . The receiver observes distorted versions of the transmit signal and estimates the constellation point which has most likely been transmitted. Consequently, the signals which correspond to different transmit symbols  $d_\mu$  should be discriminable as good as possible. We denote the transmit signal

$$s_\mu(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB,\mu}(t) e^{j2\pi f_c t} \right\}, \quad \mu \in \{0, \dots, M-1\},$$

which corresponds to transmitting only a single symbol  $d_\mu$  as *elementary signal*. As there are  $M$  possible transmit symbols  $d_\mu$ , we have  $M$  distinct elementary signals  $s_\mu(t)$  for an  $M$ -ary modulation scheme. The complex equivalent baseband signal corresponding to the real bandpass transmit signal  $s_\mu(t)$  is given by

$$s_{BB,\mu}(t) = d_\mu g_t(t).$$

# Correlation of Elementary Signals (2)

The correlation coefficient

$$\rho_{\mu\nu} = \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_{\mu}^2(t) dt \cdot \int_{-\infty}^{\infty} s_{\nu}^2(t) dt}}$$

can be used as a quantitative measure for the discriminability of two elementary signals  $s_{\mu}(t)$  and  $s_{\nu}(t)$ . The correlation coefficient of the real elementary signals is normalized such that it takes values in the range

$$-1 \leq \rho_{\mu\nu} \leq +1.$$

The elementary signals are

- fully correlated, if  $\rho_{\mu\nu} = +1$
- orthogonal, if  $\rho_{\mu\nu} = 0$
- antipodal, if  $\rho_{\mu\nu} = -1$ .

For optimum separability of elementary signals, it is desirable to have antipodal elementary signals as this indicates maximum “dissimilarity.”

# Correlation of Elementary Signals (3)

It can be shown that the correlation coefficient for elementary signals of a linear modulation scheme is completely determined by the constellation points  $d_\mu$  and  $d_\nu$ :

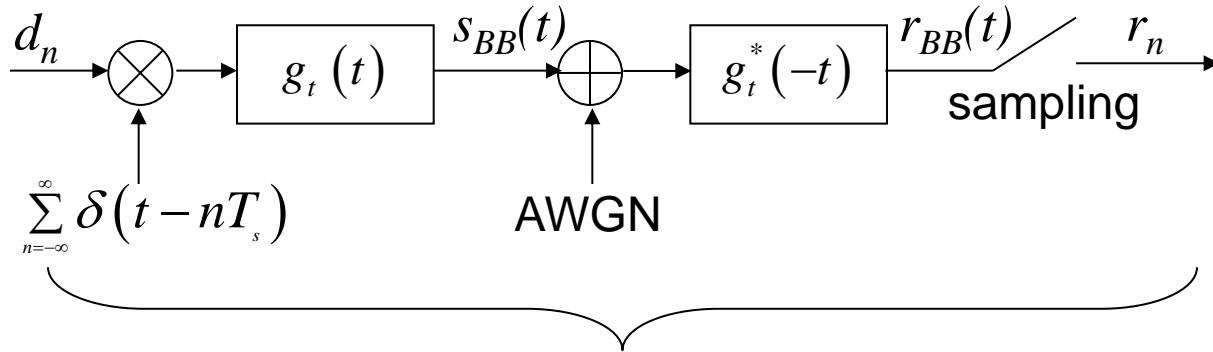
$$\rho_{\mu\nu} = \frac{\operatorname{Re}\{d_\mu d_\nu^*\}}{|d_\mu| \cdot |d_\nu|}.$$

On-off keying (OOK) is an example for orthogonal elementary signals, while BPSK achieves antipodal elementary signals. Consequently, BPSK can achieve lower bit error probability in noisy channels. However, from a practical perspective, it is sometimes advantageous to have orthogonal elementary signals as this can reduce the receiver complexity.

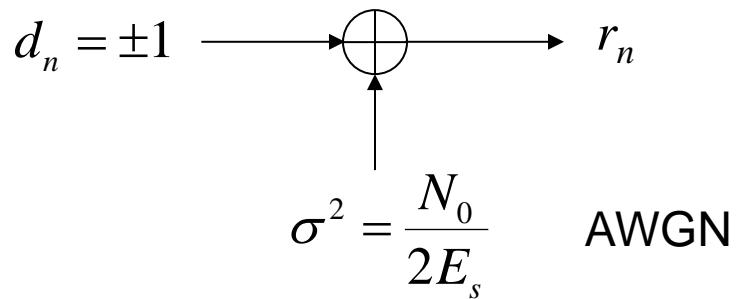
An orthogonal binary signal space constellation can be extended by adding a second pair of orthogonal constellation points which is obtained by negating the original pair. The resulting constellation is called *bi-orthogonal*. QPSK is an example for a bi-orthogonal modulation scheme.

# Bit Error Probability of BPSK in AWGN Channel

Equivalent baseband channel model:



Discrete time channel model:

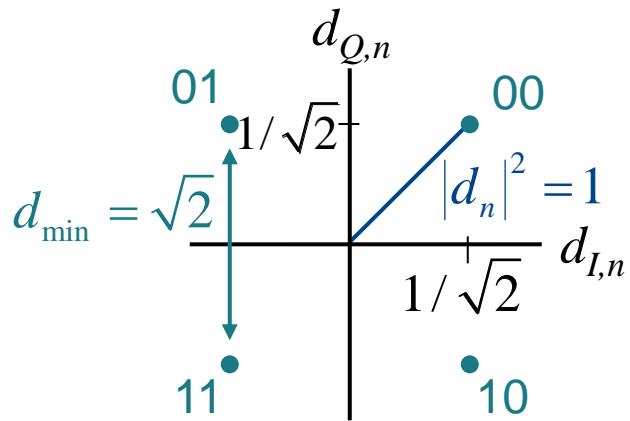


$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_s}{N_0}} \right) = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

$$E_s = E_b$$

# Bit Error Probability of QPSK with Gray Mapping in AWGN Channel

## Gray Mapping



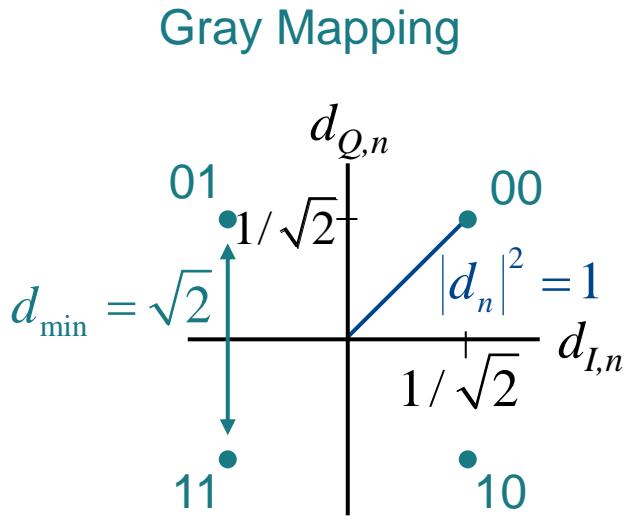
2 orthogonal BPSK signals with energy per BPSK symbol  $E_b = E_s/2$ .

$E_s$ : Energy per QPSK symbol  
 $E_b = E_s/2$ : Energy per bit

$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_s}{2N_0}} \right) = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

$$E_s = 2E_b$$

# Bit Error Probability of QPSK with Gray Mapping in AWGN Channel



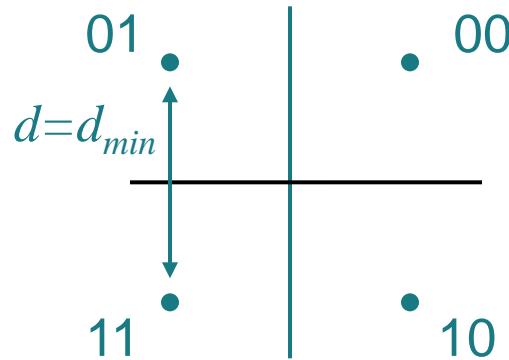
QPSK can be viewed as two orthogonal BPSK signals: The first bit determines the quadrature component, the second bit determines the inphase component. Since the transmit power has to be split between both quadrature components, the energy per symbol of each BPSK signal is  $E_b = E_s/2$ . Hence, QPSK requires a 3 dB higher  $E_s/N_0$  than BPSK for the same bit error probability. A comparison in terms of per information bit SNR  $E_b/N_0$  takes into account that 2 bits can be transmitted per symbol. Therefore, QPSK requires only half the bandwidth of BPSK for the same bit rate which means that only half the noise power falls within the pass band. Consequently, the fair comparison in terms of  $E_b/N_0$  yields the same performance for BPSK and QPSK.

$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_s}{2N_0}} \right) = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

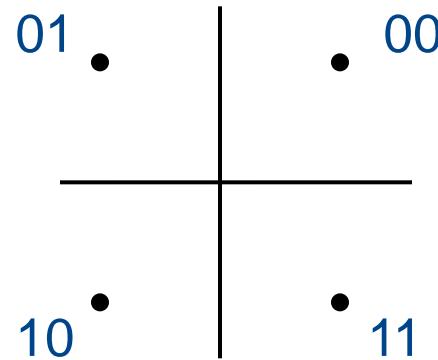
$$E_s = 2E_b$$

# Gray Mapping

Gray Mapping

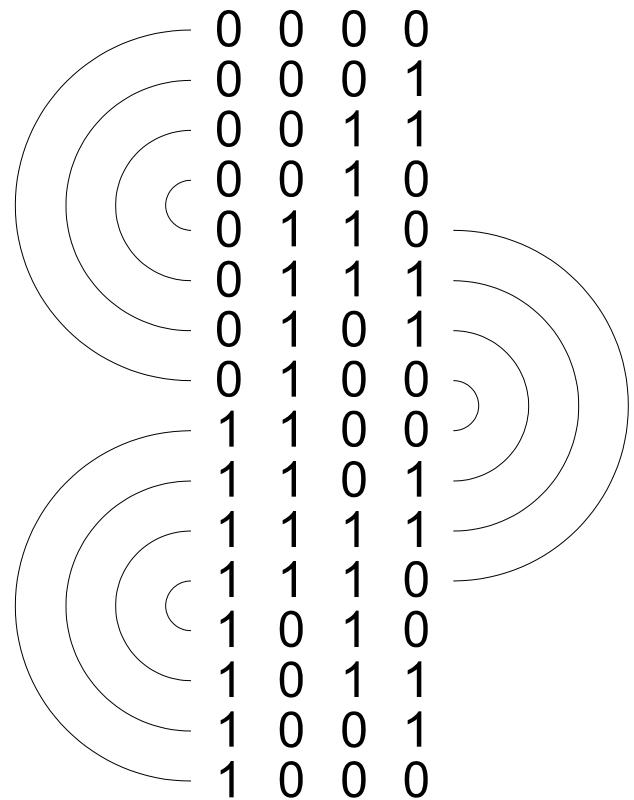
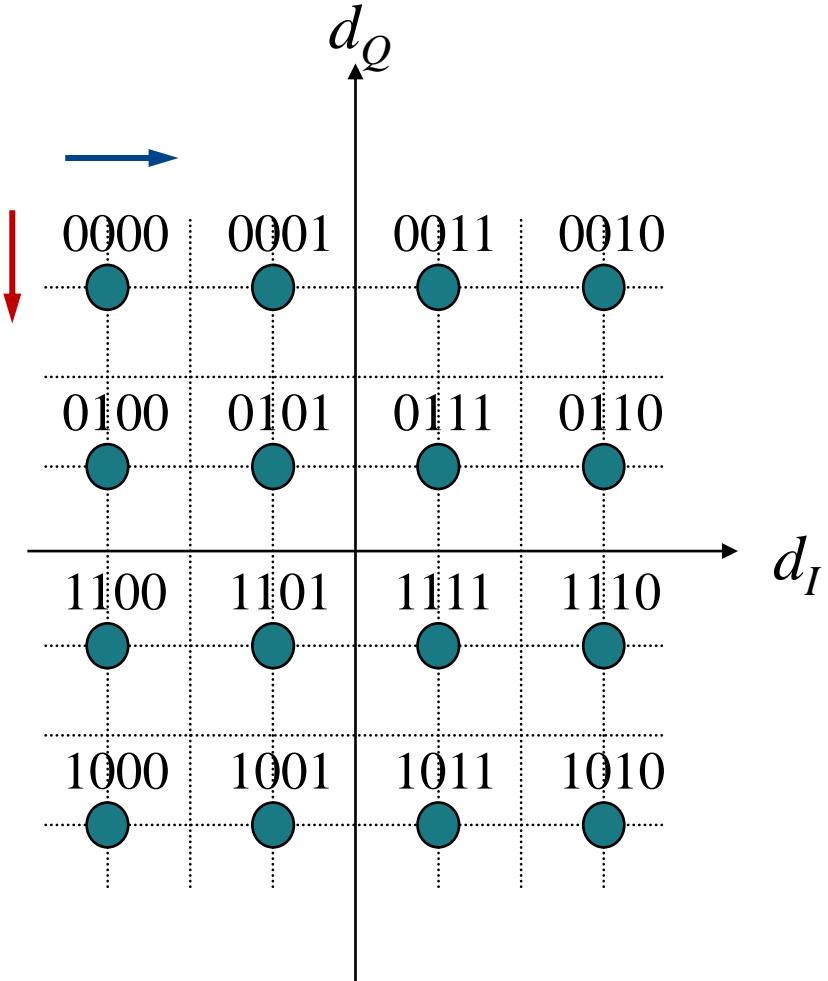


Natural Mapping

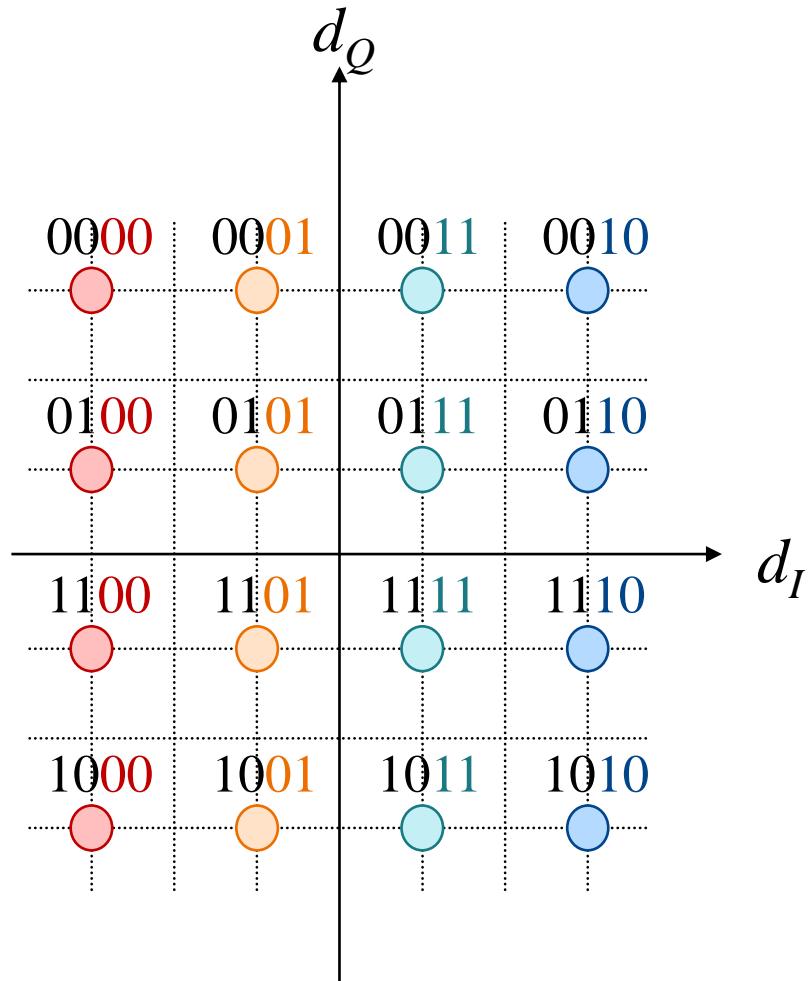


With Gray-Mapping, constellation points with minimum Euclidean distance  $d_{min}$  differ in only 1 bit. The most likely symbol error at reasonably high SNR is to confuse constellation points at minimum Euclidean distance. For this most likely error event, we have only 1 bit error per symbol error. This minimizes the average bit error probability. In contrast, for natural mapping error events at minimum Euclidean distance with 2 bit errors per symbol error can occur.

# Gray Mapping for 16 QAM (1)

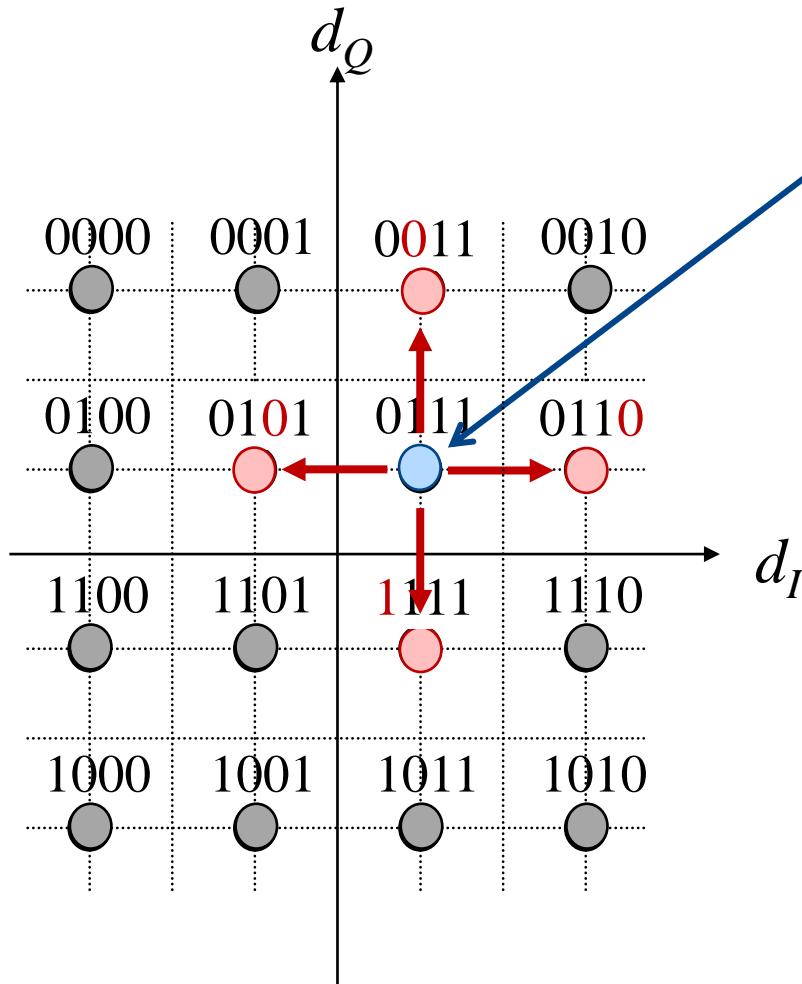


# Gray Mapping for 16 QAM (2)



- First two bits indicate level in quadrature component
- Last two bits indicate level in inphase component

# Symbol Error Probability and Bit Error Probability with Gray Mapping



most likely symbol detection errors at medium or high SNR

→ symbol error probability  $P_s$

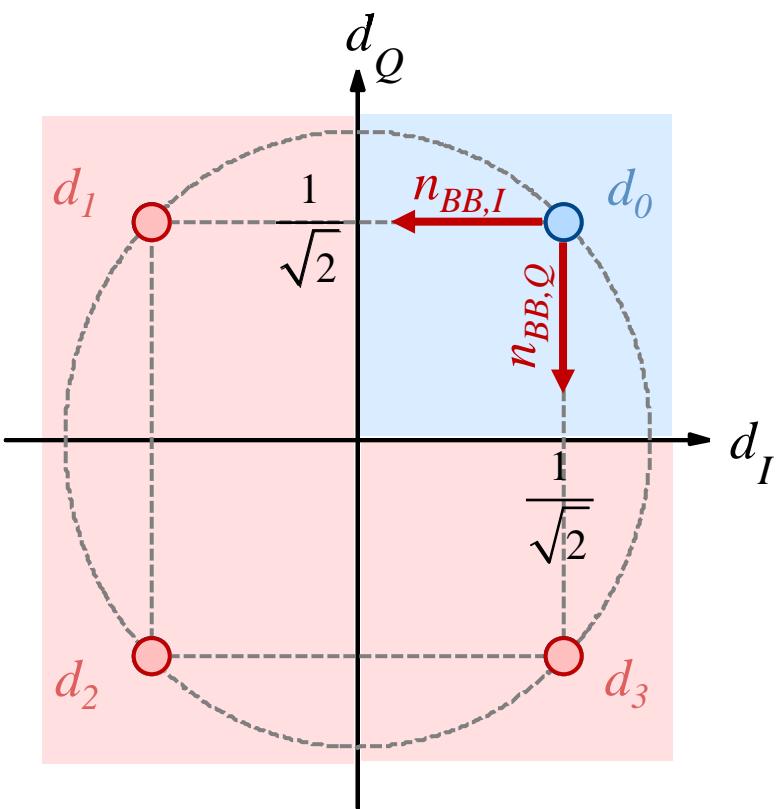
→ 1 bit error per symbol error

→  $\log_2(M) = 4$  bit per symbol

Bit error probability with Gray mapping at medium to high SNR:

$$P_b \approx \frac{1}{\log_2 M} P_s$$

# Symbol Error Probability of QPSK in AWGN Channel



$$P_s = \sum_{\mu=0}^{M-1} P(\hat{d} \neq d_\mu | d_\mu) P(d_\mu)$$

detection error probability is the same for all constellation points  $d_\mu$

→ needs to be determined for only one constellation point  $d_\mu$

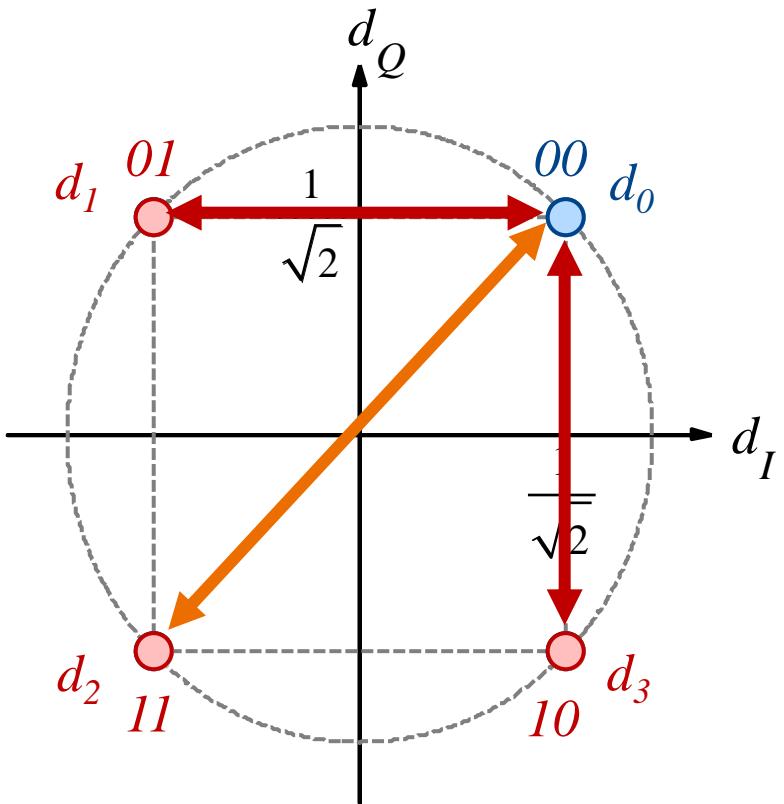
$$\Rightarrow P_s = P(\hat{d} \neq d_\mu | d_\mu)$$

$$P_s = 1 - \bar{P}_s$$

$$\bar{P}_s = P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) P\left(n_{BB,Q} > -\frac{1}{\sqrt{2}}\right)$$

$$P_s = \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) - \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \right]^2$$

# Euclidean Distance and Hamming Distance



## Euclidean distance:

Geometric distance

$$d_E(d_0, d_1) = 2 \cdot \frac{1}{\sqrt{2}} = \sqrt{2} = d_E(d_0, d_3)$$

$$d_E(d_0, d_2) = 2$$

→ important for symbol error probability  $P_s$

## Hamming distance:

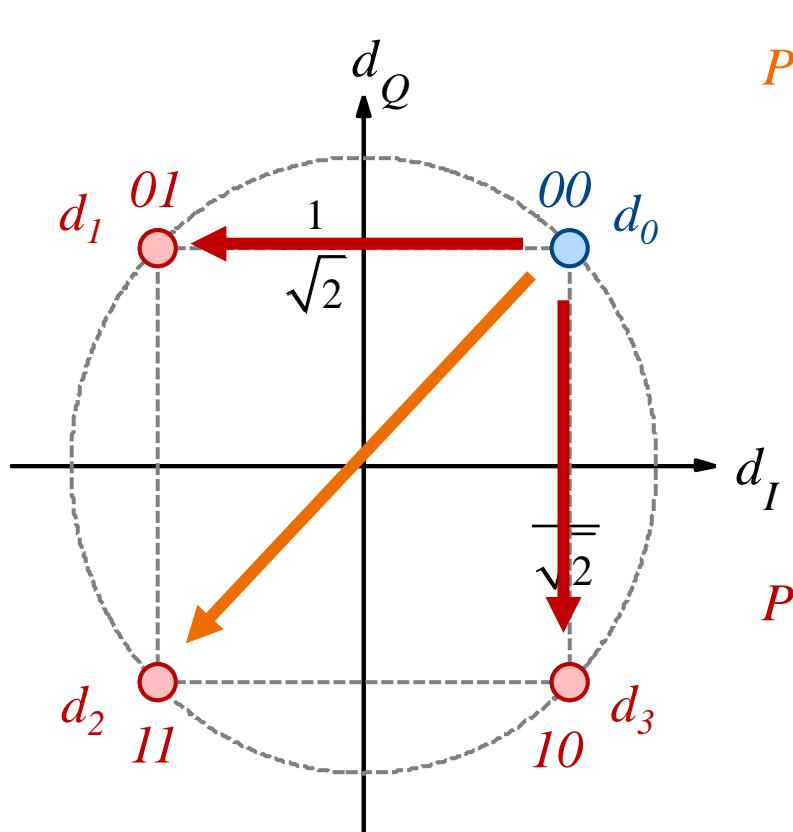
Number of bits, in which two symbols differ:

$$d_H(d_0, d_1) = 1 = d_H(d_0, d_3)$$

$$d_H(d_0, d_2) = 2$$

→ important for bit error probability  $P_b$

# Individual Symbol Error Probabilities of QPSK in AWGN Channel



$$P(d_0 \rightarrow d_2) = P\left(n_{BB,I} < -\frac{1}{\sqrt{2}}\right)P\left(n_{BB,Q} < -\frac{1}{\sqrt{2}}\right)$$

$$= \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \right]^2$$

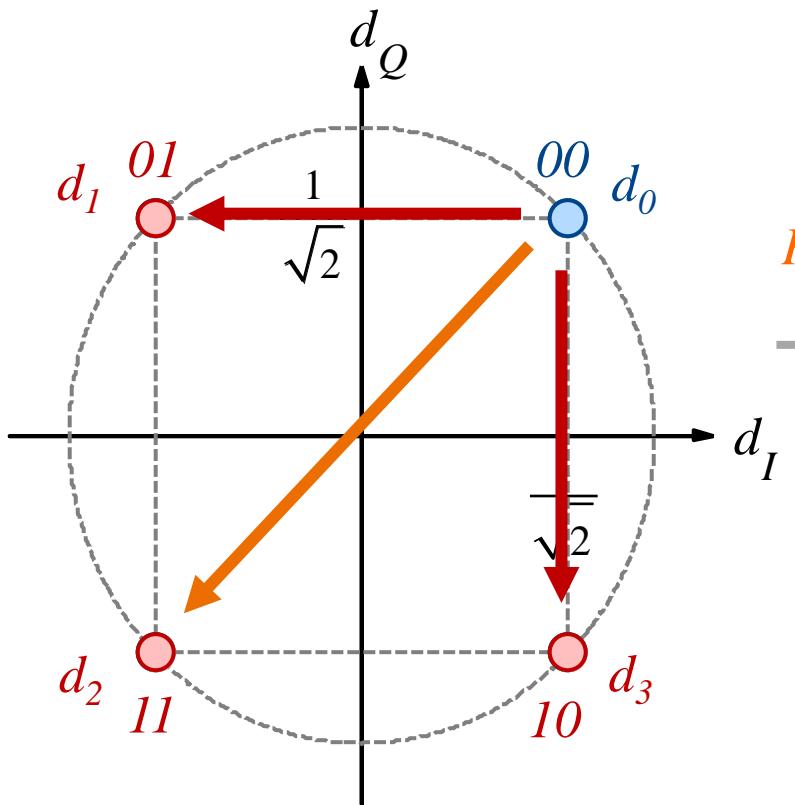
$$P(d_0 \rightarrow d_1) = P\left(n_{BB,I} < -\frac{1}{\sqrt{2}}\right)P\left(n_{BB,Q} > -\frac{1}{\sqrt{2}}\right)$$

$$= \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \left[ 1 - \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \right]$$

$$= \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) - \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \right]^2$$

$$= P(d_0 \rightarrow d_3)$$

# Bit Error Probability of QPSK in AWGN Channel (1)



$$P(d_0 \rightarrow d_1) = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) - \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \right]^2$$

$$= P(d_0 \rightarrow d_3)$$

$$P(d_0 \rightarrow d_2) = \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \right]^2$$

1 out of 2 bits is wrong

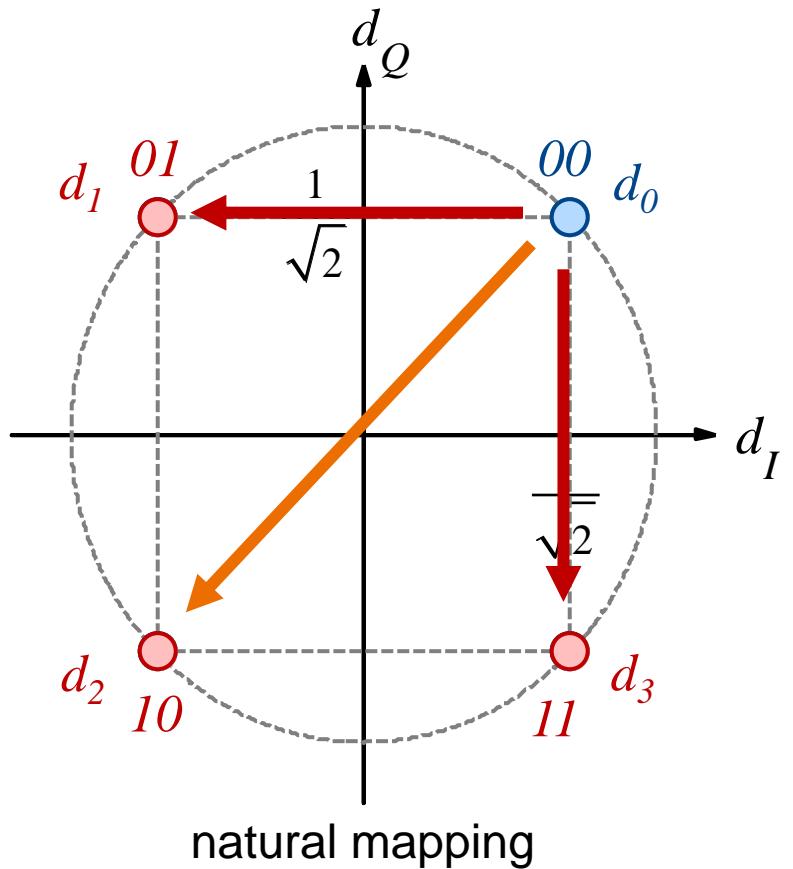
$$d_H(d_0, d_1) = 1$$

$$d_H(d_0, d_2) = 2$$

$$P_b = \left| \frac{1}{2} \right| P(d_0 \rightarrow d_1) + \left| \frac{2}{2} \right| P(d_0 \rightarrow d_2)$$

$$+ \left| \frac{1}{2} \right| P(d_0 \rightarrow d_3)$$

# Bit Error Probability of QPSK in AWGN Channel (2)



in general:

1 out of 2 bits is wrong

$$d_H(d_0, d_1) = 1$$

$$d_H(d_0, d_3) = 2$$

$$P_b = \frac{1}{2} P(d_0 \rightarrow d_1) + \frac{1}{2} P(d_0 \rightarrow d_3)$$

$$+ \frac{1}{2} P(d_0 \rightarrow d_2)$$

$$P_b = \sum_{\mu=0}^{M-1} P(d_\mu) \sum_{\nu=0}^{M-1} \frac{d_H(d_\mu, d_\nu)}{\log_2 M} P(d_\mu \rightarrow d_\nu)$$

# Symbol Error Probability of QPSK in AWGN Channel-Derivation (1)

$$P_s = 1 - \bar{P}_s$$

$$\bar{P}_s = P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) P\left(n_{BB,Q} > -\frac{1}{\sqrt{2}}\right)$$

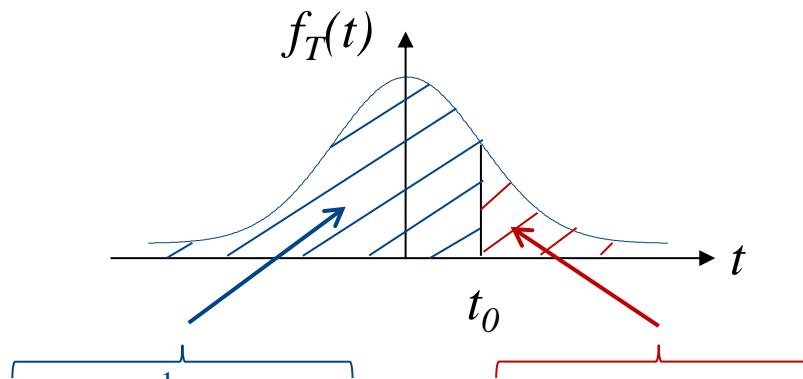
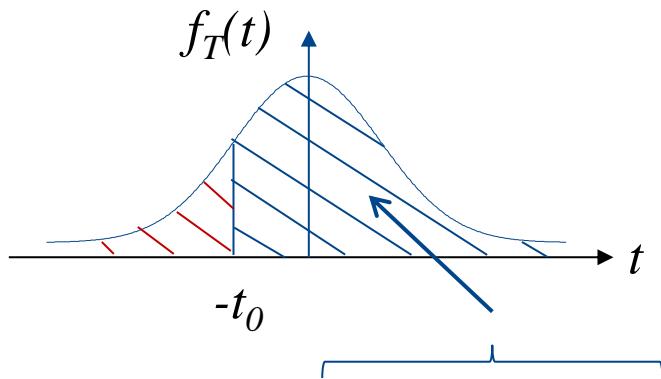
$$P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) = \int_{-\frac{1}{\sqrt{2}}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_{N_{BB,I}}} e^{-\frac{n_{BB,I}^2}{2\sigma_{N_{BB,I}}^2}} dn_{BB,I}$$

substitution:

$$t = \frac{n_{BB,I}}{\sqrt{2}\sigma_{N_{BB,I}}} \quad \Rightarrow \quad \frac{dt}{dn_{BB,I}} = \frac{1}{\sqrt{2}\sigma_{N_{BB,I}}}$$

$$P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) = \int_{-\frac{1}{\sqrt{2}\sigma_{N_{BB,I}}}}^{\infty} \frac{\sqrt{2}\sigma_{N_{BB,I}}}{\sqrt{2\pi}\sigma_{N_{BB,I}}} e^{-t^2} dt = \frac{1}{\sqrt{\pi}} \int_{-\frac{1}{\sqrt{2}\sigma_{N_{BB,I}}}}^{\infty} e^{-t^2} dt$$

# Symbol Error Probability of QPSK in AWGN Channel-Derivation (2)



$$P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) = \frac{1}{\sqrt{\pi}} \int_{-\frac{1}{\sqrt{2}} / \frac{2\sigma_{N_{BB,I}}}{\sqrt{\pi}}}^{\infty} e^{-t^2} dt = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\frac{1}{2\sigma_{N_{BB,I}}}} e^{-t^2} dt = 1 - \frac{1}{\sqrt{\pi}} \int_{\frac{1}{2\sigma_{N_{BB,I}}}}^{\infty} e^{-t^2} dt$$

$$\sigma_{N_{BB,I}}^2 = \frac{N_0}{2E_s}$$

$$P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right) = 1 - \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right)$$

$$\frac{1}{2} \operatorname{erfc}\left(\frac{1}{2\sigma_{N_{BB,I}}}\right)$$

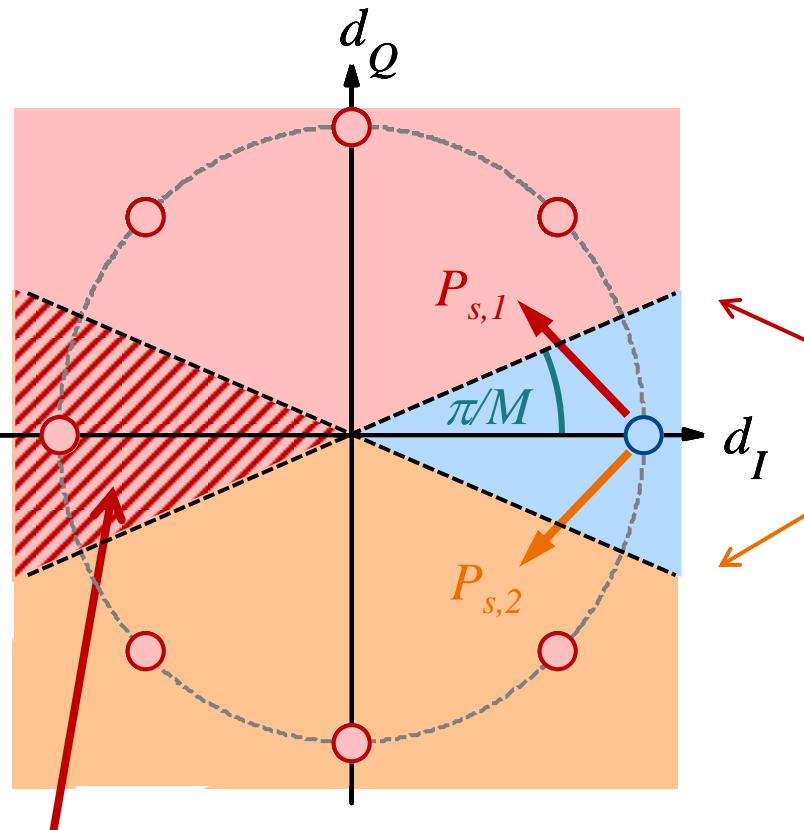
# Symbol Error Probability of QPSK in AWGN Channel-Derivation (3)

$$\bar{P}_s = P\left(n_{BB,I} > -\frac{1}{\sqrt{2}}\right)P\left(n_{BB,Q} > -\frac{1}{\sqrt{2}}\right) = \left[1 - \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right)\right]^2$$

$$= 1 - \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) + \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \right]^2$$

$$P_s = 1 - \bar{P}_s = \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) - \frac{1}{4} \left[ \operatorname{erfc}\left(\sqrt{\frac{E_s}{2N_0}}\right) \right]^2$$

# Approximate Symbol Error Probability of $M$ -PSK in AWGN Channel



$$P_s \approx 2P_{s,1}$$

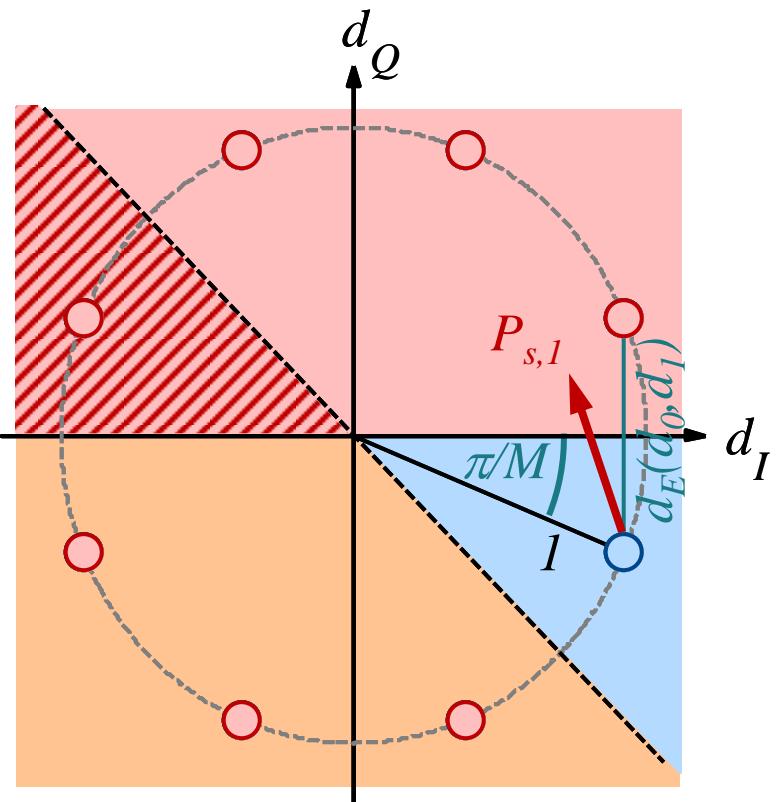


Crossing upper limit and crossing lower limit are error events with equal probability.

Shaded area counted twice  
→ error probability overestimated!

At high SNR, the probability for a received sample in shaded area is low.  
→ good approximation

# Approximate Symbol Error Probability of $M$ -PSK in AWGN Channel



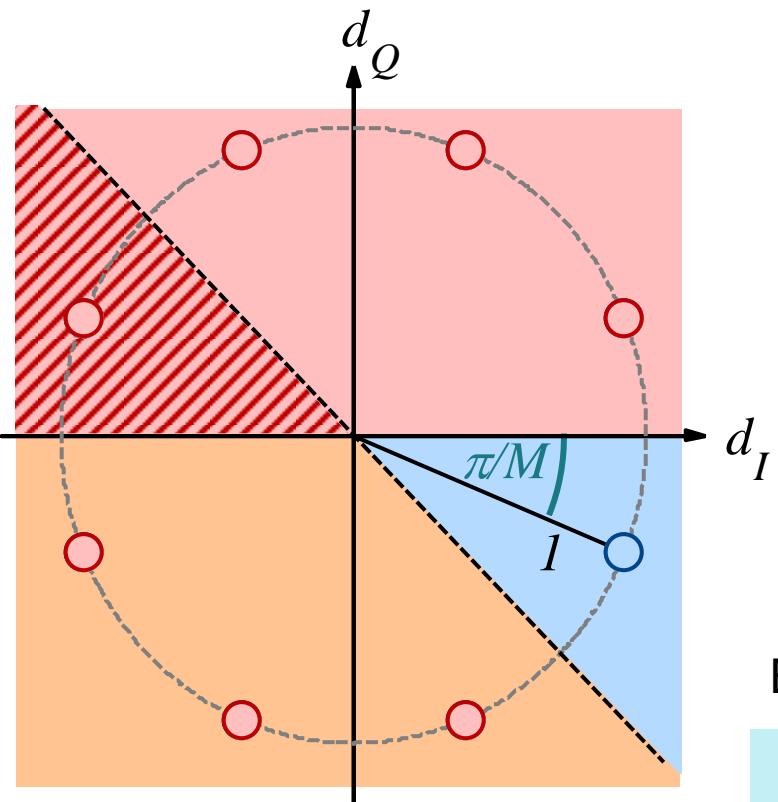
$$P_s \approx 2 \mathbf{P}_{s,1} = 2P\left(n_{BB,Q} \geq \frac{d_E(d_0, d_1)}{2}\right)$$

$$\frac{d_E(d_0, d_1)}{2} = \sin\left(\frac{\pi}{M}\right)$$

$$\sigma_{N_{BB,Q}}^2 = \frac{N_0}{2E_s}$$

$$P_s \approx \operatorname{erfc}\left(\sqrt{\frac{E_s}{N_0}} \cdot \sin\left(\frac{\pi}{M}\right)\right)$$

# Approximate Bit Error Probability of $M$ -PSK in AWGN Channel



Symbol error probability:

$$P_s \approx \operatorname{erfc} \left( \sqrt{\frac{E_s}{N_0}} \cdot \sin \left( \frac{\pi}{M} \right) \right)$$

$$E_s = \log_2(M) E_b$$

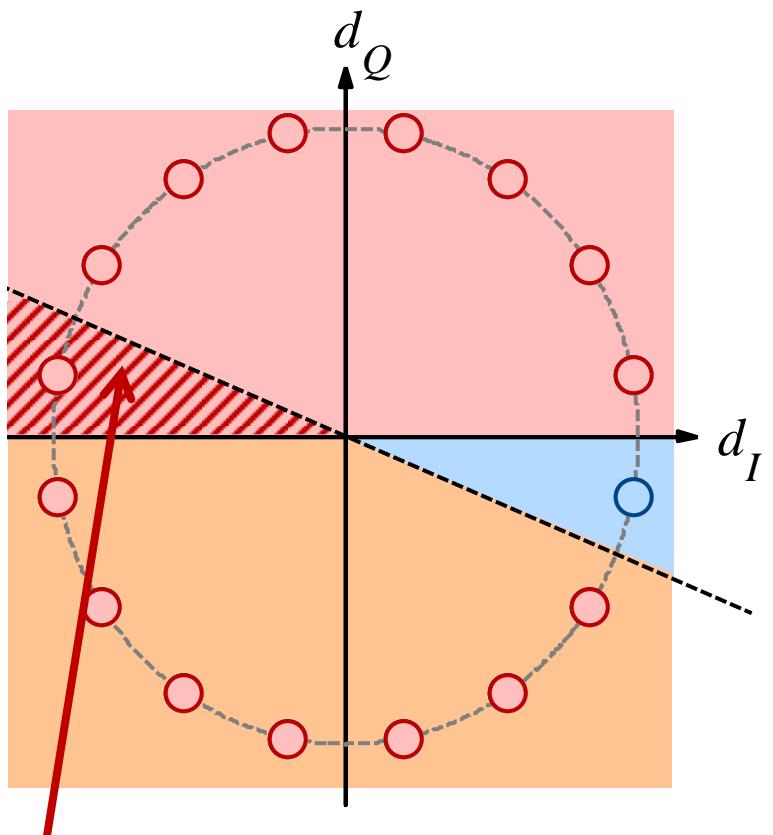
↓  
energy per  $M$ -PSK symbol  
↑  
energy per bit

Bit error probability with Gray mapping:

$$P_b \approx \frac{1}{\log_2(M)} P_s$$

$$\approx \frac{1}{\log_2(M)} \operatorname{erfc} \left( \sqrt{\frac{\log_2(M) E_b}{N_0}} \cdot \sin \left( \frac{\pi}{M} \right) \right)$$

# Symbol Error Probability Approximation Error for $M$ -PSK in AWGN Channel



Symbol error probability:

$$P_s \approx \text{erfc} \left( \sqrt{\frac{E_s}{N_0}} \cdot \sin \left( \frac{\pi}{M} \right) \right)$$

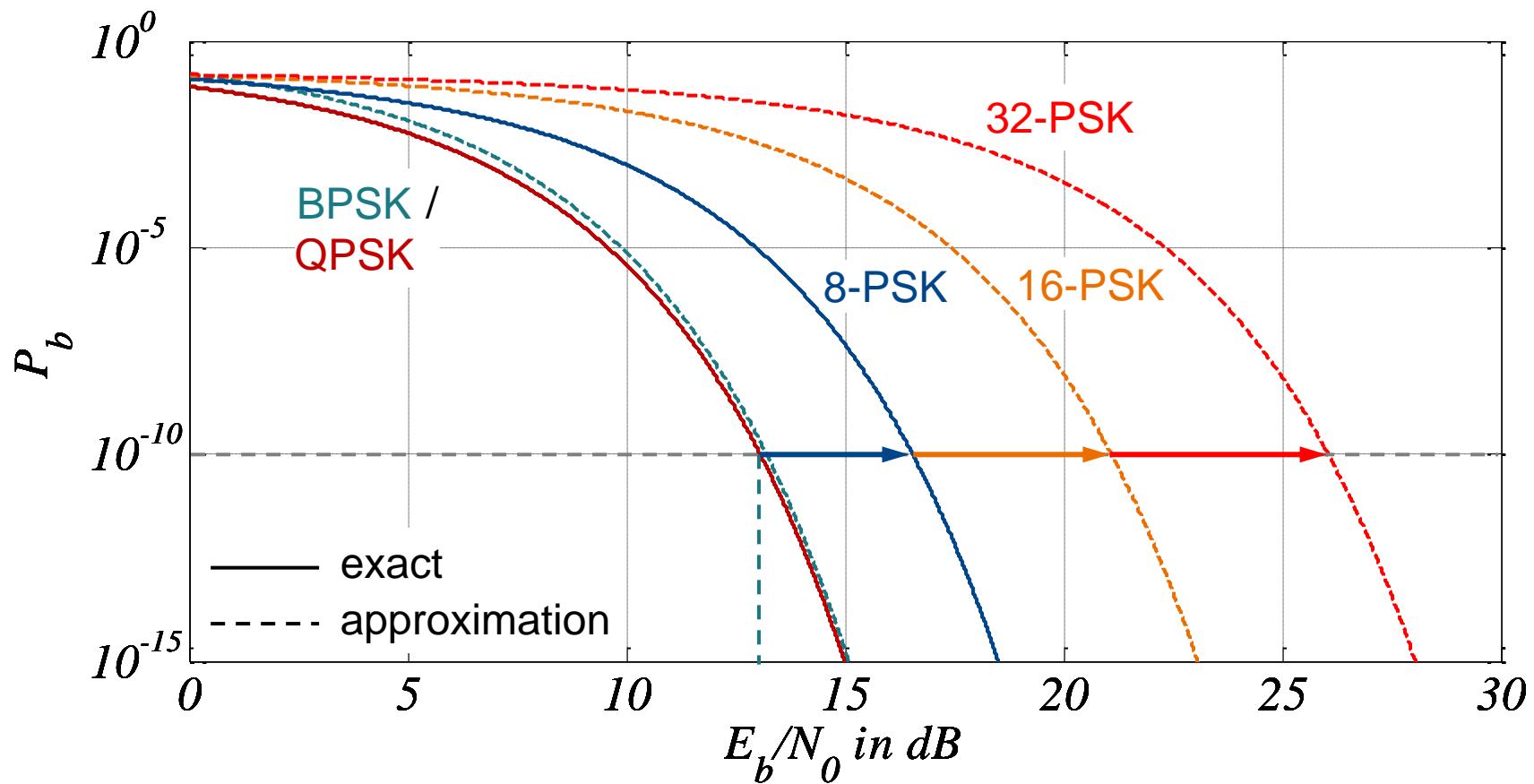
Approximation error decreases  
with increasing PSK order  $M$ .

# Comparison of Exact and Approximate Bit Error Probabilities of $M$ -PSK in AWGN Channel

	$M$	$P_b$ (exact)	$P_b$ (approximation)
BPSK	2	$\frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$	$\operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$
QPSK	4	$\frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$	$\frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$
8-PSK	8	$\frac{1}{3}\operatorname{erfc}\left(\sqrt{\frac{3E_b}{N_0}} \cdot \sin\left(\frac{\pi}{8}\right)\right)$ $\cdot \left[1 - \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{3E_b}{N_0}} \cdot \sin\left(\frac{3\pi}{8}\right)\right)\right]$ $+ \frac{1}{3}\operatorname{erfc}\left(\sqrt{\frac{3E_b}{N_0}} \cdot \sin\left(\frac{3\pi}{8}\right)\right)$	$\frac{1}{3}\operatorname{erfc}\left(\sqrt{\frac{3E_b}{N_0}} \cdot \sin\left(\frac{\pi}{8}\right)\right)$

Exact solutions exist only for  $M \leq 8$ .

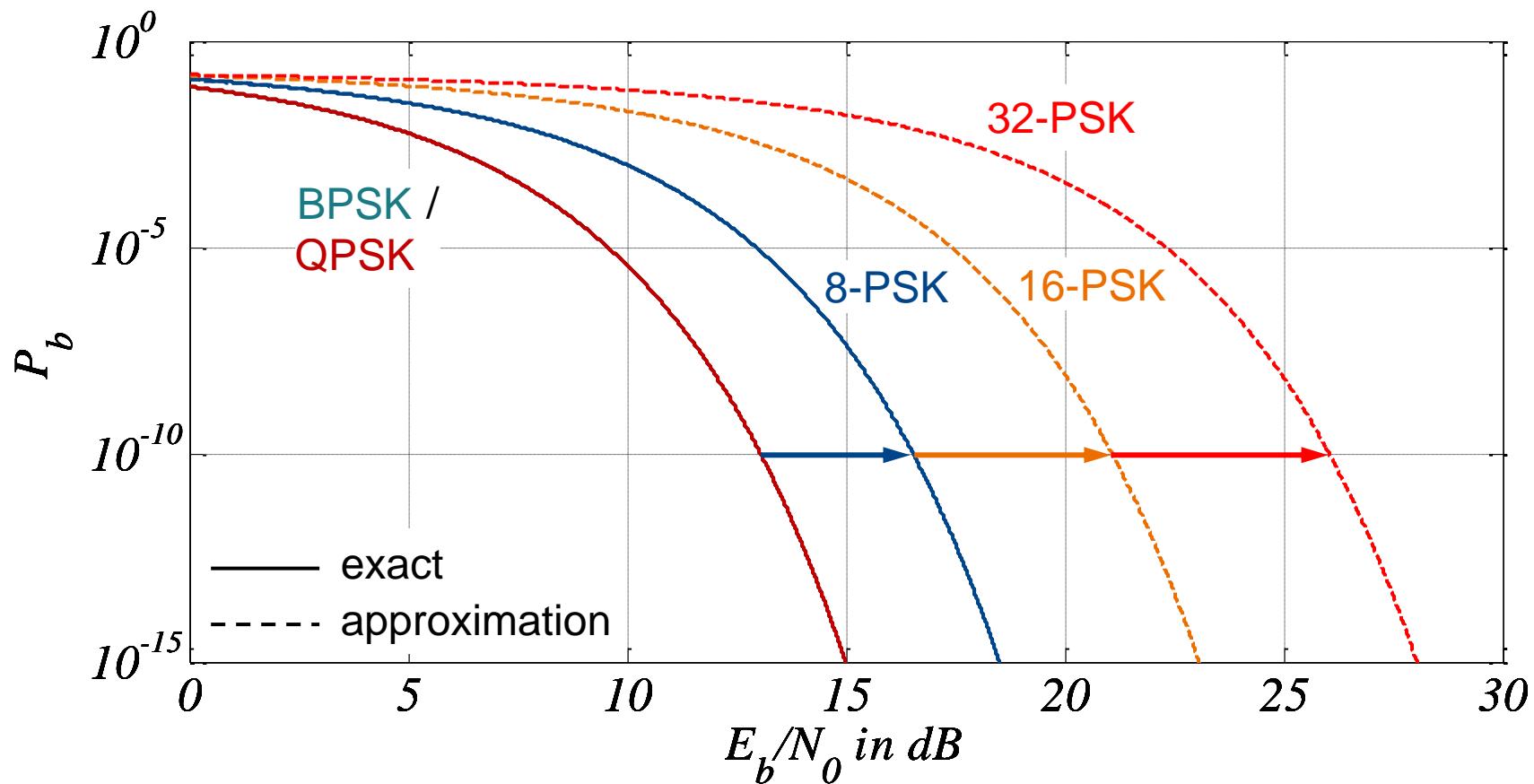
# Approximate Bit Error Probability of $M$ -PSK in AWGN Channel (1)



Required  $E_b/N_0$  for  $P_b=10^{-10}$ :

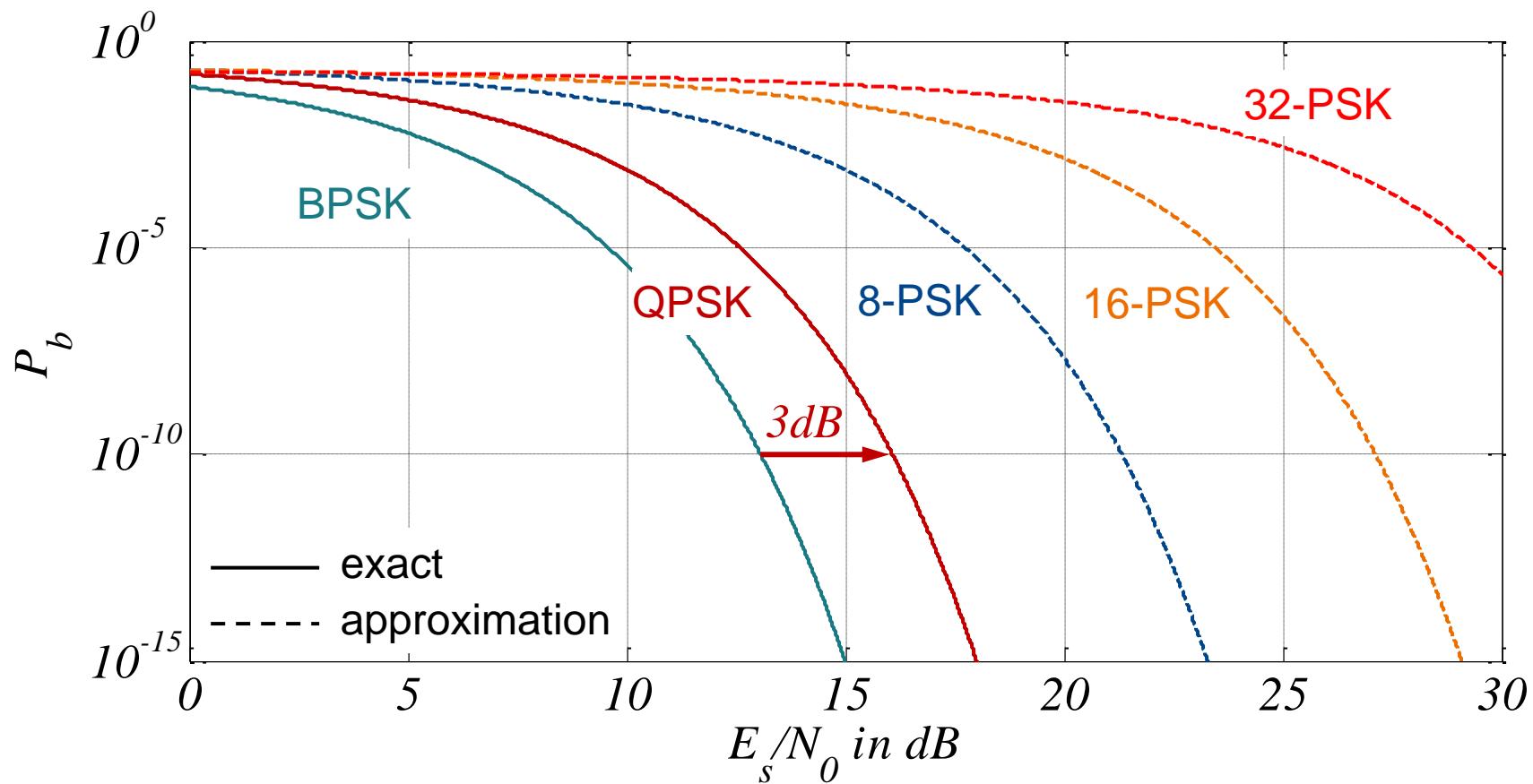
BPSK / QPSK	$+3.5\text{dB}$	8-PSK	$+4.5\text{dB}$	16-PSK	$+5\text{dB}$	32-PSK
13.1 dB		16.6 dB		21.1 dB		26.1 dB

# Approximate Bit Error Probability of $M$ -PSK in AWGN Channel (2)



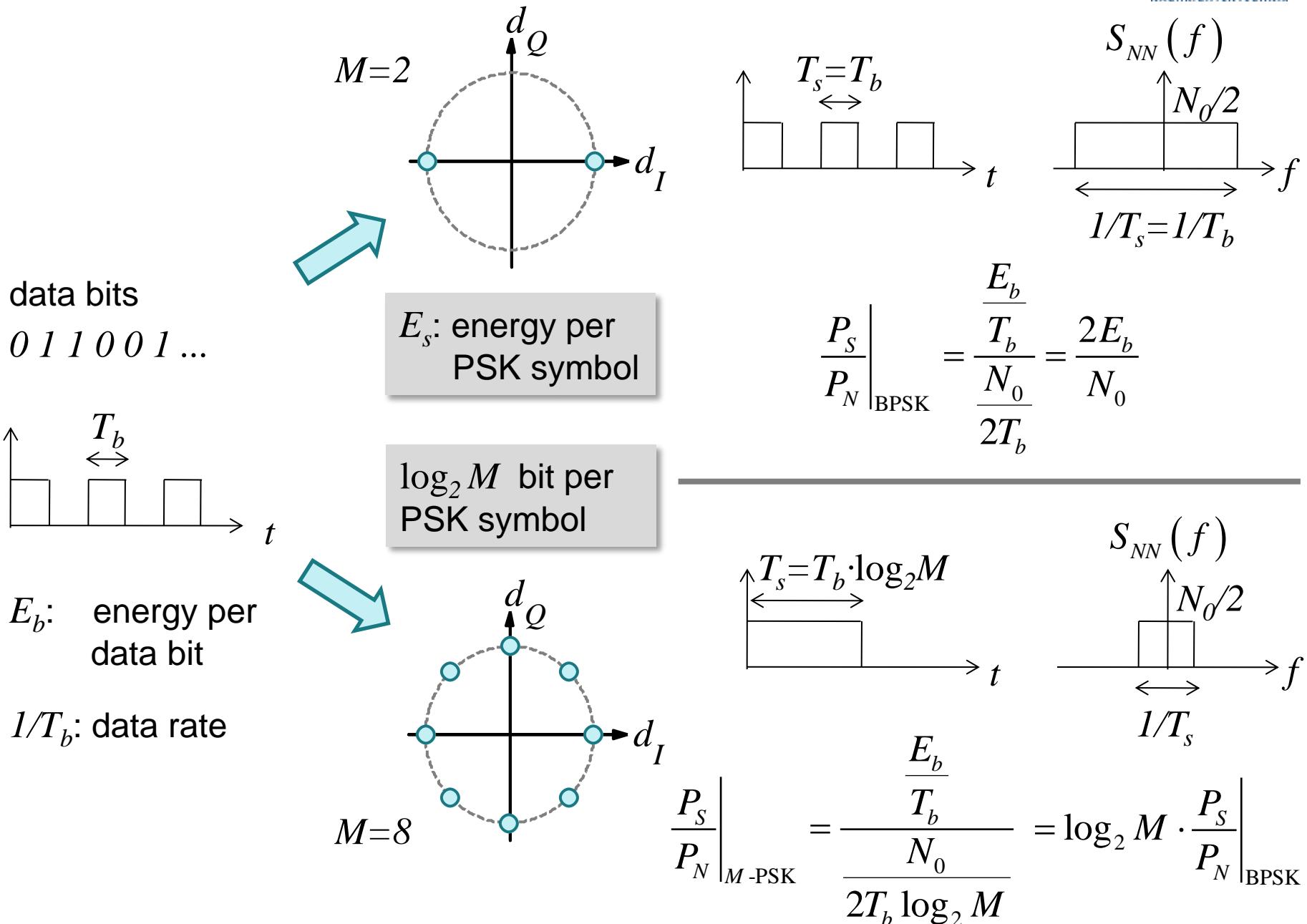
Higher order modulation achieves higher bandwidth efficiency on the expense of a higher required SNR for a certain target error probability.

# Approximate Bit Error Probability of $M$ -PSK vs. $E_s/N_0$ in AWGN Channel



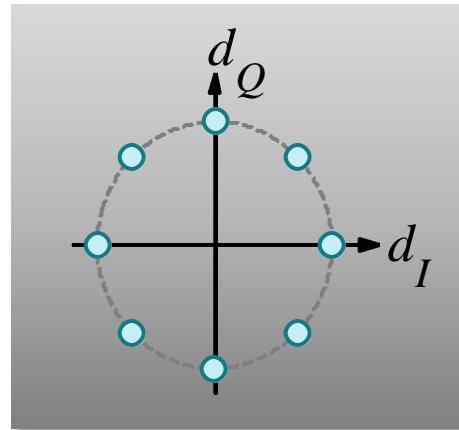
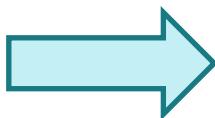
$$E_s = \log_2(M)E_b$$

# Fair Comparison of Modulation Schemes: $E_b$ and $E_s$



# Fair Comparison of Modulation Schemes: $E_b$ and $E_s$

info bits



$E_b$ : energy per info bit

$T_b$ : info bit duration

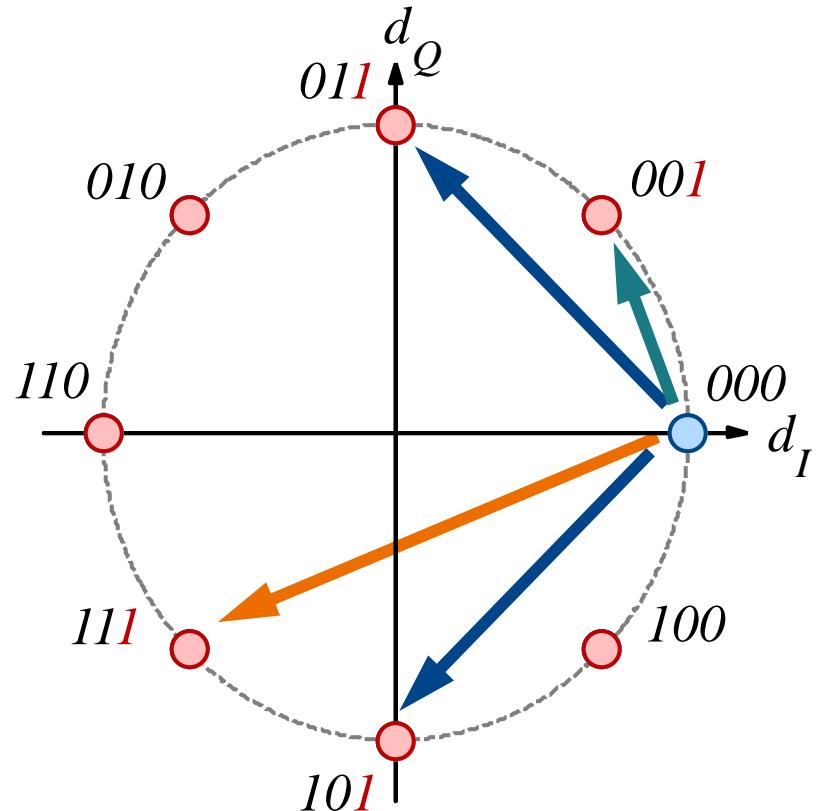
$\frac{1}{T_b}$ : info bit rate

$E_s = \log_2(M)$   $E_b$ : energy per transmit symbol

$T_s = \log_2(M)$   $T_b$  : transmit symbol duration

$\frac{1}{T_s} = \frac{1}{\log_2(M) T_b}$  : Baud rate

# Individual Bit Error Probabilities of 8-PSK

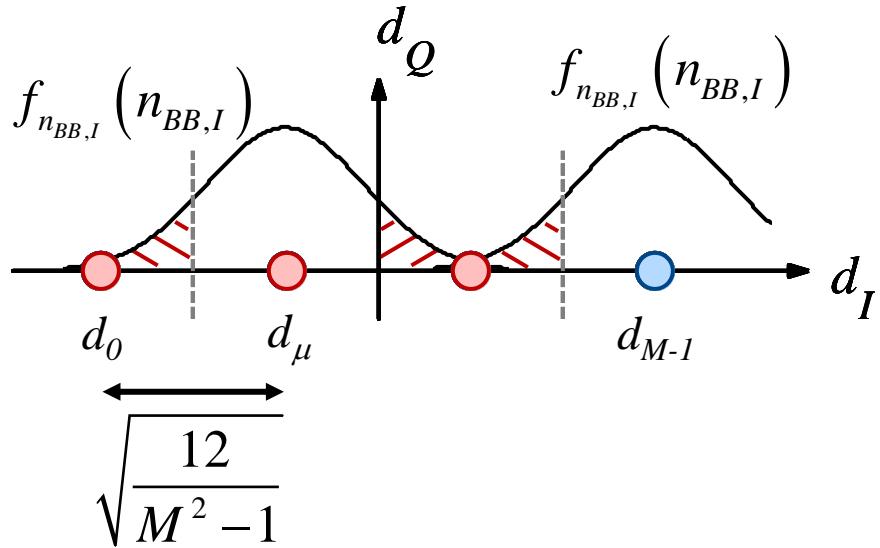


	bit 1	bit 2	bit 3
$P_1$	1	0	1
$P_2$	1	1	2
$P_3$	1	2	1
$P_4$	1	1	0

Occurrences of symbol error probabilities

The bit error probability depends on the position of the bit in the symbol mapping.

# Symbol Error Probability of $M$ -ASK in AWGN Channel



Symbol error probability of inner constellation points:

$$P_{s,i} = 2P\left(n_{BB,I} \geq \sqrt{\frac{3}{M^2 - 1}}\right)$$

$$= \operatorname{erfc}\left(\sqrt{\frac{3}{M^2 - 1} \frac{\bar{E}_s}{N_0}}\right)$$

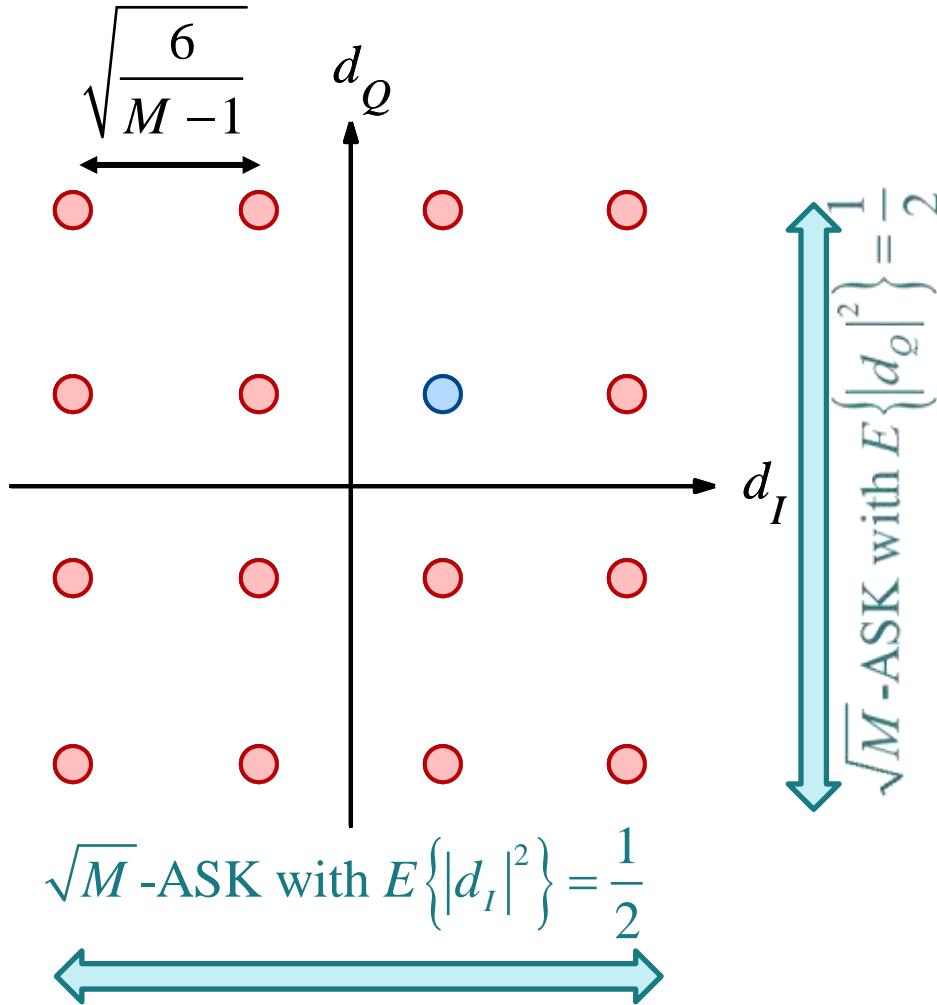
Symbol error probability of outer constellation points:

$$P_{s,o} = \frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{3}{M^2 - 1} \frac{\bar{E}_s}{N_0}}\right)$$

Assumption: All constellation points equally likely, i.e.  $P(d_\mu) = \frac{1}{M}$

$$P_s = \frac{1}{M} [2P_{s,o} + (M-2)P_{s,i}] = \frac{M-1}{M} \operatorname{erfc}\left(\sqrt{\frac{3}{M^2 - 1} \frac{\bar{E}_s}{N_0}}\right)$$

# Symbol Error Probability of $M$ -QAM in AWGN Channel



Probability for correct symbol detection:

$$\bar{P}_s = 1 - P_s$$

Noise in inphase and quadrature component is statistically independent

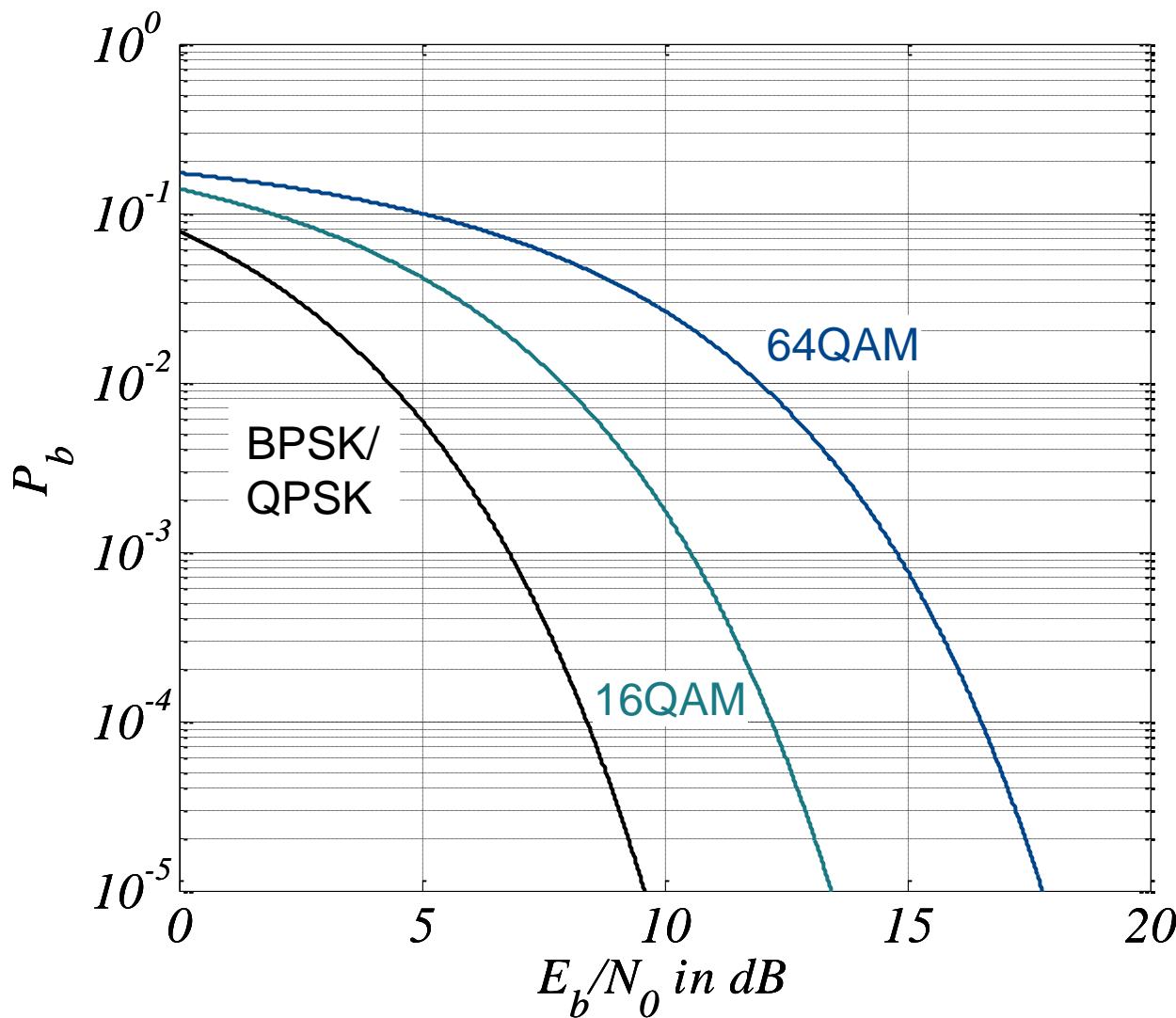
$$\Rightarrow \bar{P}_s = \left( 1 - P_{s, \sqrt{M}\text{-ASK}} \Big| \bar{E}_s \rightarrow \frac{\bar{E}_s}{2} \right)^2$$

$$P_s = 1 - \left( 1 - P_{s, \sqrt{M}\text{-ASK}} \Big| \bar{E}_s \rightarrow \frac{\bar{E}_s}{2} \right)^2$$

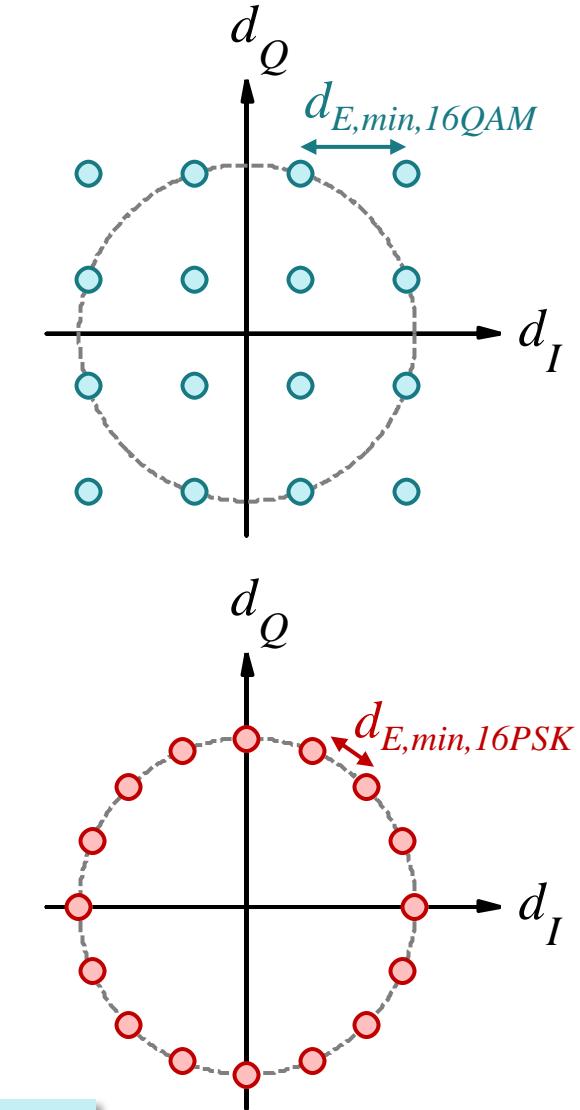
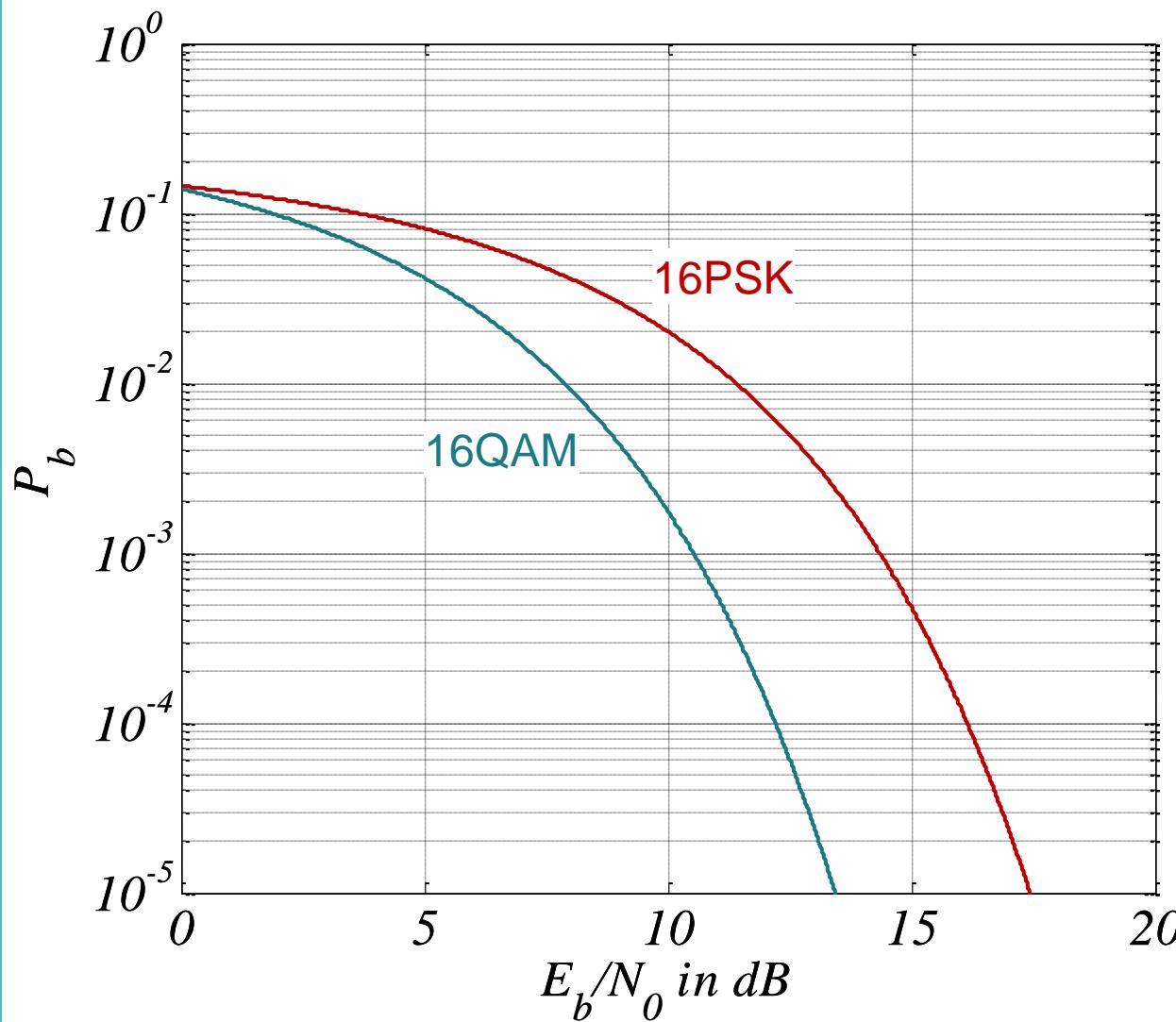
Assumption: All constellation points equally likely, i.e.

$$P(d_\mu) = \frac{1}{M}$$

# Bit Error Probability of $M$ -QAM with Gray Mapping in AWGN Channel

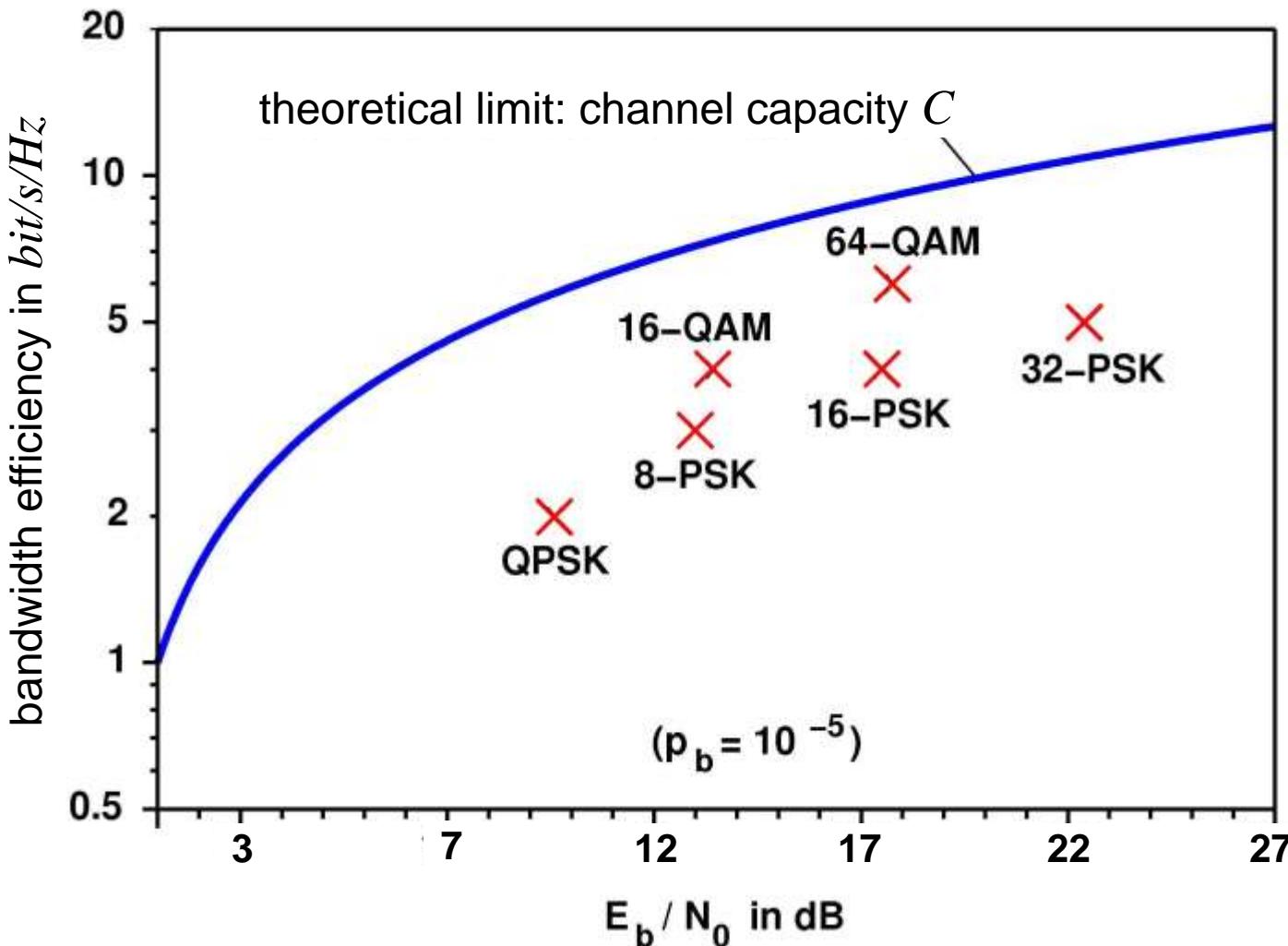


# Bit Error Probability of 16-QAM and 16-PSK with Gray Mapping in AWGN Channel



The minimum Euclidean distance  $d_{E,min}$ , which dominates the error probability, is smaller for 16-PSK compared to 16-QAM.

# Bandwidth Efficiency of Uncoded Linear Digital Modulation



# Hierarchical Modulation/ Multilevel Modulation (1)

The standard digital modulation schemes such as  $M$ -PSK or  $M$ -QAM are characterized by the same minimum Euclidean distances among adjacent constellation points in all directions. This yields the same symbol error probability for each constellation point in symmetric channels (except for the outer constellation points which have fewer neighbours). However, in some applications it is useful to have different Euclidean distances among adjacent constellation points of a higher order modulation scheme. The constellation points can be organized in clusters. The Euclidean distance  $d_1$  among constellation points within the same cluster is smaller than the Euclidean distance  $d_2$  between constellation points which belong to different clusters. Consequently, the symbol error probability between constellation points within a cluster is higher than the probability of deciding for a wrong cluster. This property is exploited in the concept of *hierarchical modulation* (also called *multilevel modulation*).

The mapping of bits to constellation points is done such that a subset of  $b_1$  bits defines the cluster and the remaining  $b_2 = \log_2 M - b_1$  bits define the particular constellation point within the cluster. E.g. in a constellation with  $M=16$  constellation points (i.e. 4 bit per symbol), the first two bits may define the cluster, e.g. the quadrant of the complex domain, whereas the last two bits choose one out of the four constellation points within a cluster.

This allows for detection at different quality levels at the receiver. A high quality receiver with low noise figure and, hence, high SNR may be able to distinguish constellation points at the minimum Euclidean distance  $d_{\min} = d_1$  of the constellation. Therefore, it can detect all  $b$  bits per symbol. A cheaper low quality detector may have a higher noise figure and hence face lower SNR. Consequently, it may not be able to distinguish constellation points at the minimum Euclidean distance  $d_{\min} = d_1$  of the constellation due to too high noise variance. However, it may still be able to detect the cluster to which the transmitted symbol belongs.

# Hierarchical Modulation/ Multilevel Modulation (2)

Therefore, it can at least detect the  $b_1 < \log_2 M$  bits which define the respective cluster. This is an advantage over non-hierarchical modulation, where all bits would be lost in case of too low SNR.

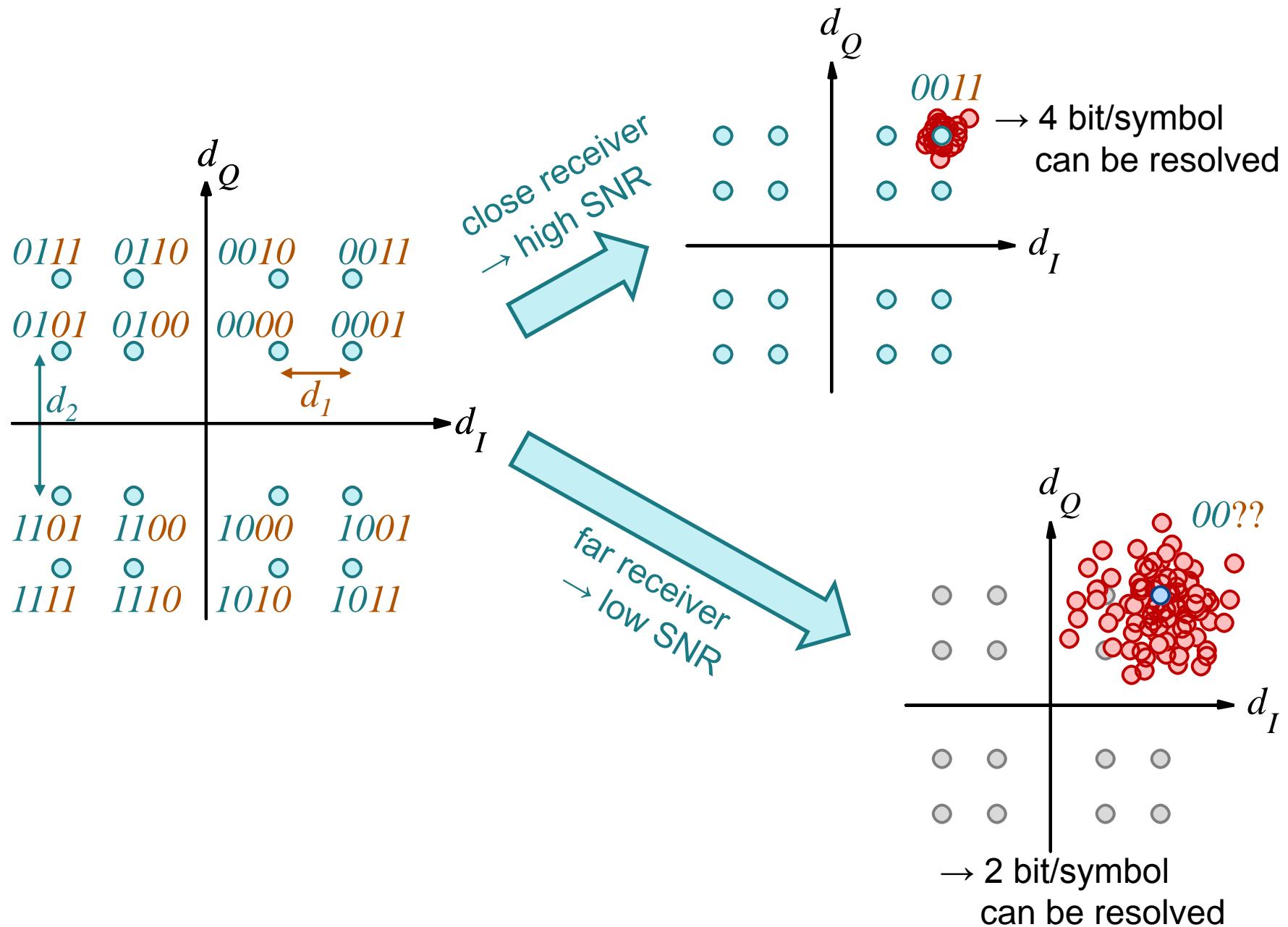
In case of  $M=16$  constellation points and  $M_1=4$  clusters (quadrants), the low end receiver would essentially see a QPSK modulation, whereas the high end receiver sees all  $M=6$  constellation points.

Another application for hierarchical modulation would be a wireless broadcast scenario, where receivers are located at different distance from the transmitter and, hence, face different SNR due to different signal attenuation.

Video broadcast is an application, where hierarchical modulation is used: Some video codecs allow to code the video signal in a hierarchical way such that a subset of the coded bits define a low resolution video which contains the essential information. The remaining bits provide incremental information for a higher resolution and, hence, better video quality. Hierarchical video coding can be combined with hierarchical modulation: The low resolution video data is mapped to the  $b_1$  bits of the modulation scheme, which define the cluster. The remaining bits of the video data are mapped to the  $\log_2 M - b_1$  bits which define the constellation point within a cluster. With such a transmission scheme, a receiver which is located close to the transmitter and faces sufficiently high SNR can receive a high resolution video. A distant receiver with lower SNR will not be able to receive high resolution video quality but may still be able to resolve a low resolution video quality.

The effect of allowing for different reception quality rather than a hard work/fail behaviour is called *graceful degradation*.

# Hierarchical Modulation/Multilevel Modulation



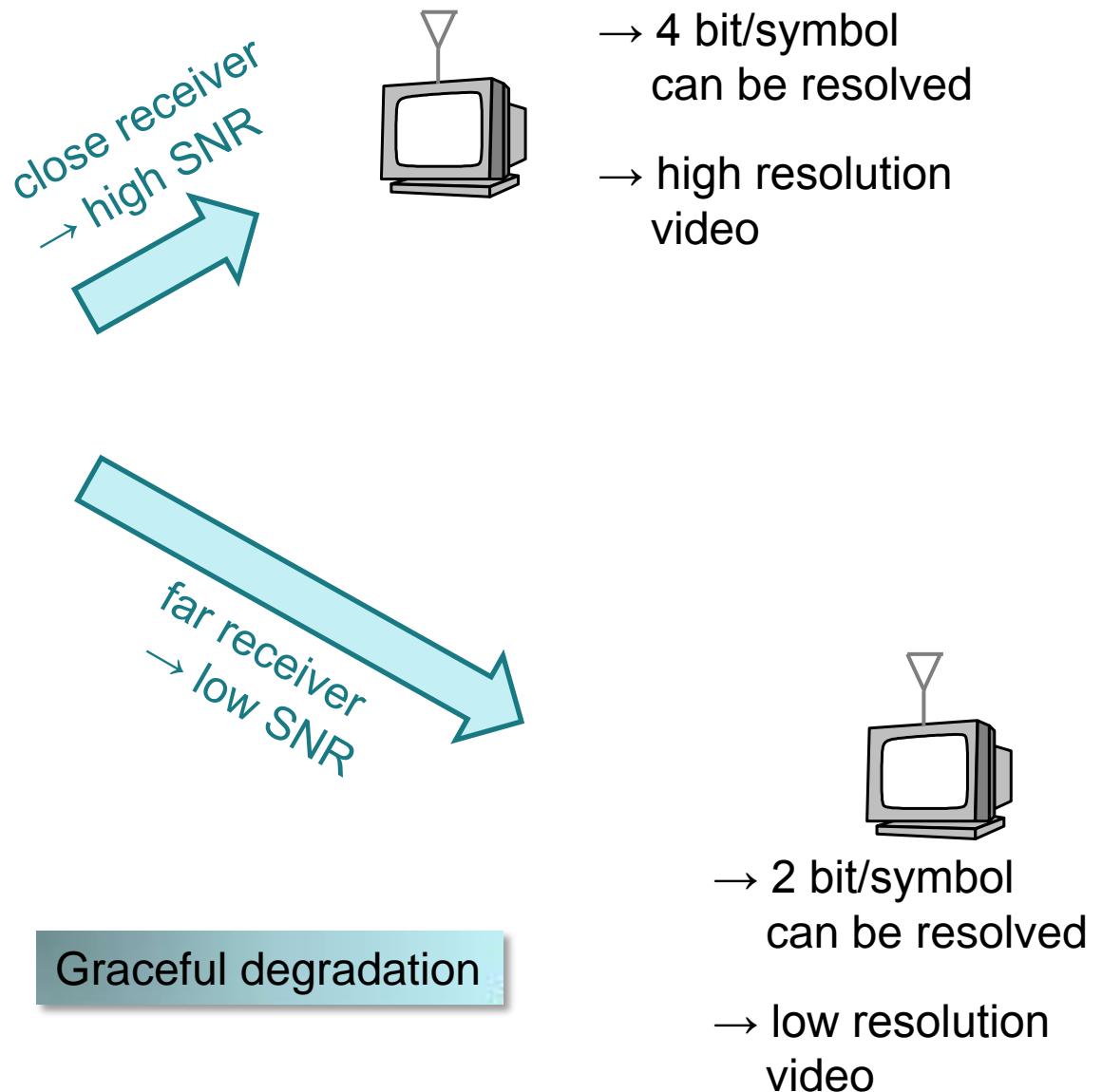
# Hierarchical Modulation/Multilevel Modulation

## Example: Wireless Video Transmission

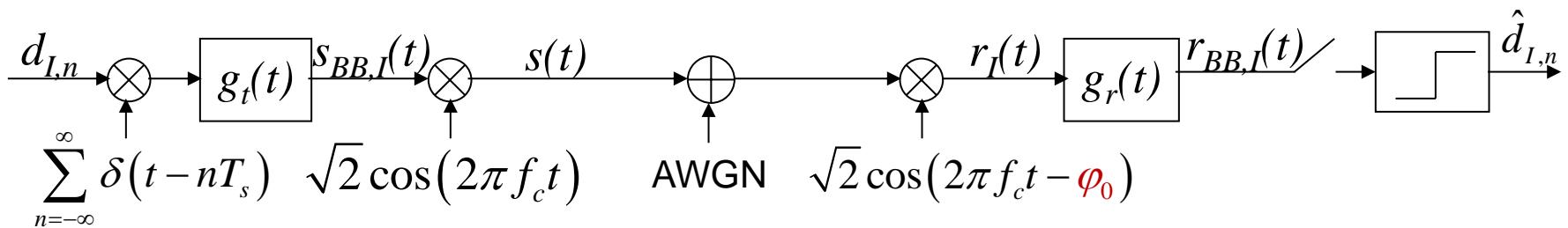


Video broadcast with

- hierarchical video codec
- hierarchical modulation



# BPSK with Carrier Phase Offset

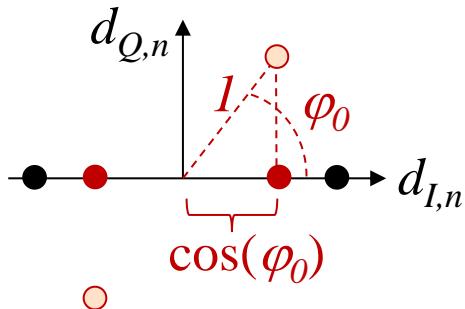


$$r_I(t) = s_{BB,I}(t) \cdot \sqrt{2} \cdot \cos(2\pi f_c t) \cdot \sqrt{2} \cdot \cos(2\pi f_c t - \varphi_0)$$

$$= s_{BB,I}(t) \left[ \cos(\varphi_0) + \cos(4\pi f_c t + \varphi_0) \right]$$

eliminated by low pass filter

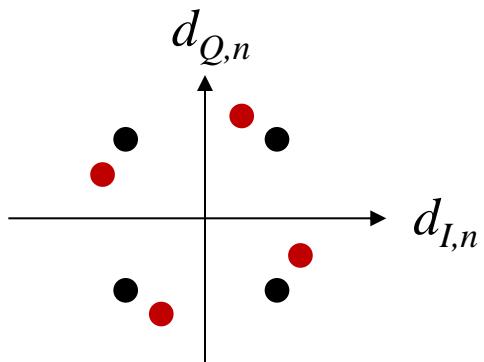
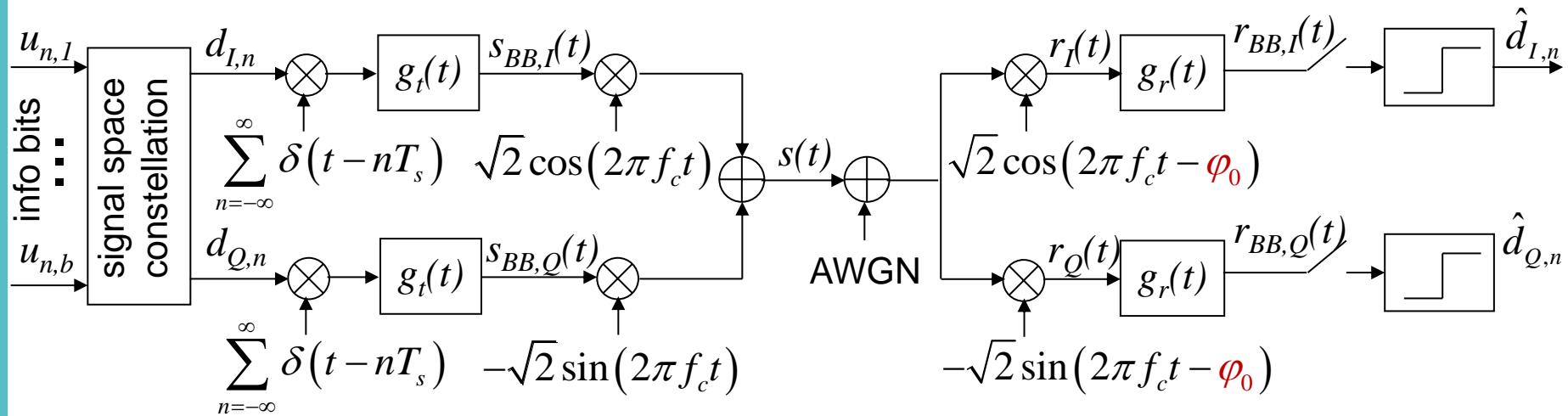
$$\Rightarrow r_{BB,I}(t) = s_{BB,I}(t) \underbrace{\cos(\varphi_0)}_{\in [-1, +1]} \quad \text{attenuation}$$



An unknown phase offset  $\varphi_0$  at the receiver causes an attenuation of the received signal by a factor  $\cos \varphi_0$ . This makes the signal more sensitive to noise.

For phase offsets  $\pi/2 \leq \varphi_0 \leq 3/2 \pi$ ,  $\cos \varphi_0$  becomes negative which results in detection errors even in noiseless transmission.

# QPSK with Carrier Phase Offset



An unknown phase offset  $\varphi_0$  at the receiver causes a rotation of the signal space constellation which shifts the constellation points closer to the decision thresholds. This makes the signal more sensitive to noise. Detection errors occur even in noiseless transmission for phase offsets  $\pi/4 \leq \varphi_0 \leq 7\pi/4$ .

# Coherent and Non-Coherent Detection

Modulation schemes which carry information in the carrier phase (e.g. PSK, QAM, ASK) need to be detected coherently, i.e. multiplication with the carrier at the receiver needs to be done with correct phase. Consequently, the carrier phase has to be recovered at the receiver before demodulation. Inaccurate phase recovery results in an unknown rotation of the signal space constellation of the sampled signal at the matched filter output which means increased detection error probability.

A solution to the problem is differential phase shift keying (DPSK) where the information is not contained in the absolute phase of the transmit symbols but in the phase difference of successive transmit symbols. Hence, a carrier phase offset does not harm the detection result and DPSK can be demodulated non-coherently, i.e. without carrier phase recovery. However, the price we pay is an increased noise power which results in a 3 dB higher required SNR compared to coherent PSK for the same error probability. Furthermore, differential modulation with modulation methods other than PSK is difficult.

Differential demodulation typically requires that the channel coefficient is about constant for at least two successive symbol durations. Channel estimation for coherent detection typically requires a significantly longer channel coherence time. Therefore, differential modulation with incoherent detection is attractive in highly time-variant channels, i.e. in channels with short coherence time. This is e.g. the case in wireless channels with high Doppler frequency, e.g. due to high transmitter and/or receiver speed and high carrier frequency.

# Differential Phase Shift Keying (DPSK)

In differential phase shift keying (DPSK), the information bit sequence is first mapped to a  $M$ -PSK symbol  $u_n$ . The transmit symbol  $d_n$  is then determined by a multiplication of the current information symbol by the previous transmit symbol, i.e.

$$d_n = u_n \cdot d_{n-1}.$$

Since  $u_n$  and  $d_n$  are PSK symbols, i.e.  $u_n = e^{j\varphi_{u,n}}$  and  $d_n = e^{j\varphi_{d,n}}$ , the information is contained in the phase difference of successively transmitted symbols  $d_{n-1}$  and  $d_n$ :

$$u_n = \frac{d_n}{d_{n-1}} = \frac{e^{j\varphi_{d,n}}}{e^{j\varphi_{d,n-1}}} = e^{j(\varphi_{d,n} - \varphi_{d,n-1})}.$$

At the receiver, we observe noisy samples  $y_n = h_n d_n + n_n$ , where  $n_n$  denotes additive noise and  $h_n = |h_n| e^{j\varphi_{h,n}}$  is a complex factor which models inaccurate carrier phase recovery. As long as  $h_n \approx h_{n-1}$  is about constant during two symbol periods, the information symbol can be estimated without explicit carrier phase estimation according to

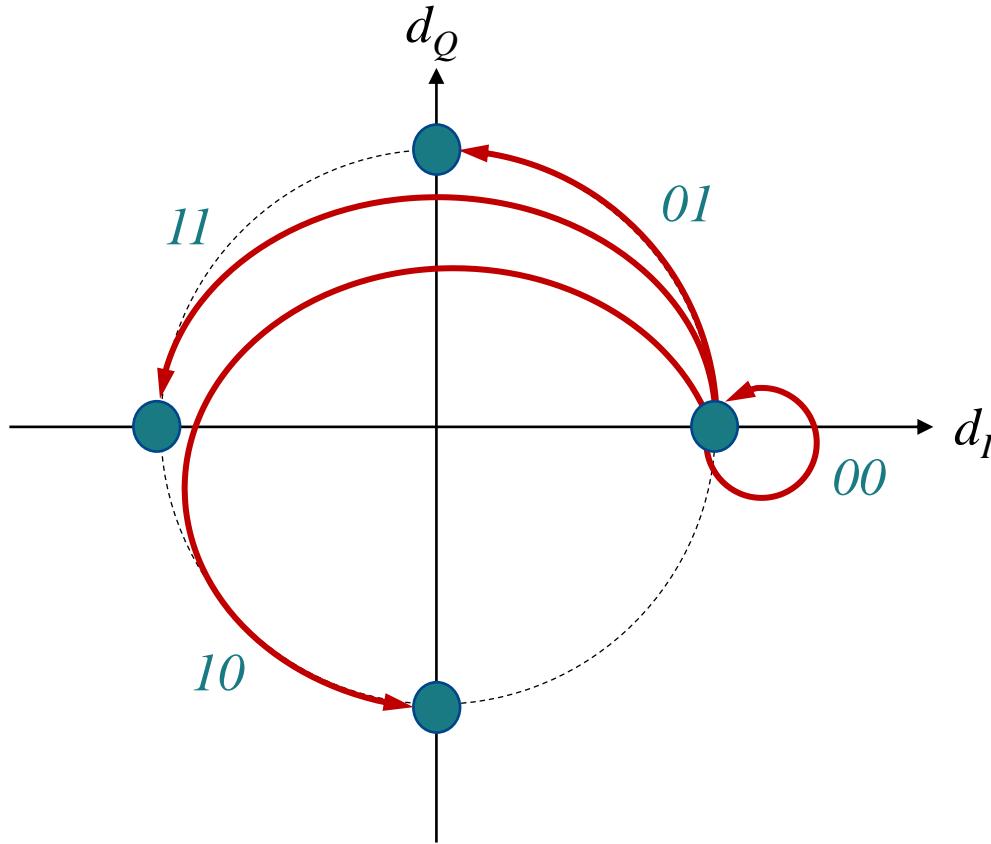
$$\hat{u}_n = \frac{y_n}{y_{n-1}}$$

which in the noise-free case yields

$$\hat{u}_n = \frac{y_n}{y_{n-1}} = \frac{h_n e^{j\varphi_{d,n}}}{h_{n-1} e^{j\varphi_{d,n-1}}} = e^{j(\varphi_{d,n} - \varphi_{d,n-1})}.$$

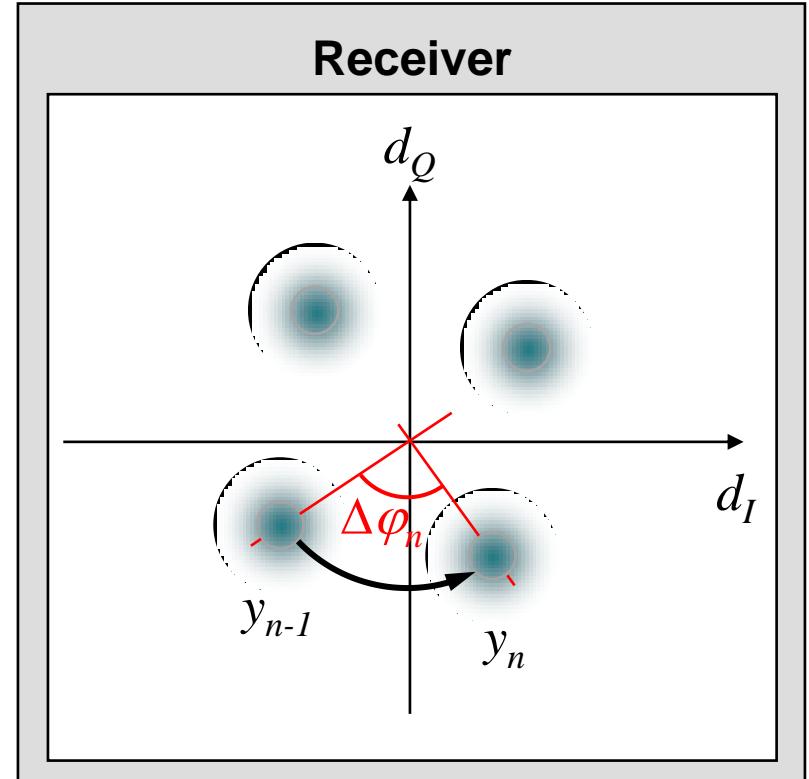
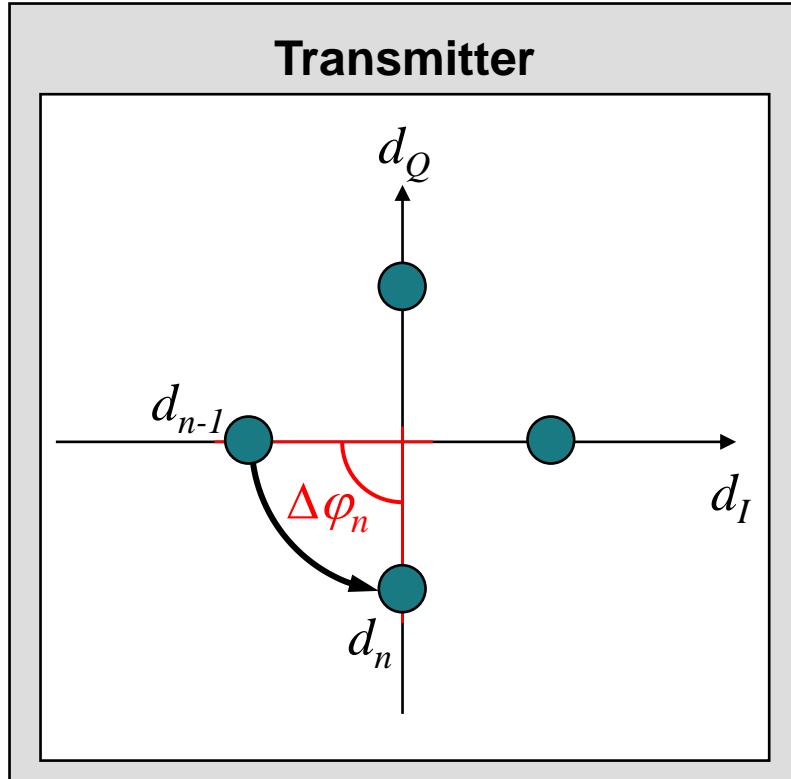
# Differential Phase Shift Keying (DPSK)

Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



# Differential Phase Shift Keying (DPSK)

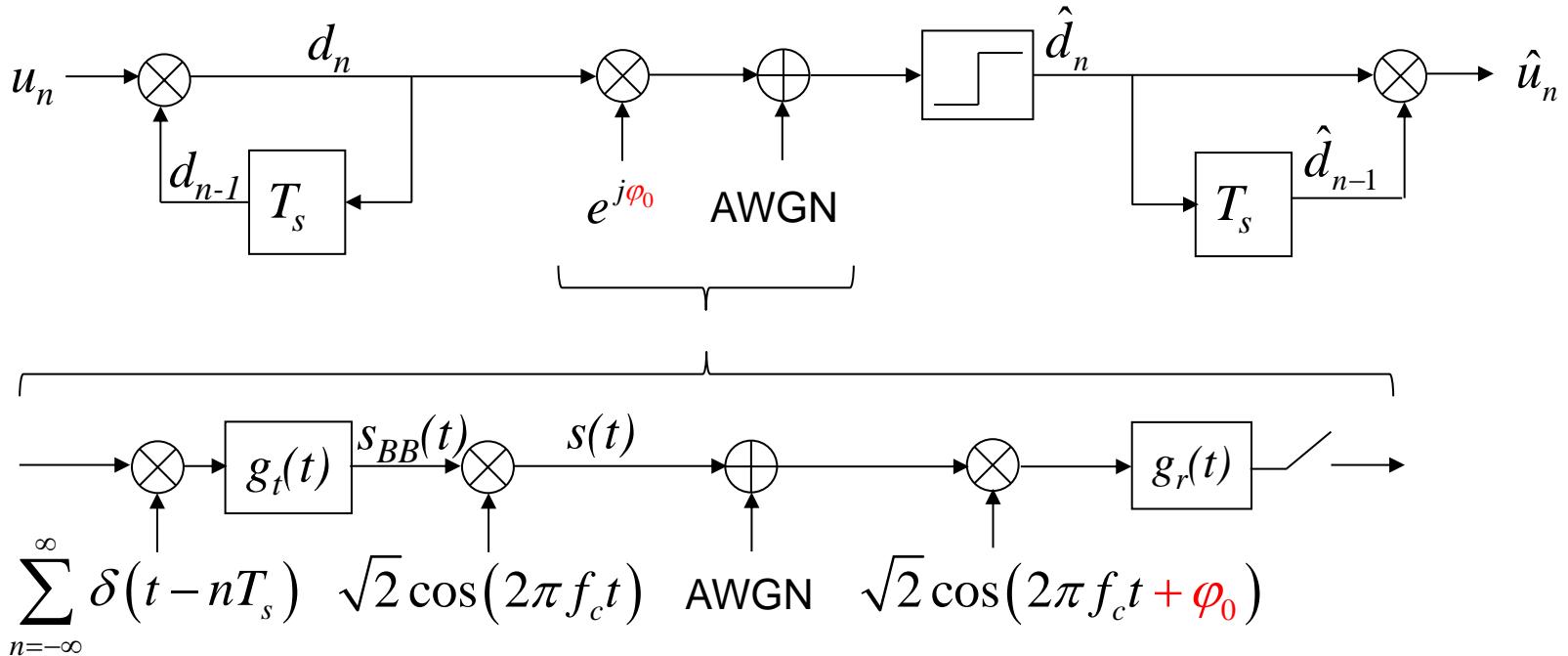
Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



Information (phase difference  $\Delta\varphi_n$ ) can be recovered independent of unknown carrier phase offset.

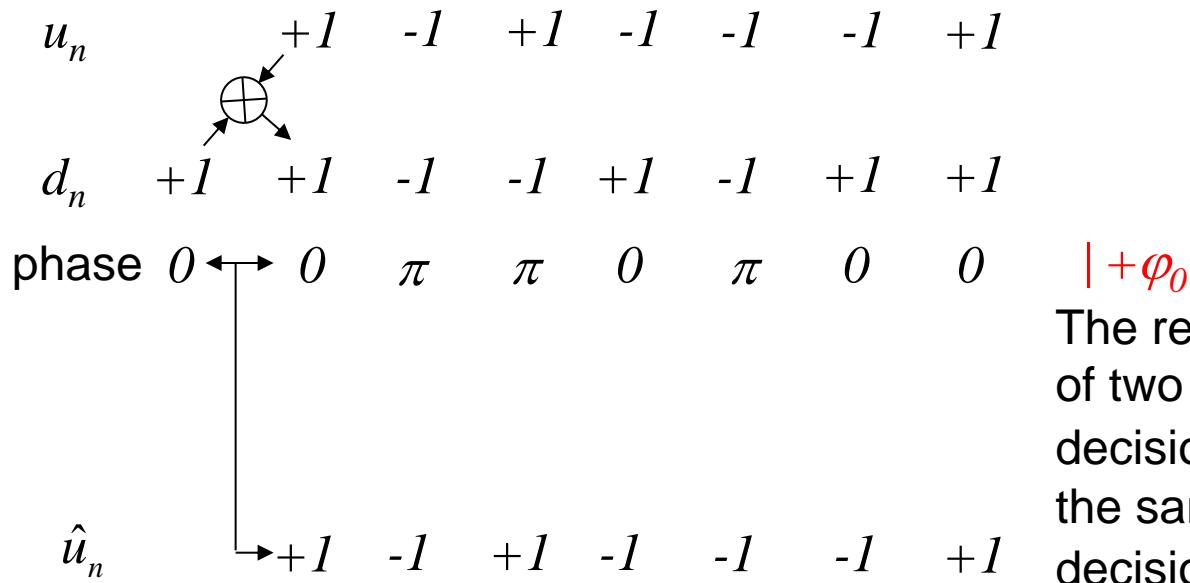
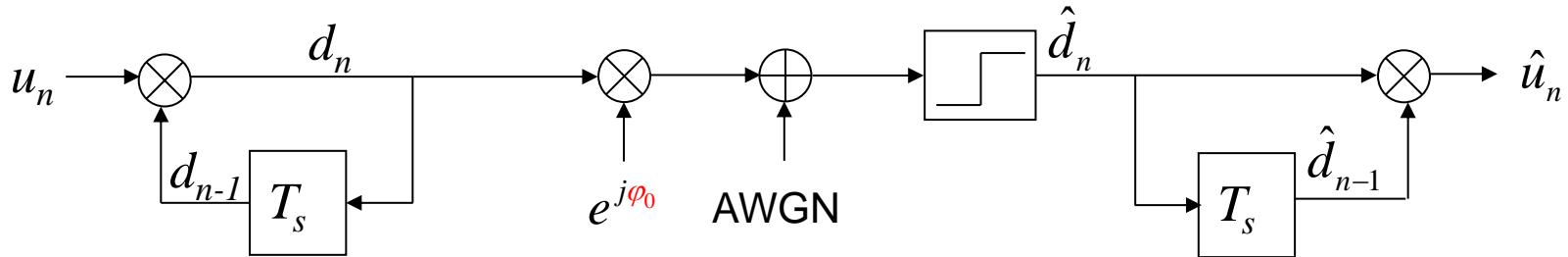
# Differential Binary Phase Shift Keying (DPSK) (1)

Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



# Differential Binary Phase Shift Keying (DPSK) (2)

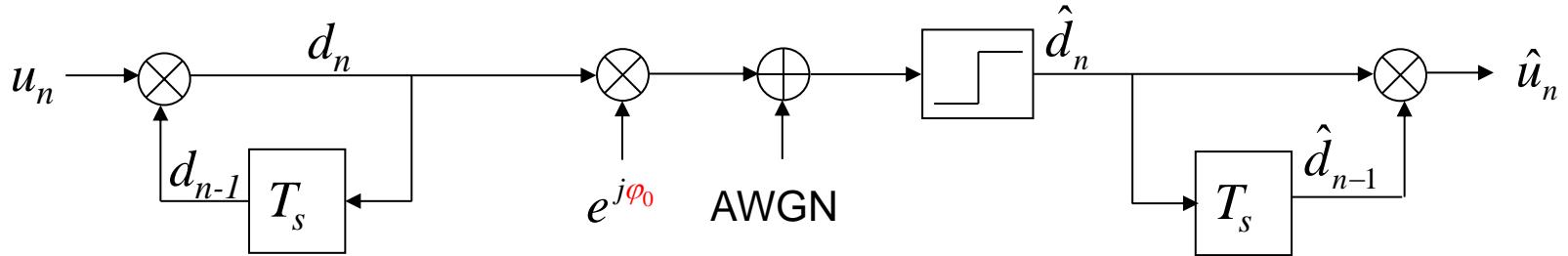
Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



The receiver compares the phases of two successive symbols. The decision is  $\hat{u}_n = +1$  if the phase is the same for both symbols, the decision is  $\hat{u}_n = -1$  if the phase changes. A constant phase offset  $\varphi_0$  does not cause detection errors.

# Differential Binary Phase Shift Keying (DPSK) (3)

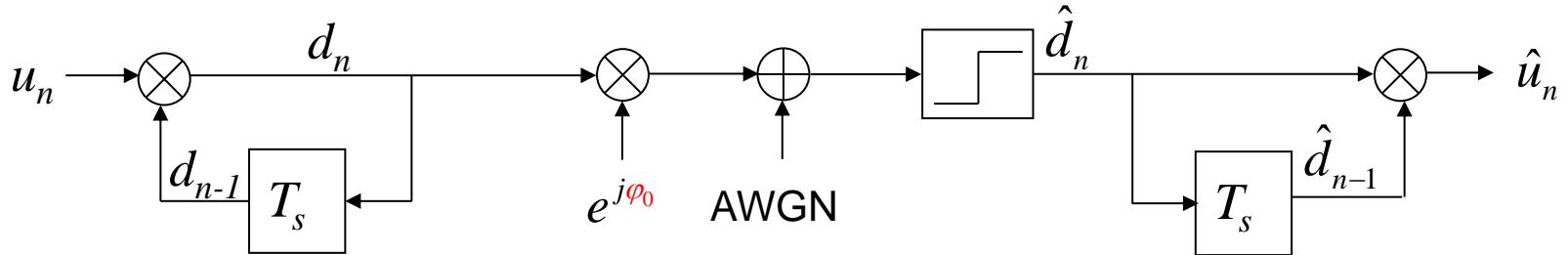
Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



$u_n$	+1	-1	+1	-1	-1	-1	+1	
$d_n$	+1	+1	-1	-1	+1	-1	+1	
phase	0	0	$\pi$	$\pi$	0	$\pi$	0	+ $\varphi_0$
$\hat{d}_n$	+1	+1	-1	-1	+1	-1	+1	
$\hat{u}_n$	+1	-1	+1	-1	-1	-1	+1	

# Differential Binary Phase Shift Keying (DPSK) (4)

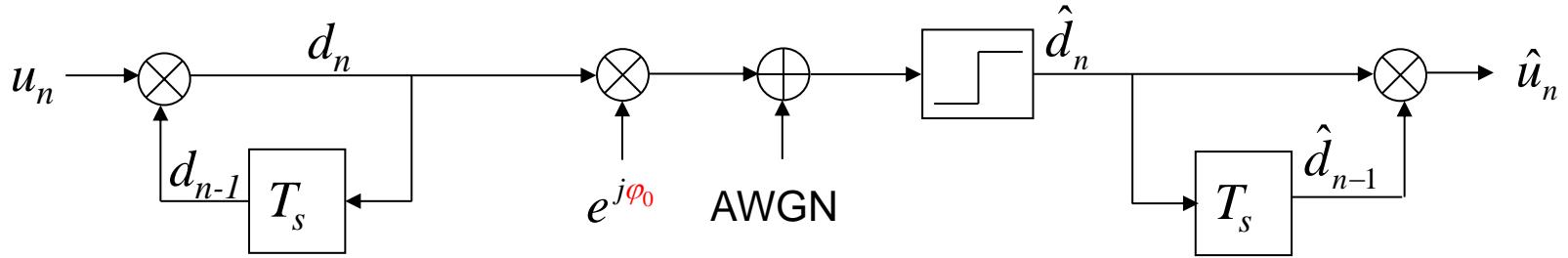
Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



$u_n$	+1	-1	+1	-1	-1	-1	+1
$d_n$	+1	+1	-1	-1	+1	-1	+1
phase	0	0	$\pi$	$\pi$	0	$\pi$	0
							$  +\varphi_0 = \pi$
$\hat{d}_n$	-1	-1	+1	+1	-1	+1	-1
$\hat{u}_n$	+1	-1	+1	-1	-1	-1	+1

# Differential Binary Phase Shift Keying (DPSK) (5)

Transmit information in the phase difference of two successive symbols rather than in the absolute phase.



$u_n$	+1	-1	+1	-1	-1	-1	+1	
$d_n$	+1	+1	-1	-1	+1	-1	+1	
phase	0	0	$\pi$	$\pi$	0	$\pi$	0	$+ \varphi_0$
$\hat{d}_n$	+1	+1	-1	-1	+1	-1	+1	
$\hat{u}_n$	+1	+1	-1	-1	-1	-1	+1	

$\oplus$

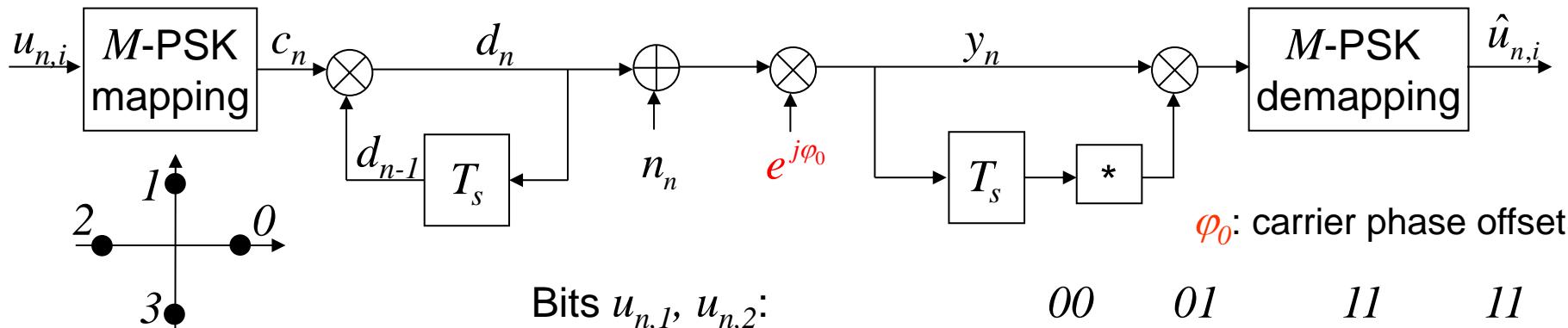
+1 transmission error

Due to differential encoding, a single transmission error causes a double error in the detected data.

# M-ary Differential Phase Shift Keying (M-DPSK)

Information is coded in the phase difference  $\Delta\varphi_n$  of two successive symbols  $d_n$  and  $d_{n-1}$ :

$$\varphi_n = \varphi_{n-1} + \Delta\varphi_n, \text{ where } \Delta\varphi_n = \frac{2\pi}{M}m \quad m=0,\dots,M-1$$



Example:

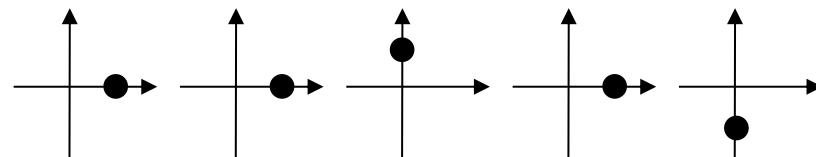
DQPSK ( $M=4$ ):

$m$	$u_{n,1}, u_{n,2}$	$\Delta\varphi_n$
0	0 0	0
1	0 1	$\pi/2$
2	1 0	$\pi$
3	1 1	$3\pi/2$

Bits $u_{n,1}, u_{n,2}$ :	00	01	11	11
Symbol index $m$ :	0	1	3	3
QPSK symbol $c_n$ :	+1	$e^{j\pi/2}$	$e^{-j\pi/2}$	$e^{-j\pi/2}$

DQPSK symbol  $d_n$ :

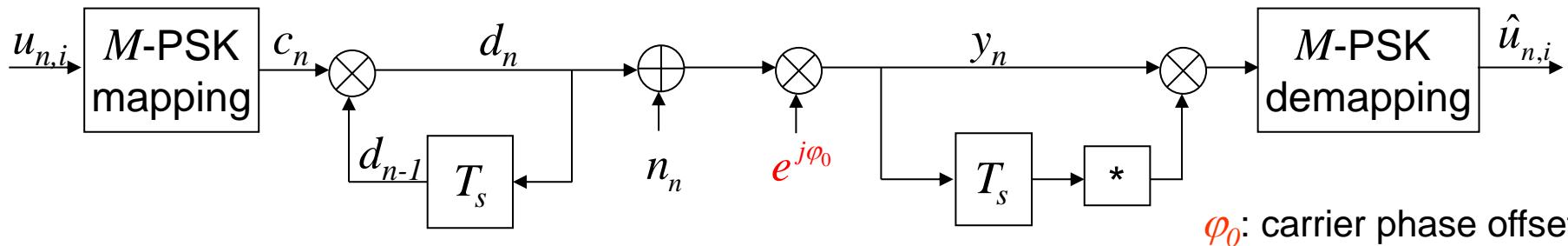
+1	+1	$e^{j\pi/2}$	+1	$e^{-j\pi/2}$
----	----	--------------	----	---------------



# M-ary Differential Phase Shift Keying (M-DPSK) – Soft Differential Demodulation

Information is coded in the phase difference  $\Delta\varphi_n$  of two successive symbols  $d_n$  and  $d_{n-1}$ :

$$\varphi_n = \varphi_{n-1} + \Delta\varphi_n, \text{ where } \Delta\varphi_n = \frac{2\pi}{M}m \quad m=0,\dots,M-1$$



$\varphi_0$ : carrier phase offset

$$\text{Receiver: } y_n = e^{j(\varphi_n + \varphi_0)} + n_n, \quad y_{n-1} = e^{j(\varphi_{n-1} + \varphi_0)} + n_{n-1}$$

$$y_n y_{n-1}^* = e^{j(\varphi_n + \varphi_0 - \varphi_{n-1} - \varphi_0)} + n_{n-1}^* e^{j(\varphi_n + \varphi_0)} + n_n e^{-j(\varphi_{n-1} + \varphi_0)} + n_n n_{n-1}^*$$

$\rightarrow 0$  for high SNR

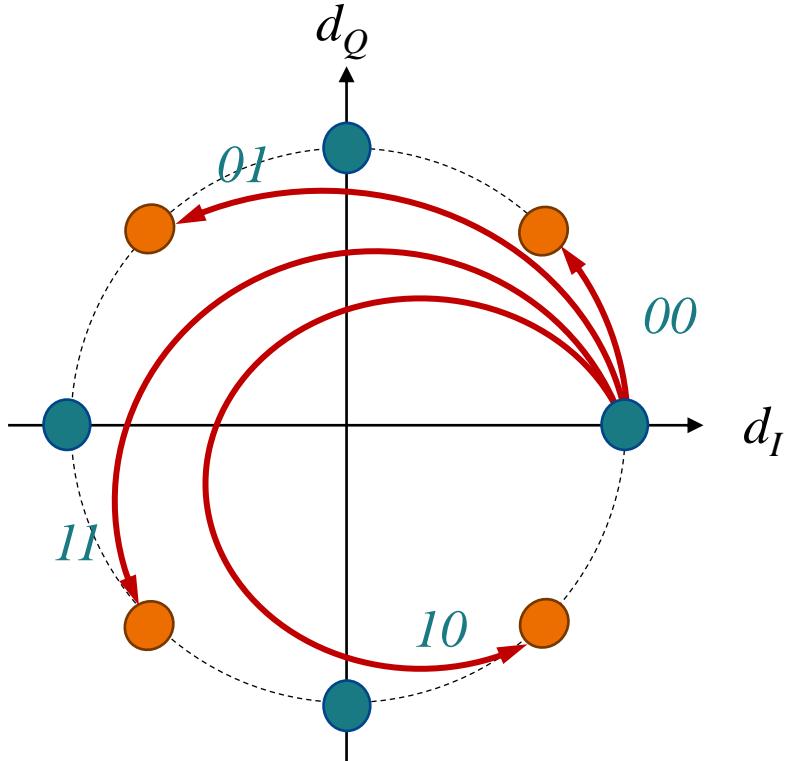
$$y_n y_{n-1}^* \approx \underbrace{e^{j(\varphi_n - \varphi_{n-1})}}_{\substack{\text{M-PSK symbol } c_n \\ \text{independent of } \varphi_0}} + \underbrace{n_{n-1}^* e^{j(\varphi_n + \varphi_0)} + n_n e^{-j(\varphi_{n-1} + \varphi_0)}}_{\text{AWGN with variance } 2\sigma_N^2}$$

DPSK can be detected non-coherently, i.e. without estimation of the carrier phase, on the expense of doubling the noise power. I.e. DPSK will require a 3 dB higher SNR compared to non-differential PSK for the same symbol error rate.

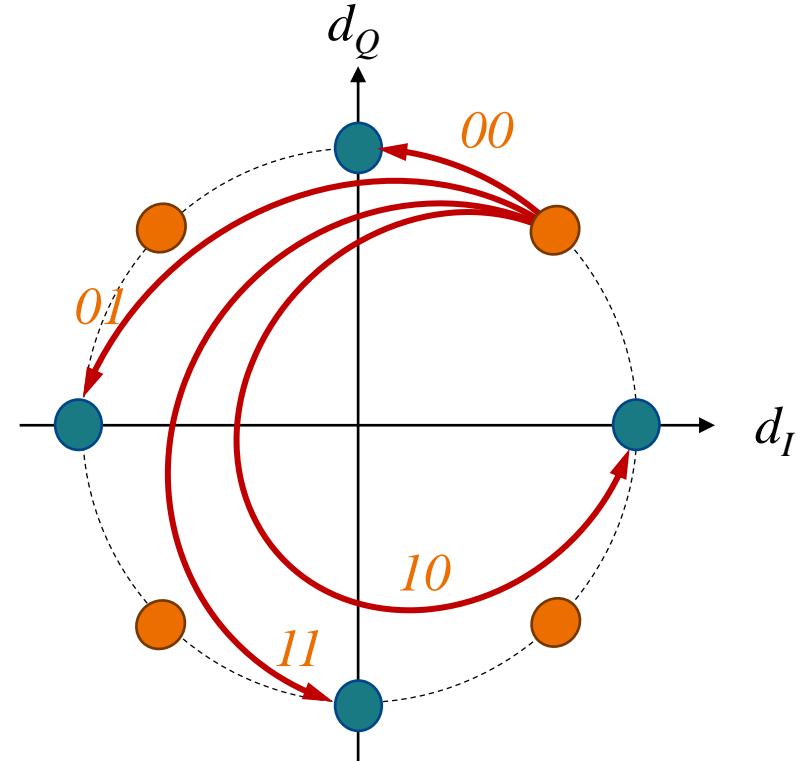
# $\pi/M$ Differential Phase Shift Keying ( $\pi/M$ -DPSK) (1)

Information is coded in the phase difference  $\Delta\varphi_n$  of two successive symbols  $d_n$  and  $d_{n-1}$ :

$$\varphi_n = \varphi_{n-1} + \Delta\varphi_n, \text{ where } \Delta\varphi_n = \frac{2\pi}{M}m + \lambda, \quad \lambda = \frac{\pi}{M}, \quad m=0,\dots,M-1$$



A **green** constellation point is always succeeded by a **red** constellation point.

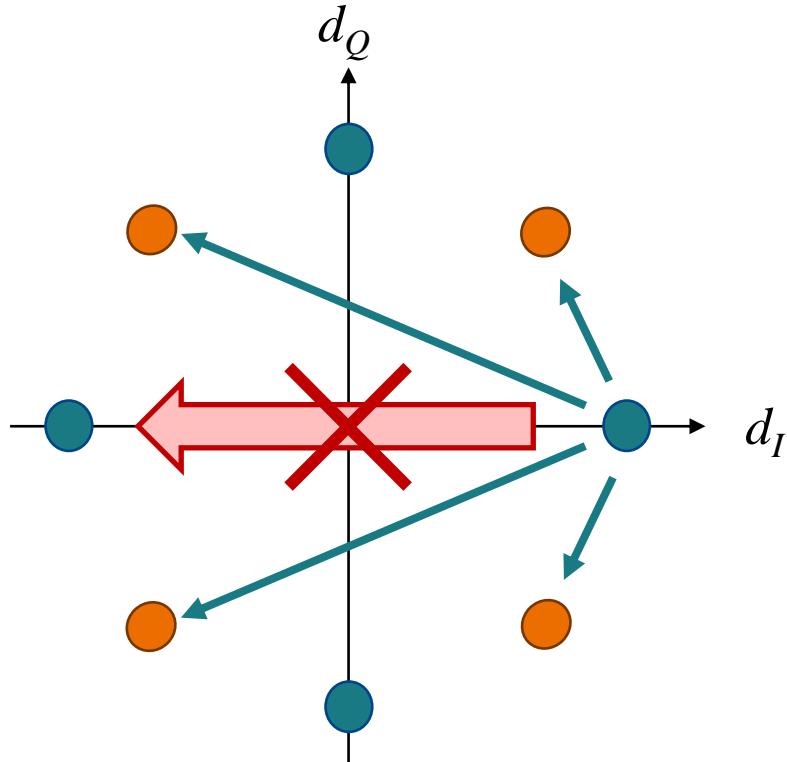


A **red** constellation point is always succeeded by a **green** constellation point.

# $\pi/M$ Differential Phase Shift Keying ( $\pi/M$ -DPSK) (2)

Information is coded in the phase difference  $\Delta\varphi_n$  of two successive symbols  $d_n$  and  $d_{n-1}$ :

$$\varphi_n = \varphi_{n-1} + \Delta\varphi_n, \text{ where } \Delta\varphi_n = \frac{2\pi}{M}m + \lambda, \quad \lambda = \frac{\pi}{M}, \quad m=0,\dots,M-1$$



Zero crossings which would cause deep envelope fades of the transmit signal are avoided.

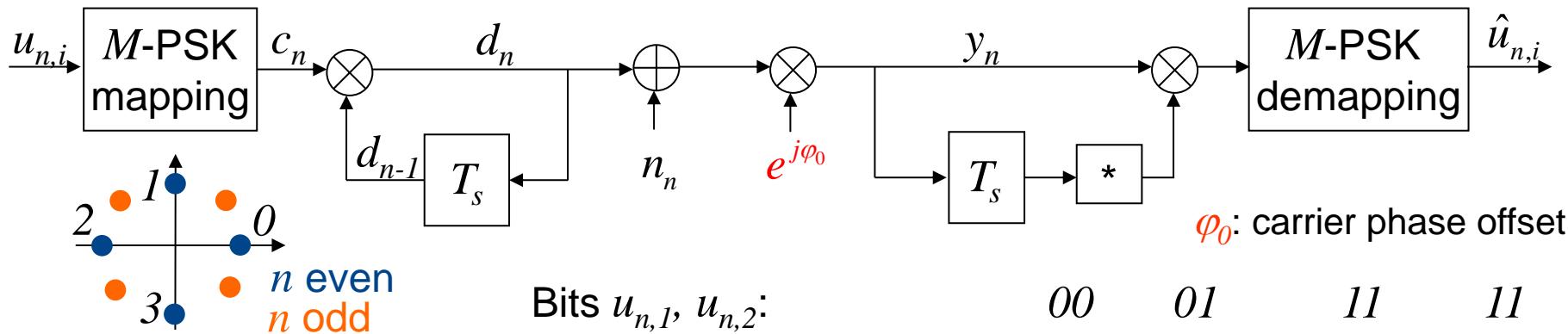
A **green** constellation point is always succeeded by a **red** constellation point.

A **red** constellation point is always succeeded by a **green** constellation point.

# $\pi/M$ Differential Phase Shift Keying ( $\pi/M$ -DPSK) (3)

Information is coded in the phase difference  $\Delta\varphi_n$  of two successive symbols  $d_n$  and  $d_{n-1}$ :

$$\varphi_n = \varphi_{n-1} + \Delta\varphi_n, \text{ where } \Delta\varphi_n = \frac{2\pi}{M}m + \lambda, \quad \lambda = \frac{\pi}{M}, \quad m=0,\dots,M-1$$



$\varphi_0$ : carrier phase offset

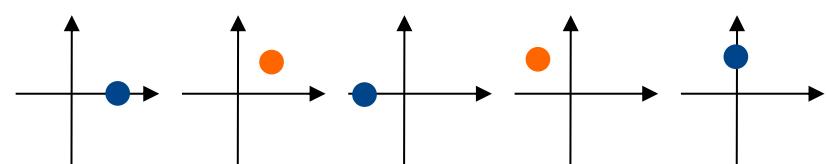
Bits $u_{n,1}, u_{n,2}$ :	00	01	11	11
Symbols $m$ :	0	1	3	3
QPSK symbols $c_n$ :	+1	$e^{j\pi/2}$	$e^{-j\pi/2}$	$e^{-j\pi/2}$

DQPSK symbols $d_n$ :	+1	$e^{j\pi/4}$	-1	$e^{j3\pi/4}$	$e^{j\pi/2}$
-----------------------	----	--------------	----	---------------	--------------

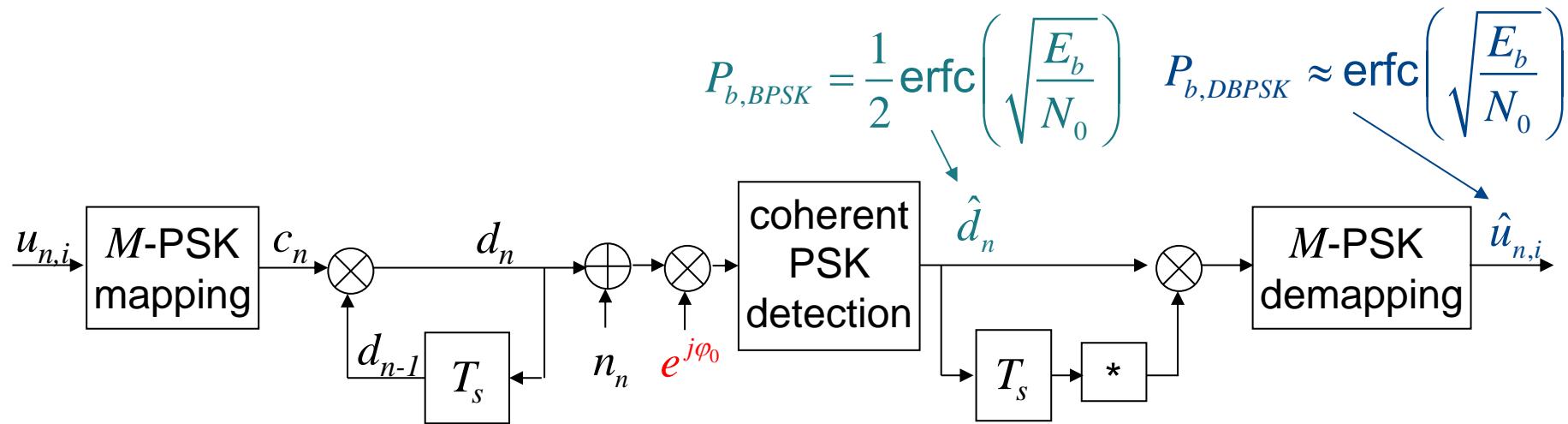
Example:

$\pi/4$ -DQPSK ( $M=4$ ):

$m$	$u_{n,1}, u_{n,2}$	$\Delta\varphi_n$
0	0 0	$\pi/4$
1	0 1	$3\pi/4$
2	1 0	$5\pi/4$
3	1 1	$7\pi/4$



# Coherent Detection of DPSK



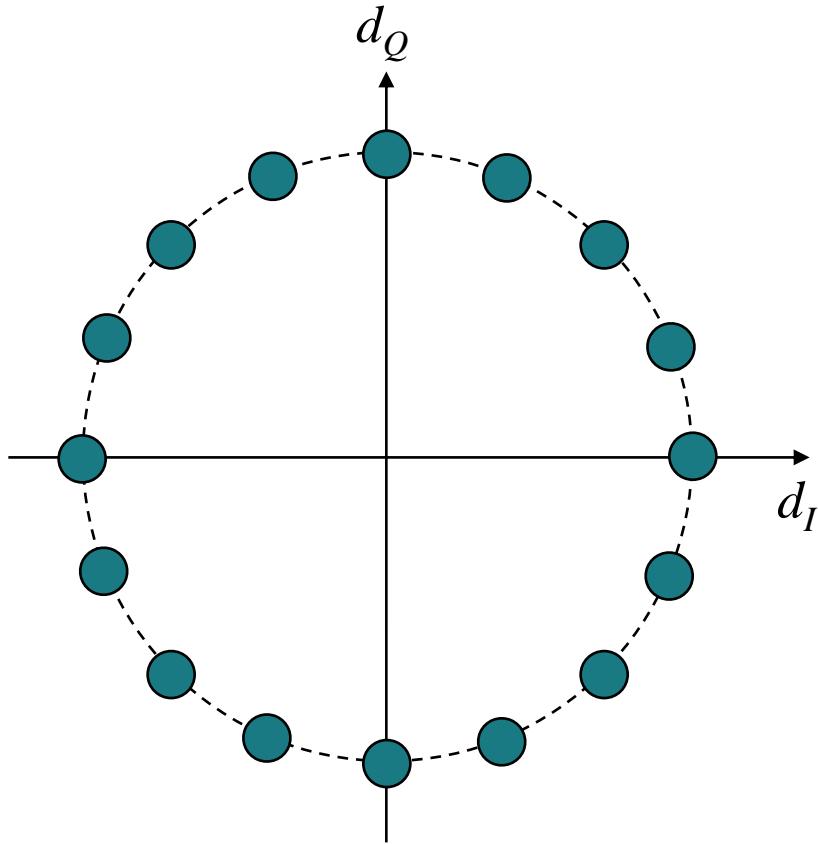
Example: Binary DPSK:

The error probability in the symbols  $\hat{d}_n$  is given by the bit error probability of coherently detected BPSK. Subsequent differential detection causes double errors. Hence, neglecting multiple symbol errors, the bit error probability of coherently detected DBPSK is twice the bit error probability of coherently detected non-differential BPSK:

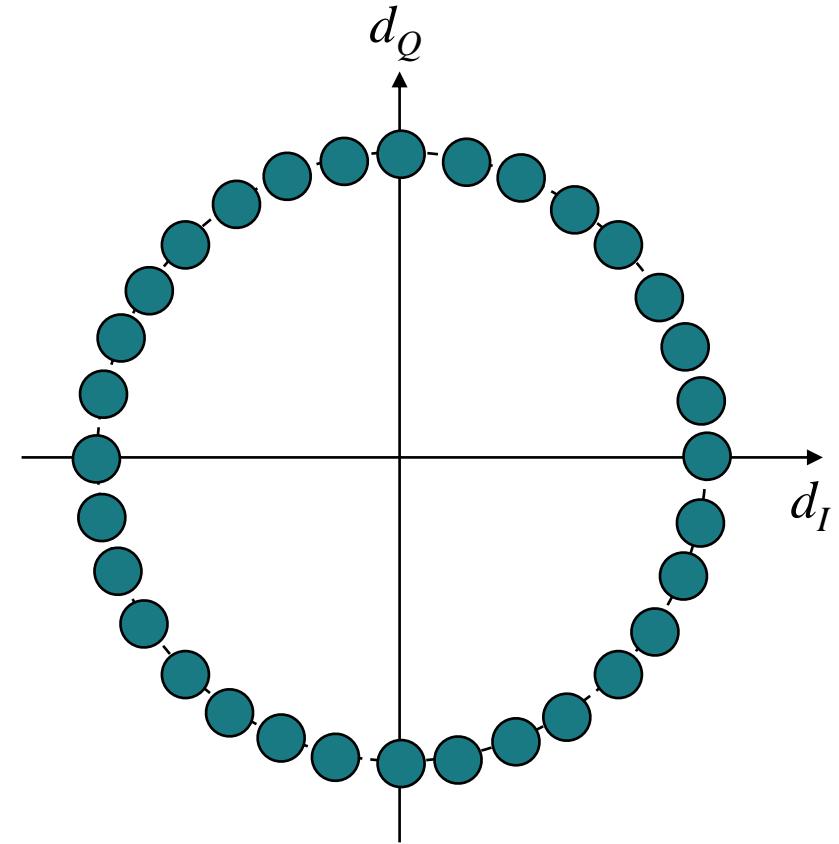
$$P_{b,DBPSK} \approx 2P_{b,BPSK} = \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

# Higher Order Phase Shift Keying

16-PSK

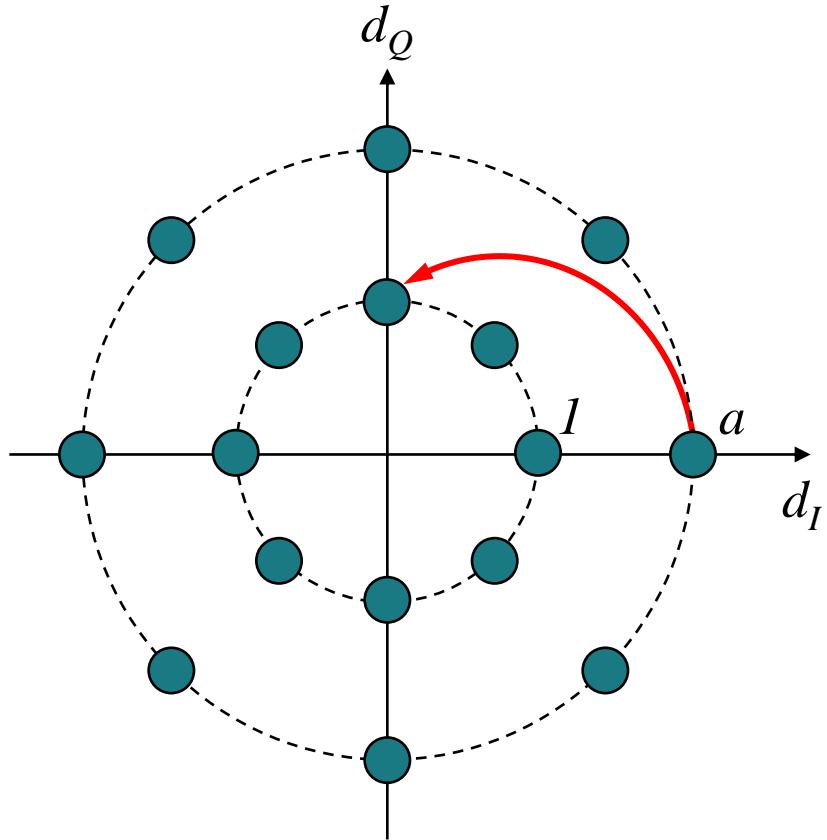


32-PSK



Problem: Euclidean distance between constellation points decreases  
→ high error probability in noisy channels.

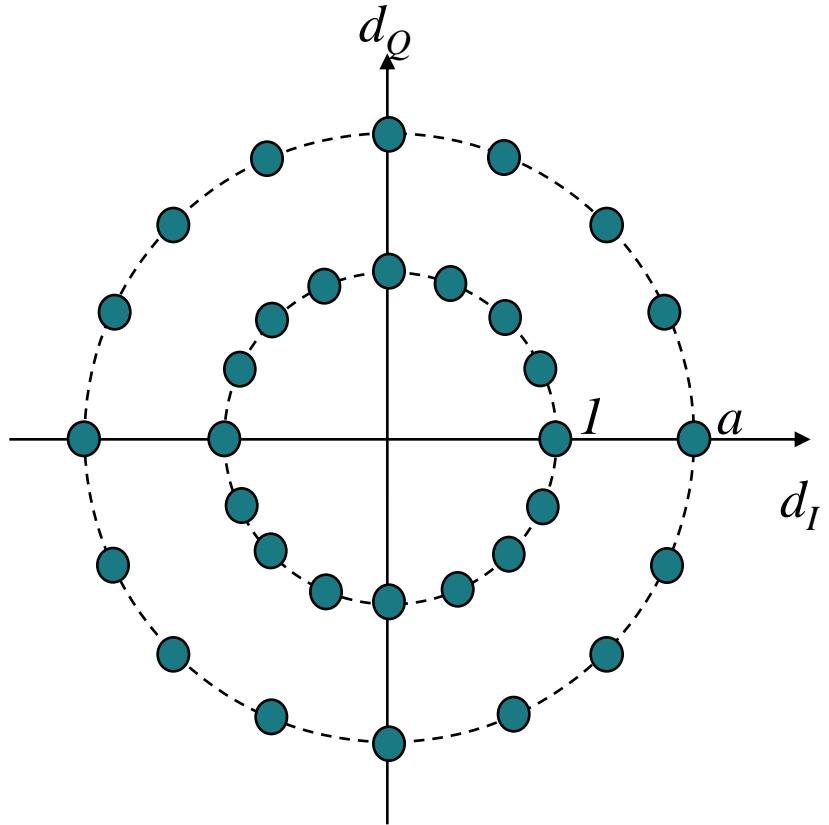
# Differential Amplitude and Phase Shift Keying: 16-DAPS K



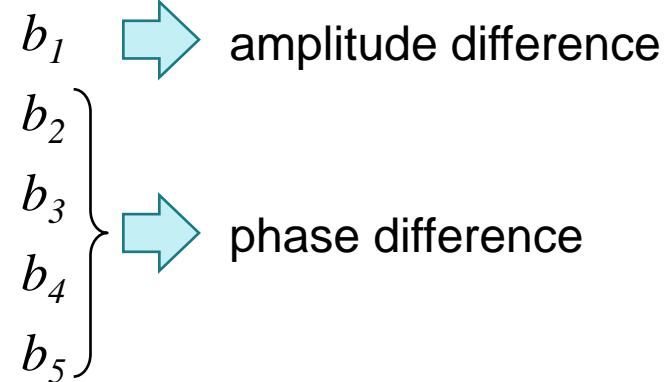
Information is contained in phase and amplitude difference of two successive symbols.

$$d_n = d_{n-1} \cdot u_n$$

# Differential Amplitude and Phase Shift Keying: 32-DAPSQ



Bit mapping:

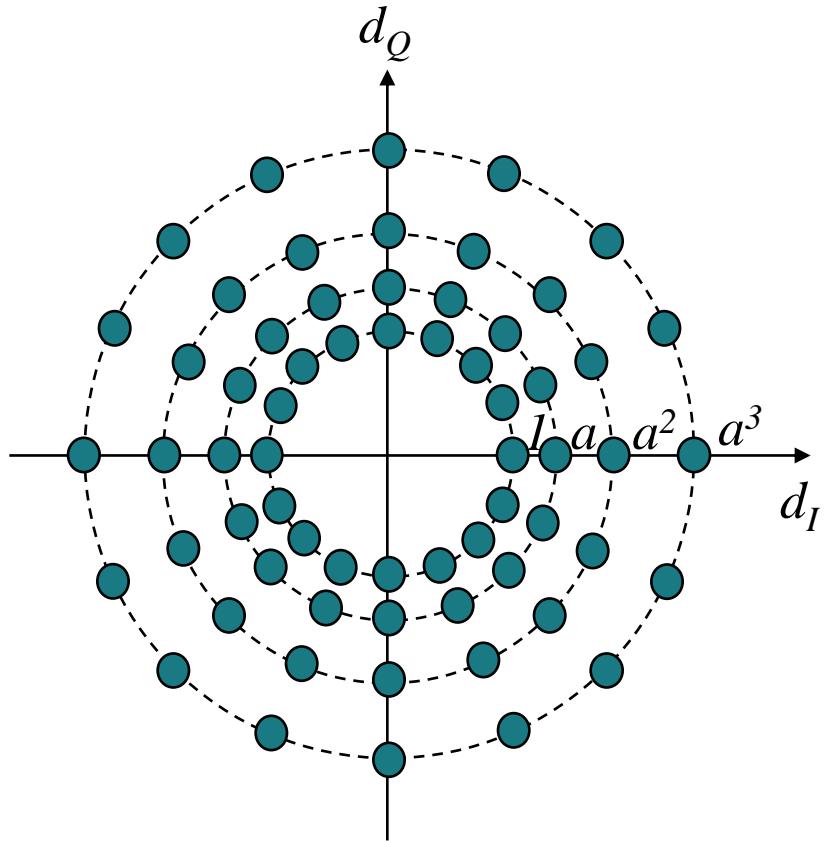


$$d_n = d_{n-1} \cdot u_n$$

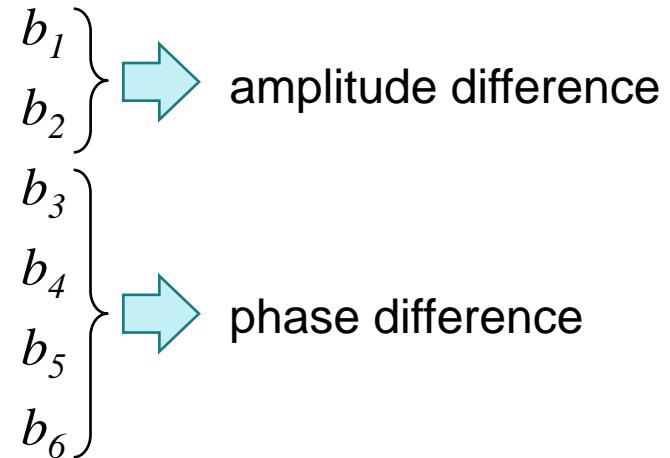
$$\arg\{u_n\} \in \{0, \Delta\varphi, \dots, 15\Delta\varphi\}; \quad \Delta\varphi = \frac{\pi}{8}$$

$$|u_n| \in \left\{ \frac{1}{a}, 1, a \right\}$$

# Differential Amplitude and Phase Shift Keying: 64-DAPSQ



Bit mapping:

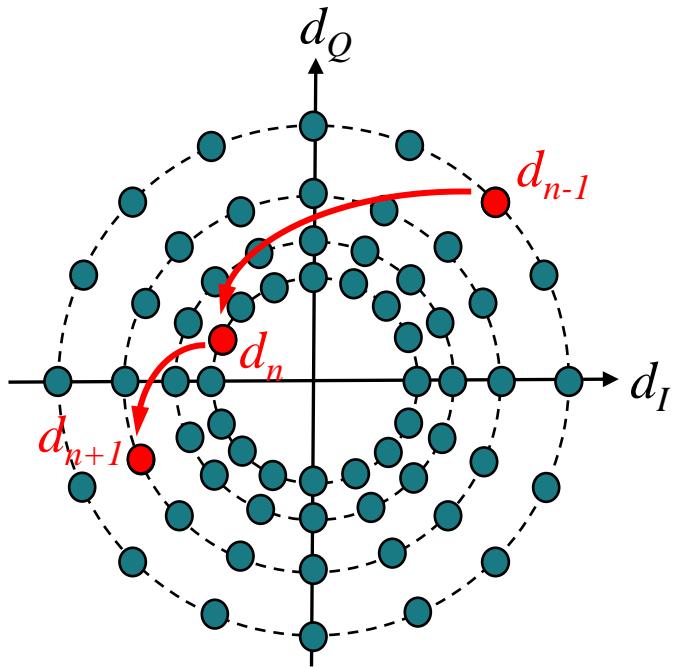


$$d_n = d_{n-1} \cdot u_n$$

$$\arg\{u_n\} \in \{0, \Delta\varphi, \dots, 15\Delta\varphi\}; \quad \Delta\varphi = \frac{\pi}{8}$$

$$|u_n| \in \left\{ \frac{1}{a^3}, \frac{1}{a^2}, \frac{1}{a}, 1, a, a^2, a^3 \right\}$$

# 64-DAPSK - Example



info bits: ... **01 11 01 11 11 00 ...**

$$|d_{n-1}| = a^3 \rightarrow u_n = a^{-3} e^{j5\pi/8}$$

$$|d_n| = 1 \rightarrow u_{n+1} = a^2 e^{j2\pi/8}$$

$$|d_{n+1}| = a^2$$

**Bit mapping ( $b_1 b_2 b_3 b_4 b_5 b_6 \rightarrow u_n$ )**

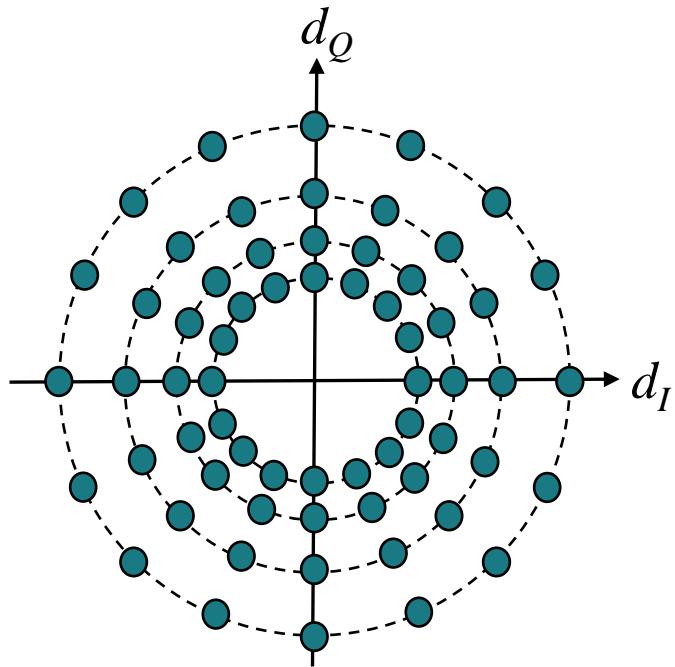
amplitude difference:  $|u_n|$

$ d_{n-1} $	info bits $b_1 b_2$			
	<b>00</b>	<b>01</b>	<b>11</b>	<b>10</b>
1	1	$a$	$a^2$	$a^3$
$a$	1	$a$	$a^2$	$1/a$
$a^2$	1	$a$	$1/a^2$	$1/a$
$a^3$	1	$1/a^3$	$1/a^2$	$1/a$

phase difference:  $\arg(u_n)$

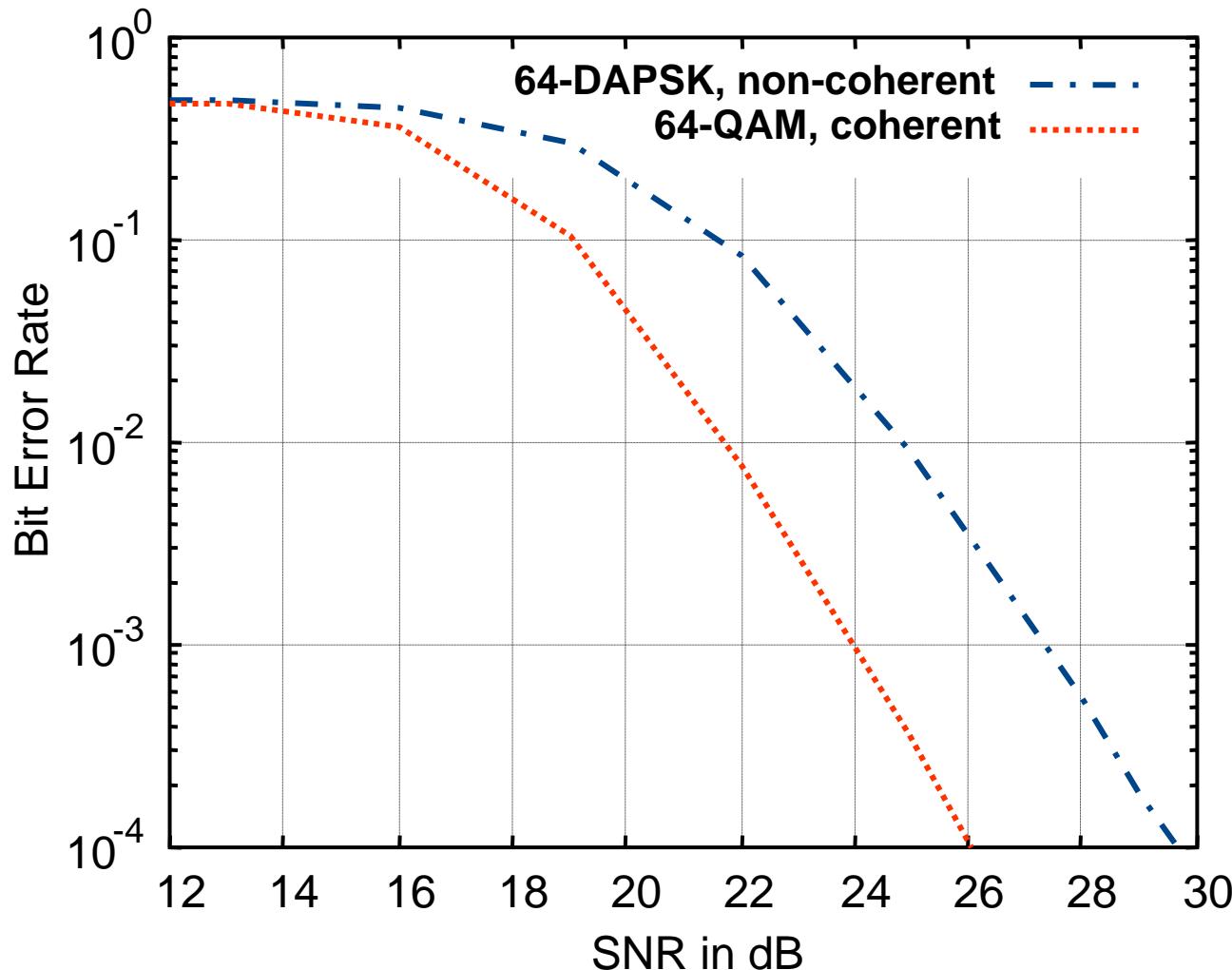
info bits $b_3 b_4 b_5 b_6$				
$b_5 b_6 \backslash b_3 b_4$	<b>00</b>	<b>01</b>	<b>11</b>	<b>10</b>
<b>00</b>	0	$\Delta\varphi$	$2\Delta\varphi$	$3\Delta\varphi$
<b>01</b>	$7\Delta\varphi$	$6\Delta\varphi$	$5\Delta\varphi$	$4\Delta\varphi$
<b>11</b>	$8\Delta\varphi$	$9\Delta\varphi$	$10\Delta\varphi$	$11\Delta\varphi$
<b>10</b>	$15\Delta\varphi$	$14\Delta\varphi$	$13\Delta\varphi$	$12\Delta\varphi$

# M-DAPSK: Parameters



cardinality	amplitude levels	phases	optimum amplitude factor $a$ for minimum average BER
$M = 16$	2	8	2
$M = 32$	2	16	1,6
$M = 64$	4	16	1,4
$M = 128$	4	32	1,3

# Performance Comparison of 64-DAPSK and 64-QAM



# Non-Linear Digital Modulation Methods: Frequency Shift Keying (FSK)

Frequency shift keying (FSK) is a non-linear modulation scheme. An FSK transmit signal  $s(t)$  can in general not be created using a symbol mapper followed by a QAM modulator. Instead, the information is contained in the instantaneous frequency of the transmit signal  $s(t)$ . It is a major advantage with respect to non-linear components such as high power amplifiers in the transmitter chain that an FSK signal has constant envelope. I.e., the magnitude of the equivalent baseband signal

$$s_{BB}(t) = A e^{j\varphi(t)}$$

is a constant and the trajectory which describes the equivalent baseband signal in the complex signal space is a circle. However, as any constant envelope signal which carries information, an FSK signal always has infinite bandwidth. Except for some special cases, it is more difficult to determine the exact shape of the average power spectral density  $S_{ss}(f)$  than for linear modulation schemes. However, numerical methods for computation of the power spectral density exist.

In order to reduce the effect of interference into adjacent frequency bands, the side lobes of the power spectral density should be made as low as possible. For that purpose, it is advantageous to design the transmit signal such that its phase  $\varphi(t)$  is a continuous function over time. I.e., the frequency should not just switch according to the current data symbol by e.g. using oscillators of different frequencies but it should be made sure that no phase discontinuity occurs at the time of switching. Sudden changes in time domain signals cause strong components at high frequencies in the frequency domain. This general rule also applies to sudden phase changes.

# Continuous Phase Modulation (CPM) (1)

Frequency shift keying with continuous phase  $\varphi(t)$  of the transmit signal  $s(t)$  is called *continuous phase modulation (CPM)*. Note, that CPM still needs infinite bandwidth. However, the attenuation of the side lobes in the power spectral density is significantly stronger than for FSK with phase discontinuities. Therefore, in practice FSK is virtually always applied as CPM. The phase of the equivalent baseband signal for CPM is given by

$$\varphi(t) = 2\pi\Delta f T_s \int_{-\infty}^t \sum_{m=-\infty}^{\infty} d_m g_t(\tau - mT_s) d\tau + \varphi_0,$$

where  $\varphi_0$  is an arbitrary start phase,  $d_m \in \{-M/2, -M/2+1, \dots, -1, +1, \dots, +M/2\}$  is the  $m$ -th data symbol and  $g_t(t)$  is called the frequency pulse. We normalize the frequency pulse such that

$$\int_{-\infty}^{\infty} g_t(t) dt = 1.$$

The frequency pulse determines, how the frequency of the transmit signal is modulated. In case of hard switching between instantaneous frequencies,  $g_t(t)$  will be a rectangular pulse. However, often other pulses are applied in order to obtain a softer frequency shift which is beneficial for reducing the side lobes of the power spectral density. Moreover, the frequency pulse  $g_t(t)$  can span over more than a symbol duration  $T_s$  resulting in partial response modulation. E.g. in the GSM system, binary FSK with a Gaussian-like pulse of duration  $5T_s$  is applied.

The integral in the equation above ensures a continuous phase. As due to the integration the instantaneous phase at time  $mT_s$  depends on all previous data symbols  $d_n$ ,  $n \leq m$ , CPM is a modulation scheme with memory.

# Continuous Phase Modulation (CPM) (2)

The integral

$$q_t(t) = \int_{-\infty}^t g_t(\tau) d\tau$$

over the frequency pulse is called the *phase pulse*  $q_t(t)$ . It characterizes how the phase of the equivalent baseband signal changes over time. In case of a rectangular frequency pulse

$$g_t(t) = \frac{1}{T_s} \text{rect}\left(\frac{t - \frac{T_s}{2}}{\frac{2}{T_s}}\right),$$

we obtain a linear phase  $\varphi(t)$ . Note, that for a rectangular frequency pulse, the phase  $\varphi(t)$  of the equivalent baseband signal shows sharp edges when the data symbol  $d_m$  changes its value. This causes relatively high side lobes in the power spectral density. Therefore, a smoother transition as can be obtained by softer (e.g. Gaussian-like) partial response frequency pulses is beneficial from a spectrum point of view.

# Modulation Index (1)

The modulation index  $h$  is a parameter for characterization of FSK. It describes the phase shift of the equivalent baseband signal which is caused by a single data symbol  $d_n$ .

The modulation index is defined by the frequency separation  $2\Delta f$  of the transmit signal for different data relative to the symbol rate  $1/T_s$ :

$$h = \frac{2\Delta f}{\frac{1}{T_s}} = 2\Delta f T_s.$$

The higher the modulation index  $h$ , the more bandwidth (in the sense that the main lobe of the power spectral density gets wider) is required for the transmit signal.

With this definition, the phase of the equivalent baseband signal  $s_{BB}(t)$  can be expressed by

$$\begin{aligned} \varphi(t) &= 2\pi\Delta f T_s \int_{-\infty}^t \sum_{m=-\infty}^{\infty} d_m g_t(\tau - mT_s) d\tau + \varphi_0 = \pi h \sum_{m=-\infty}^{\infty} d_m \int_{-\infty}^t g_t(\tau - mT_s) d\tau + \varphi_0 \\ &= \pi h \sum_{m=-\infty}^{\infty} d_m q(t - mT_s) + \varphi_0. \end{aligned}$$

It can be seen from the equation above, that in case of FSK with a rectangular full response frequency pulse  $g_t(t)$ , the phase  $\varphi(t)$  of the equivalent baseband signal changes linearly over a symbol period  $T_s$  by  $\pi h d_n$ . Here, we assumed that  $g_t(t)$  is normalized such that

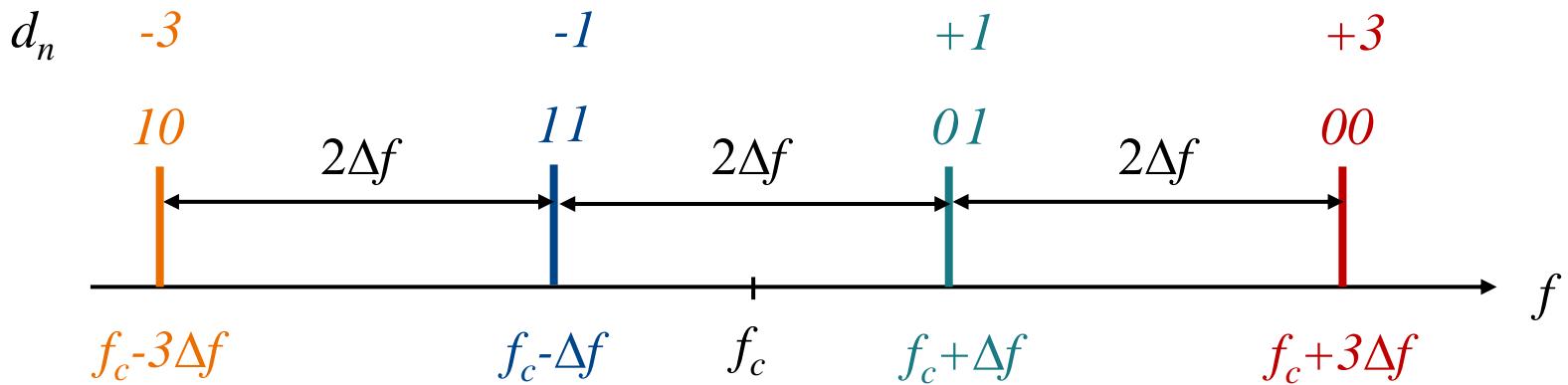
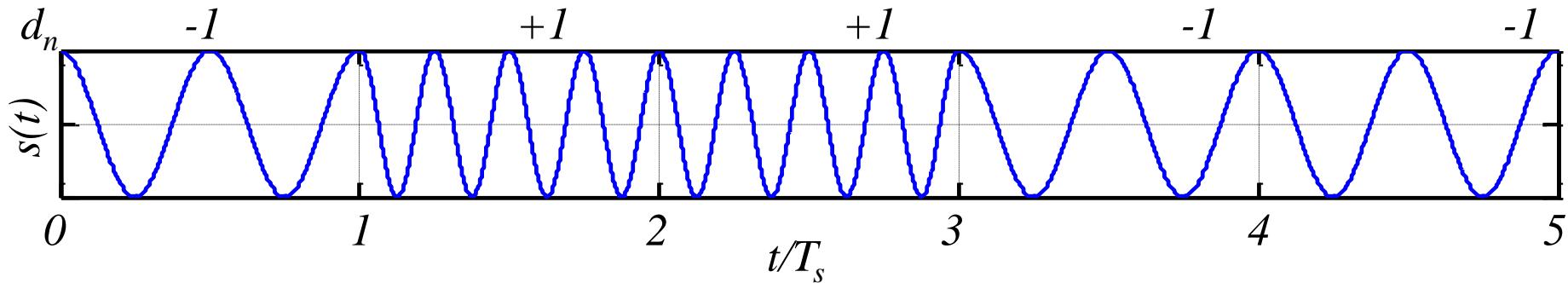
$$\int_{-\infty}^{\infty} g_t(t) dt = 1.$$

# Modulation Index (2)

The behaviour of the equivalent baseband signal  $s_{BB}(t)$  can be illustrated by the respective trajectory in the signal space. As an FSK signal  $s(t)$  has constant envelope, the trajectory which describes the equivalent baseband signal  $s_{BB}(t)$  is a circle.

For a full response frequency pulse, the trajectory will move along the circle and experience a phase shift by  $\pi h d_n$  during one symbol period  $T_s$ . Starting with an arbitrary phase  $\varphi_0$  on any point of the circle, e.g.  $\varphi_0=0$ , we can mark the possible phases at the end of each symbol period in the signal space. Those marked points correspond to the samples of the equivalent baseband signal  $s_{BB}(t)$  at multiples of the symbol duration  $T_s$  in the same way as the constellation points of a linear modulation scheme. However, note that unlike for linear modulation schemes, in case of FSK those marked points in the signal space do not directly represent the data symbols  $d_n$ .

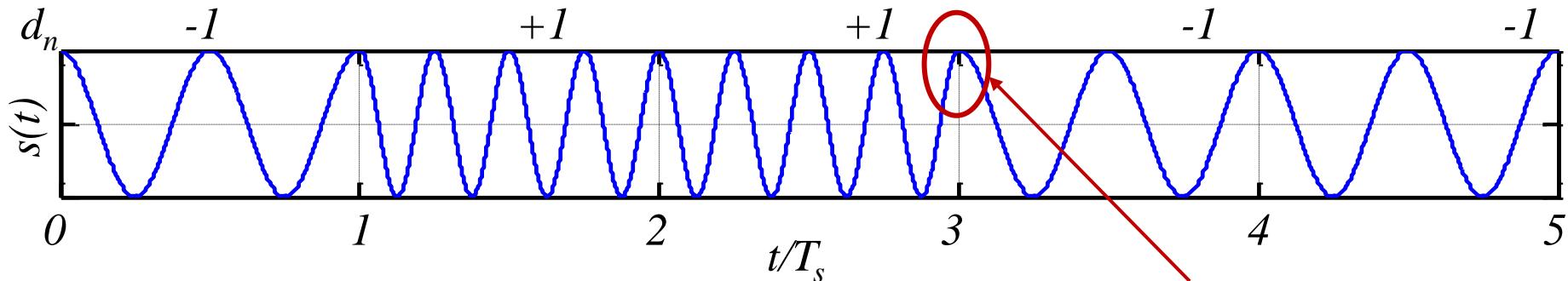
# Non-Linear Digital Modulation Methods: Frequency Shift Keying (FSK)



Instantaneous frequency:  $f_c + d_n \Delta f$

Carrier frequency:  $f_c$

# Non-Linear Digital Modulation Methods: Frequency Shift Keying (FSK)



continuous phase modulation (CPM)

Rectangular frequency pulses:

$$nT_s \leq t \leq (n+1)T_s:$$

$$s(t) = \sqrt{2} \operatorname{Re} \left\{ s_{BB}(t) e^{j2\pi f_c t} \right\} = \sqrt{2} A \cos [2\pi f_c t + 2\pi \Delta f (t - nT_s) d_n + \varphi(nT_s)]$$

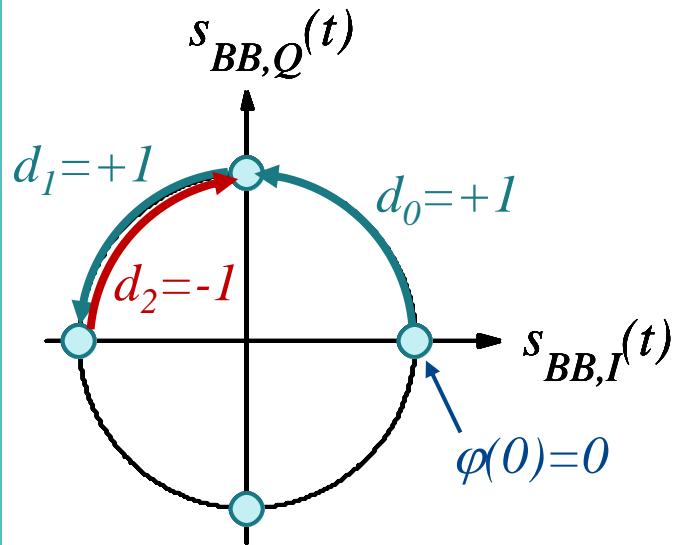
frequency separation:  $2\Delta f$

instantaneous frequency:  $f_c + d_n \Delta f$

$$s_{BB}(t) = A e^{j\varphi(t)} \quad |s_{BB}(t)| = A = \text{const}$$

$$\varphi(t) = 2\pi \Delta f (t - nT_s) d_n + \varphi(nT_s)$$

# Frequency Shift Keying (FSK): Modulation Index



Modulation index:

$$h = \frac{\text{maximum frequency separation}}{\text{symbol rate}} = \frac{2\Delta f}{1/T_s} = 2\Delta f T_s$$

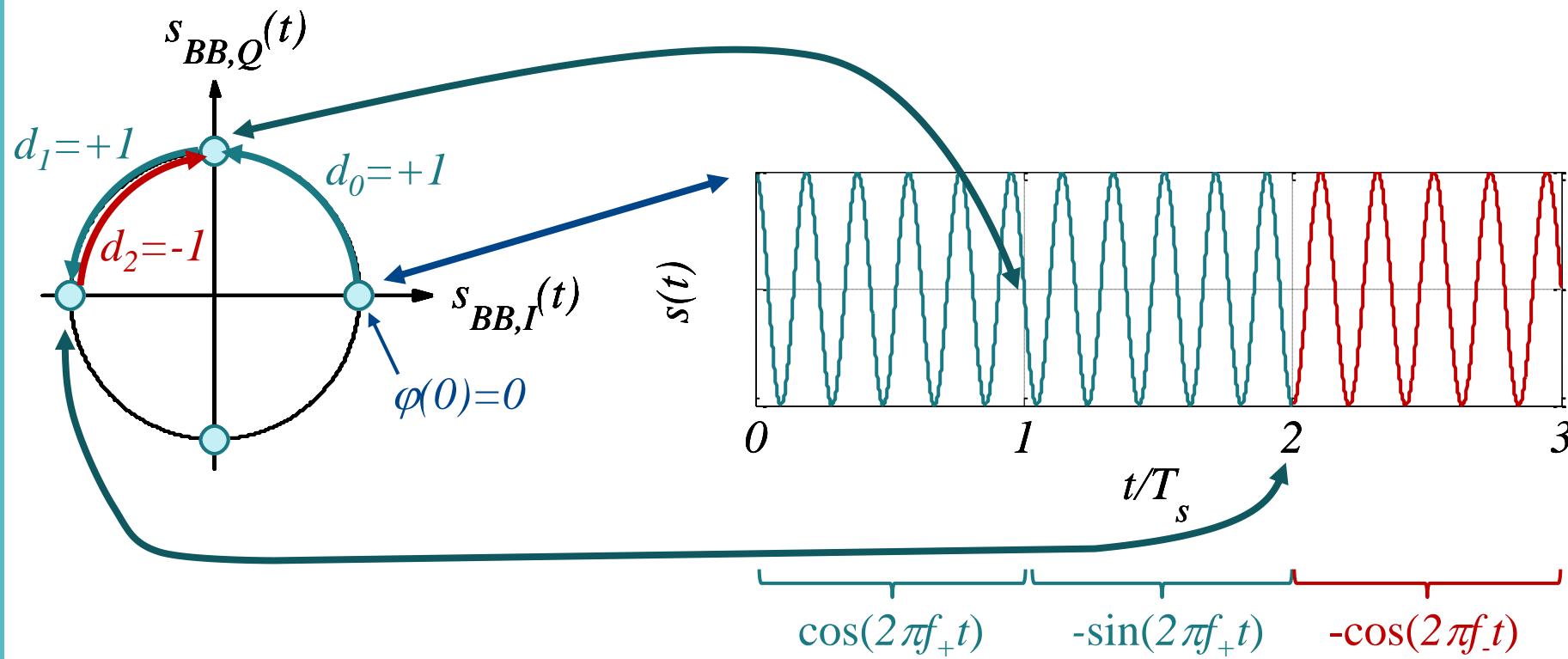
The phase of the equivalent baseband signal changes by  $\pi h \cdot d_n$  per  $T_s$ .

$$M = 2 \quad h = \frac{1}{2}$$

$$s_{BB}(t) = A e^{j\varphi(t)} \quad |s_{BB}(t)| = A = \text{const}$$

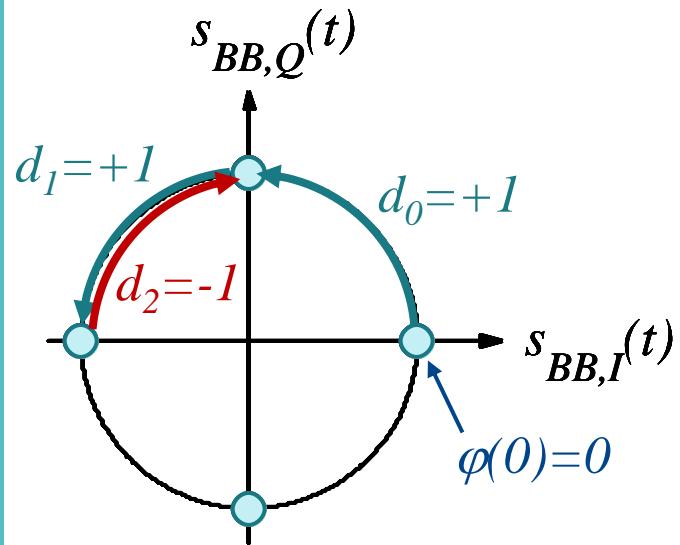
$$\varphi(t) = 2\pi\Delta f (t - nT_s) d_n + \varphi(nT_s) = \frac{\pi h}{T_s} (t - nT_s) d_n + \varphi(nT_s)$$

# Binary Frequency Shift Keying (FSK): Modulation Index $h=1/2$



→ essentially similar to QPSK

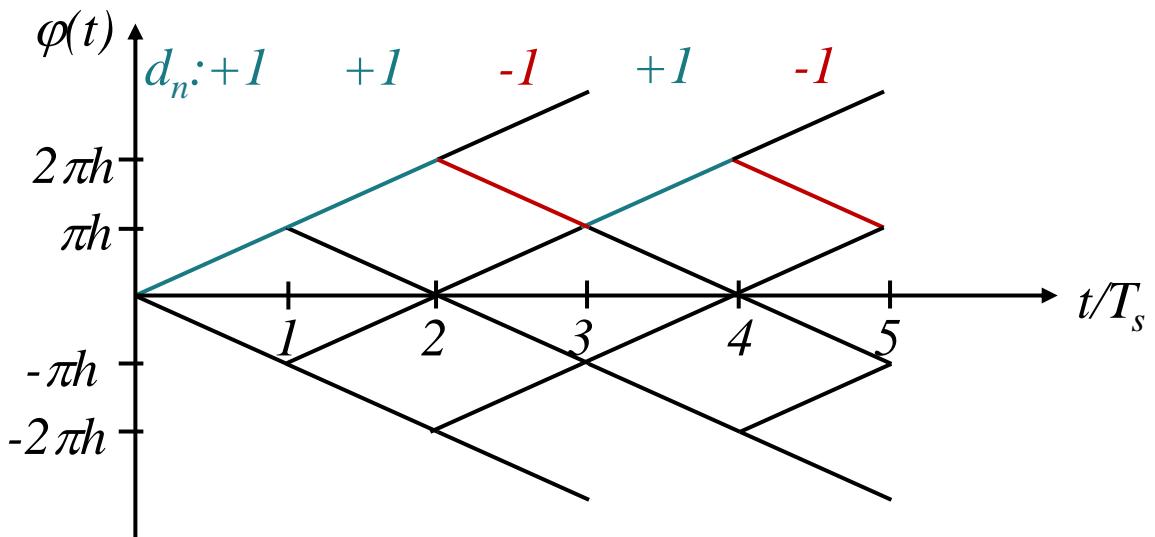
but: continuous phase !



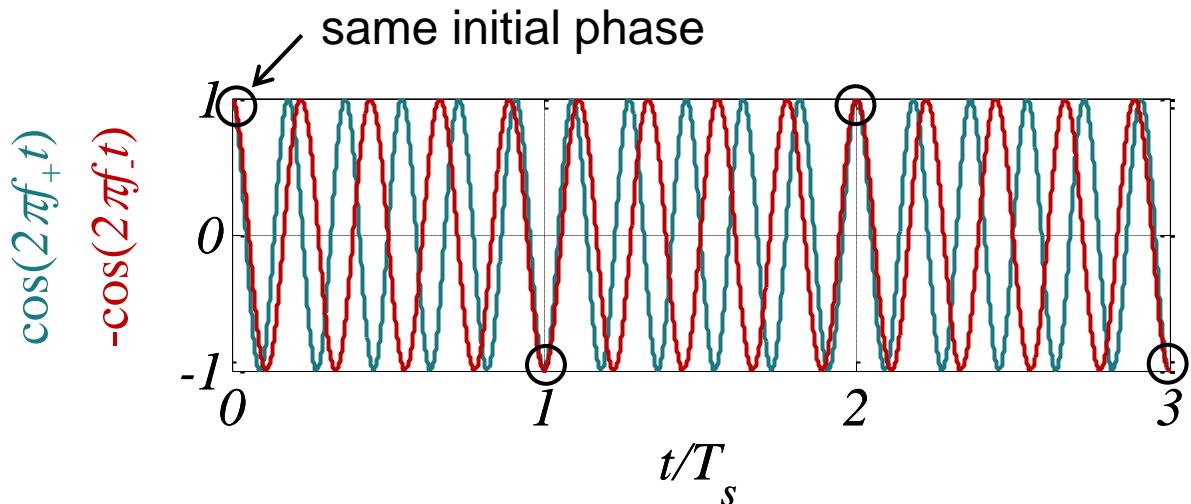
## Modulation index:

$$h = \frac{\text{maximum frequency separation}}{\text{symbol rate}} = \frac{2\Delta f}{1/T_s} = 2\Delta f T_s$$

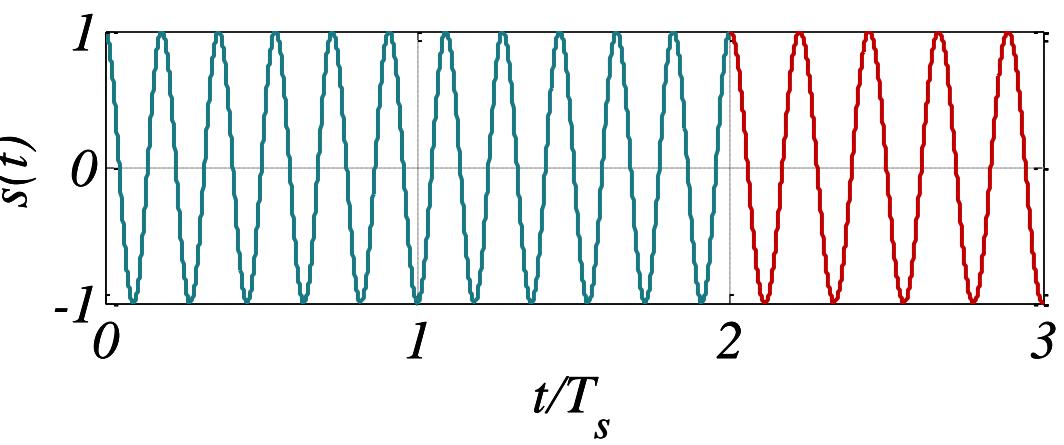
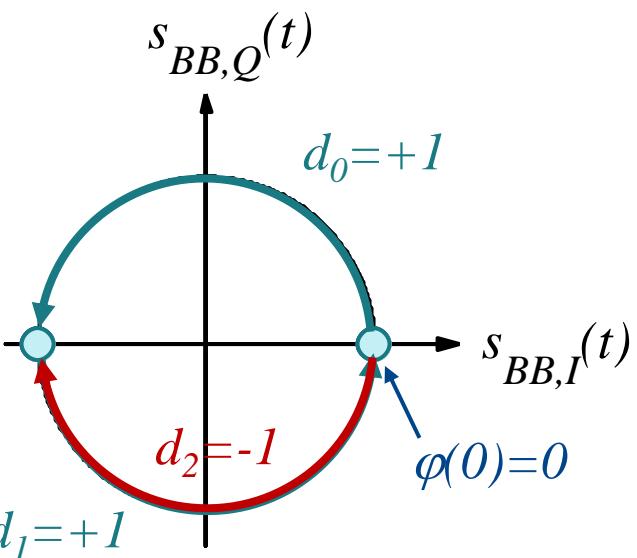
The phase  $\varphi(t)$  of the equivalent baseband signal changes by  $\pi h \cdot d_n$  per  $T_s$ .



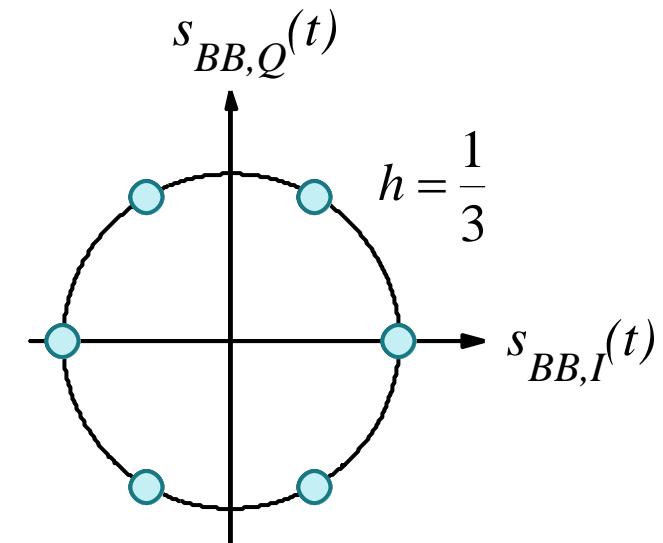
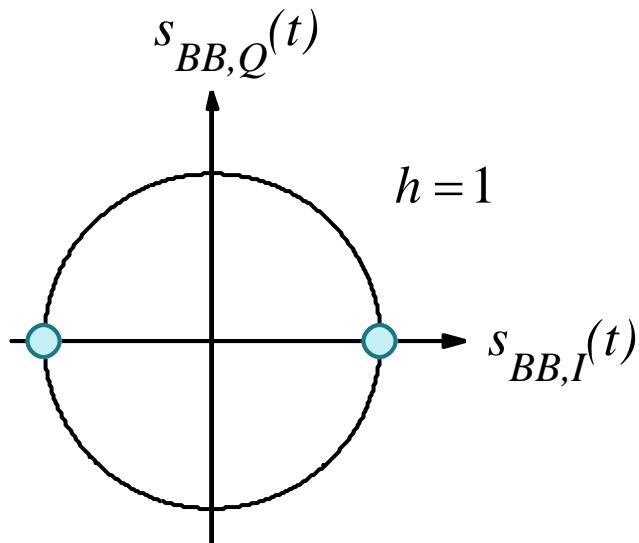
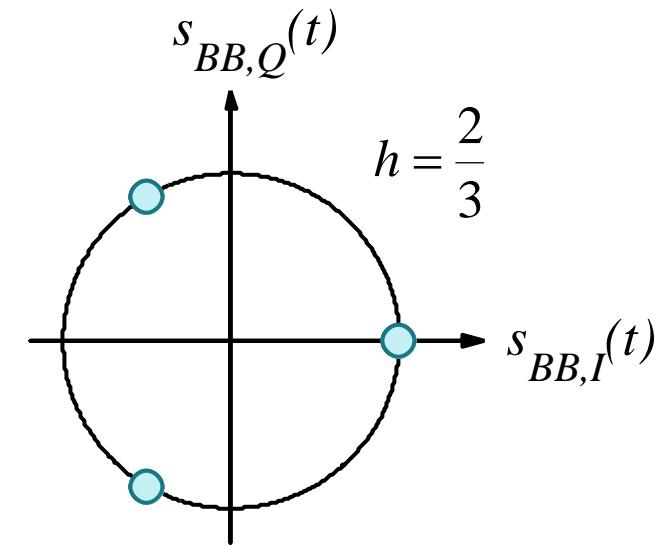
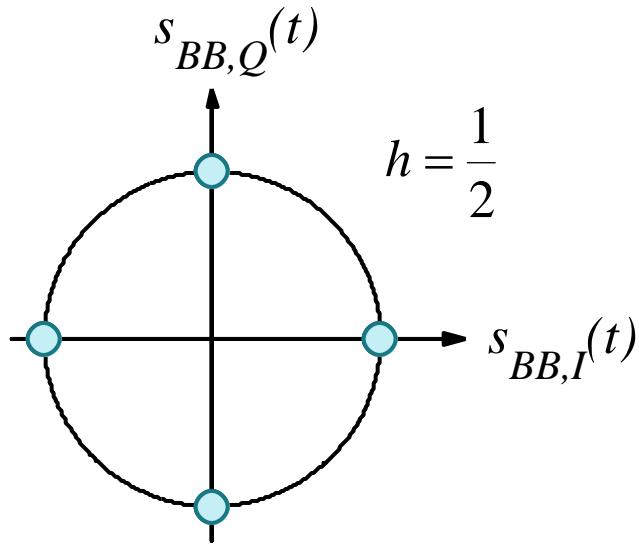
# Frequency Shift Keying (FSK): Modulation Index $h=1$



For  $h \in \mathbb{N}$ , switching between cos oscillations at frequencies  $f_+$  and  $f_-$  automatically ensures continuous phase.



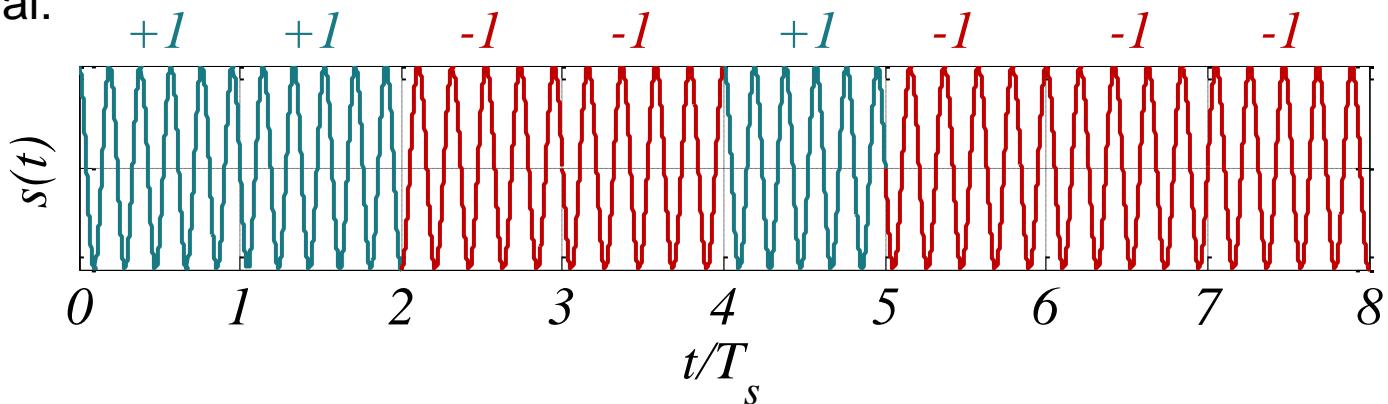
# Binary Frequency Shift Keying (FSK): Modulation Index



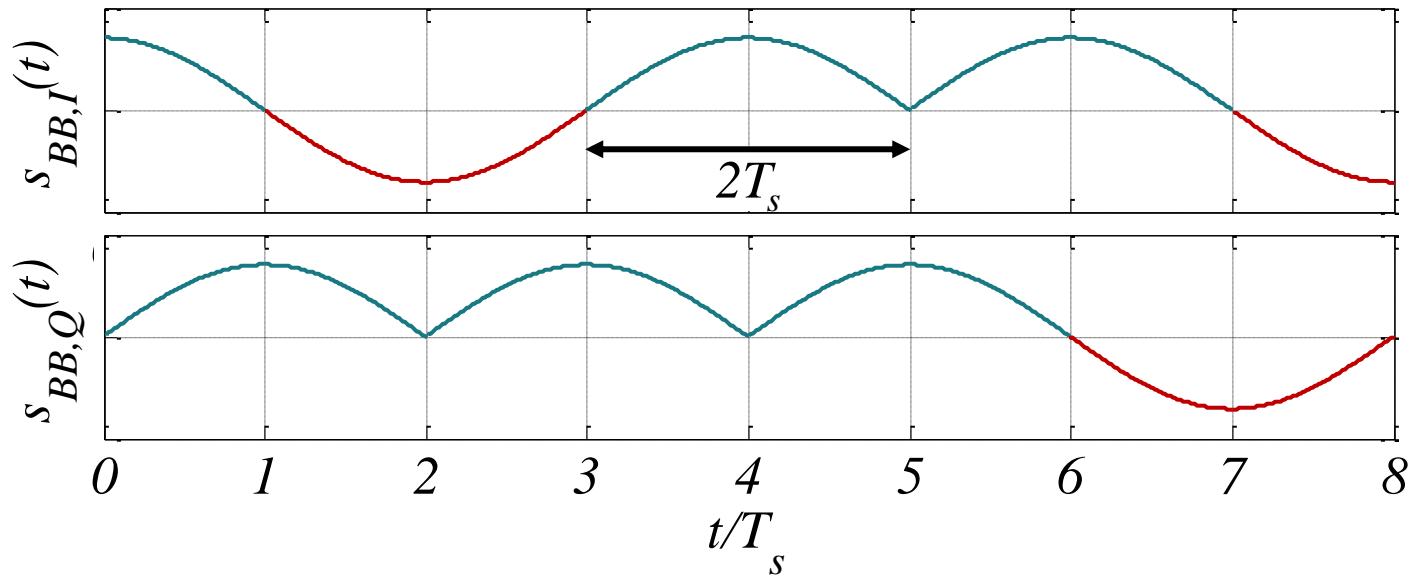
# Minimum Shift Keying (MSK): $h=1/2$

MSK: 2-FSK with modulation index  $h=1/2$  and rectangular frequency pulse.

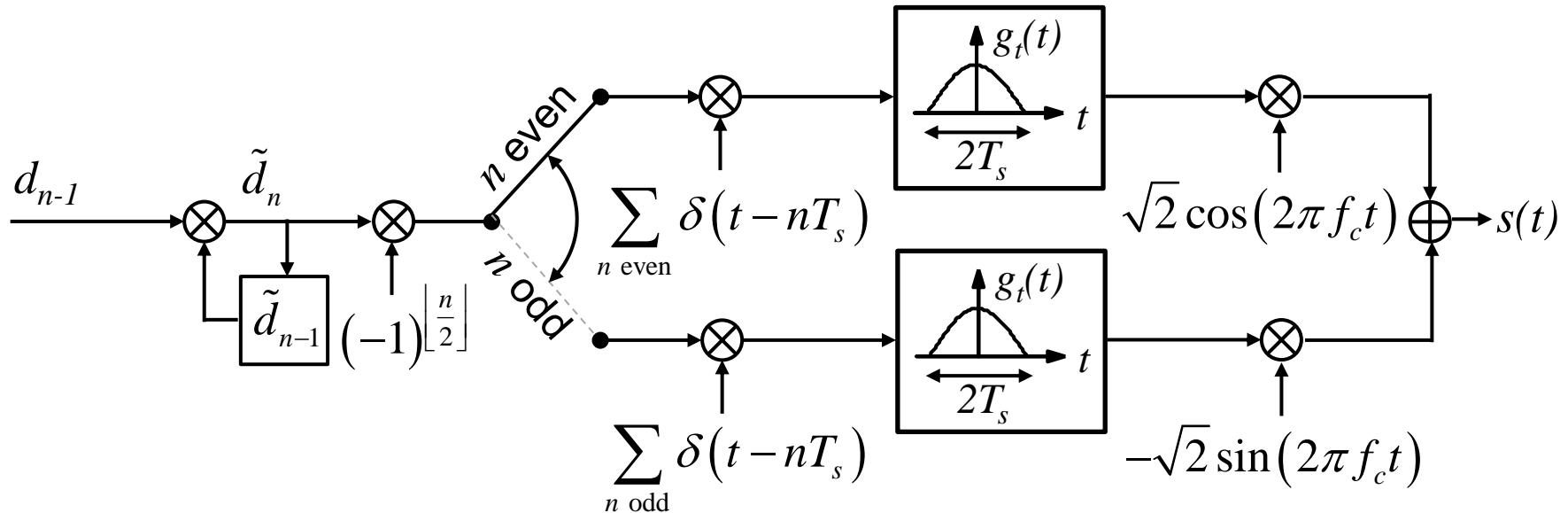
Bandpass signal:



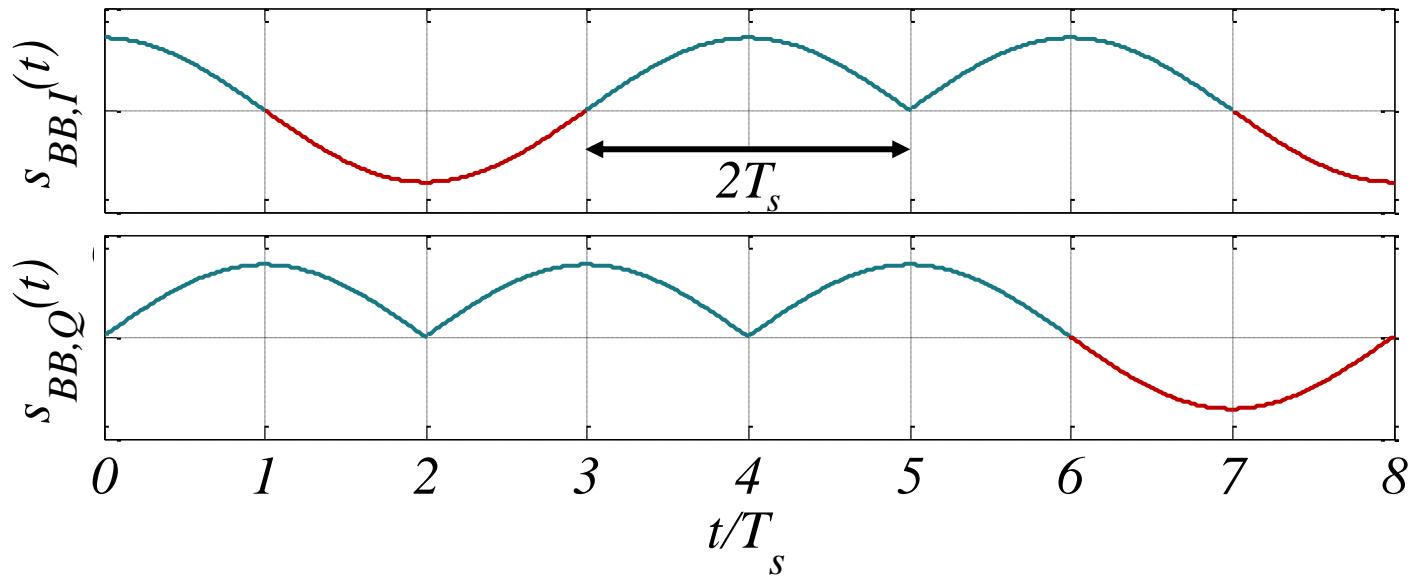
Equivalent baseband signal:



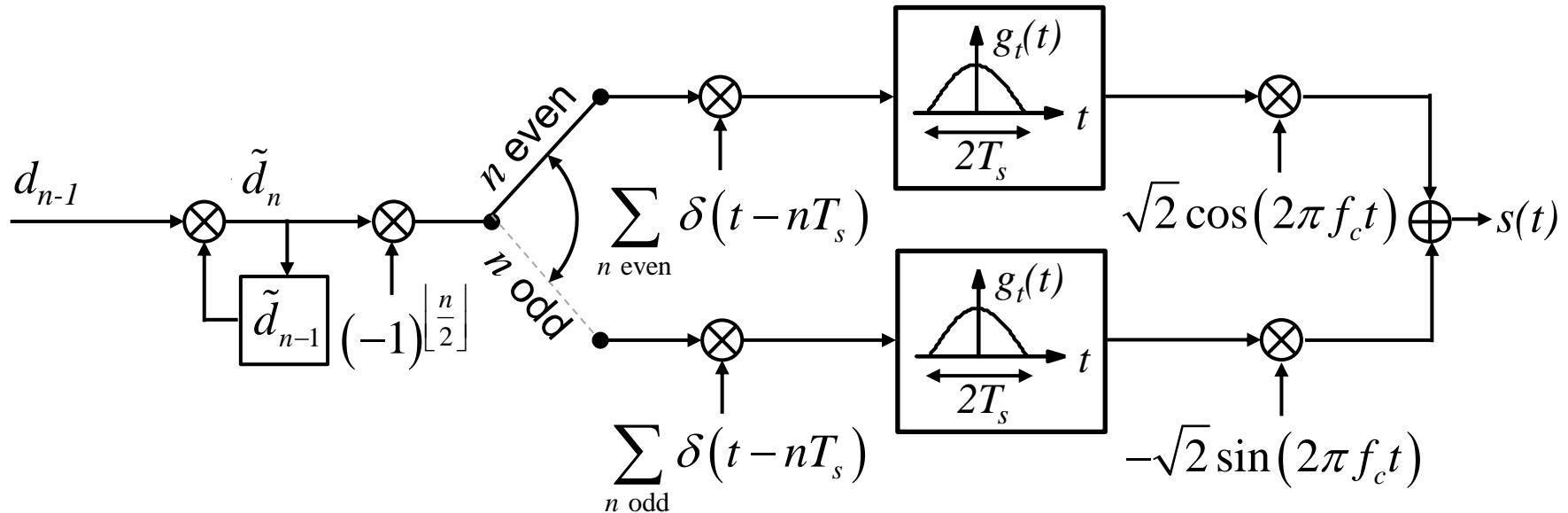
# Minimum Shift Keying (MSK): $h=1/2$ (1)



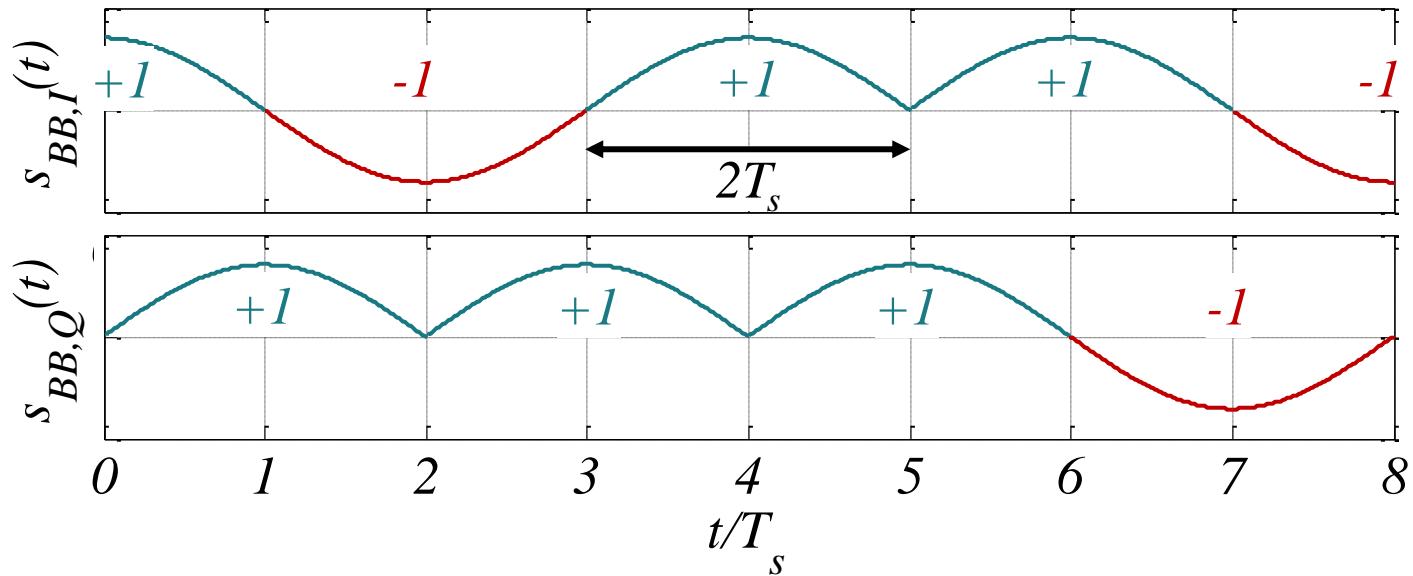
Equivalent baseband signal:



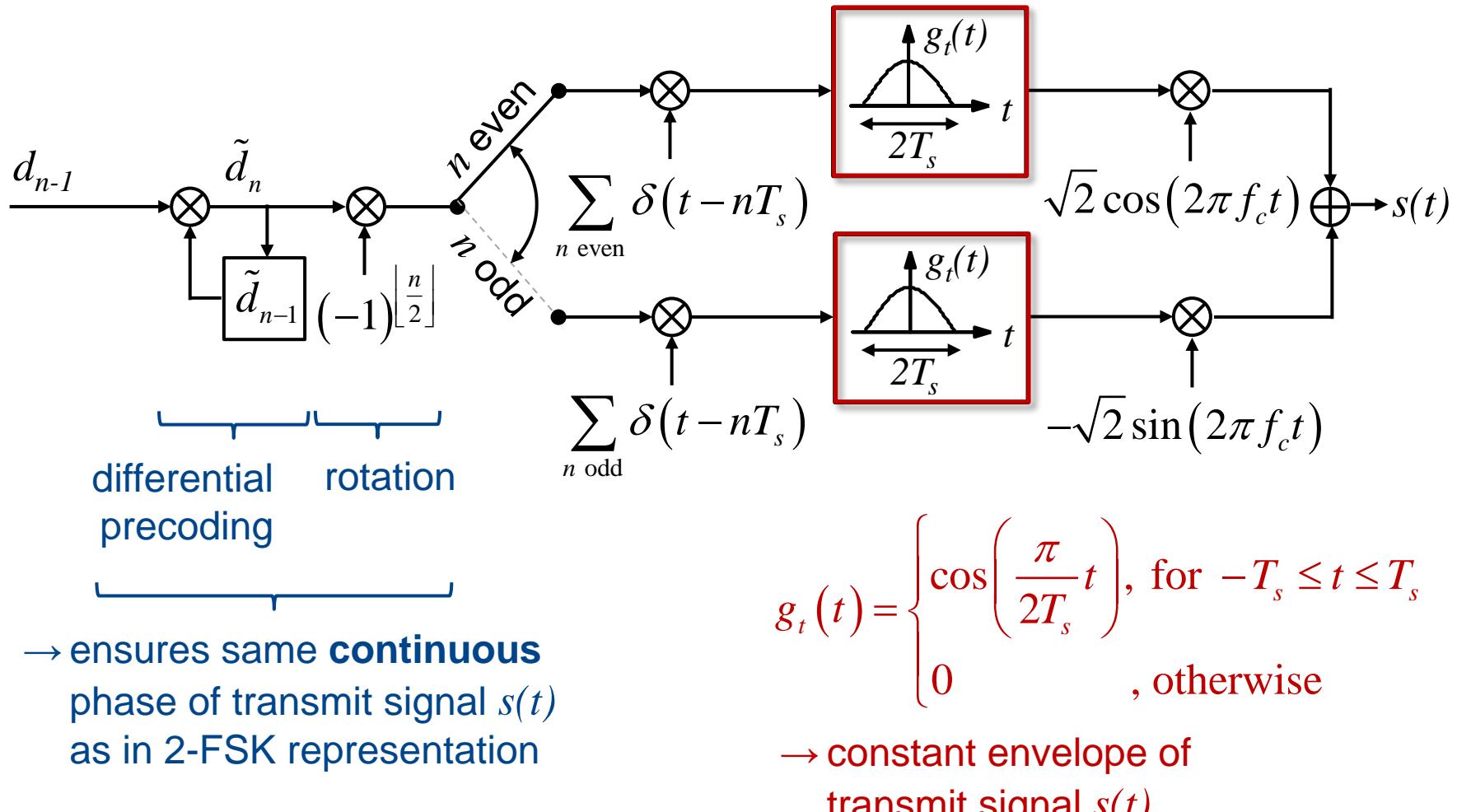
# Minimum Shift Keying (MSK): $h=1/2$ (2)



Equivalent baseband signal:



# Minimum Shift Keying (MSK): $h=1/2$ (3)

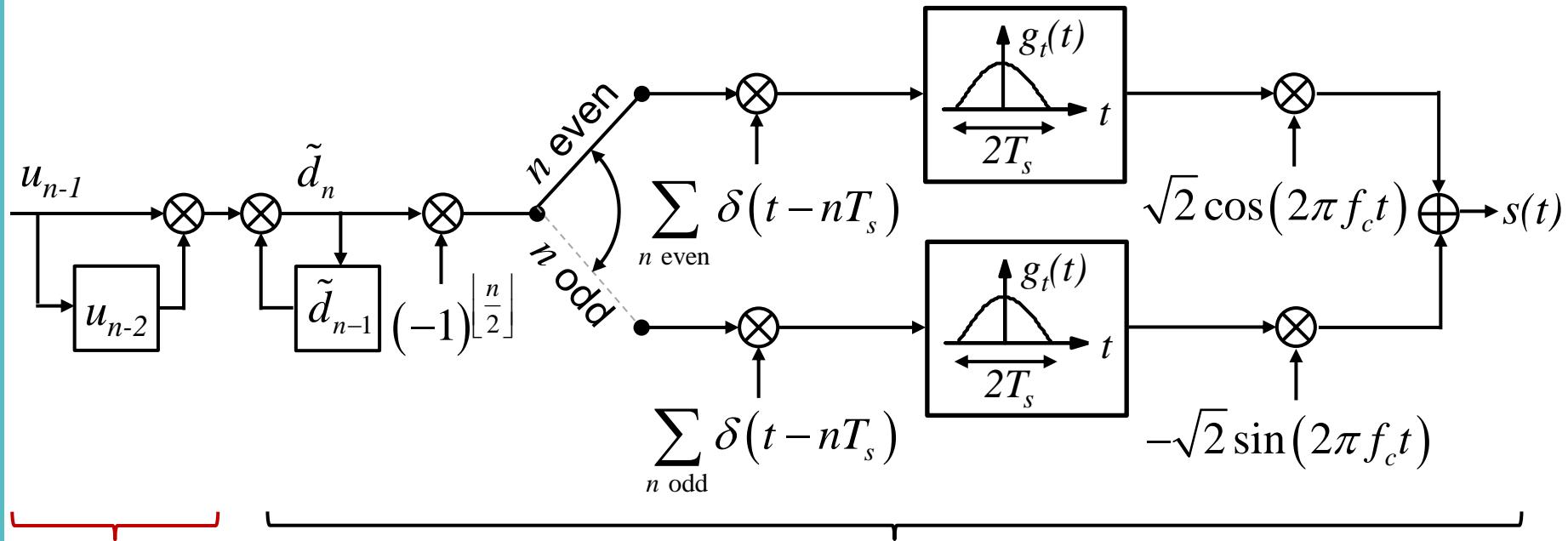


$$g_t(t) = \begin{cases} \cos\left(\frac{\pi}{2T_s}t\right), & \text{for } -T_s \leq t \leq T_s \\ 0 & , \text{ otherwise} \end{cases}$$

→ constant envelope of transmit signal  $s(t)$ .

MSK can be represented as Offset-QPSK with symbol duration  $2T_s$ , cosine pulses, differential precoding and rotation.

# Bit Error Probability of MSK



differential precoding

→ in order to avoid differential decoding at receiver

MSK modulator

$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

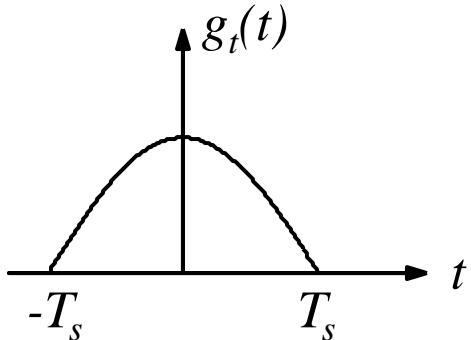
Without differential precoding:

$$P_b \approx \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)$$

due to double errors when differential decoding is done at receiver.

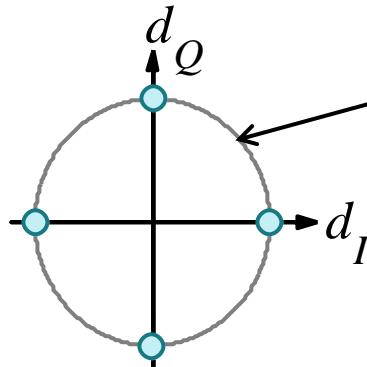
# Proof: Offset-QPSK with cos-Pulses Yields Constant Envelope of Transmit Signal (1)

MSK:  $T_s \rightarrow 2T_s$



$$g_t(t) = \begin{cases} \cos\left(\frac{\pi}{2T_s}t\right), & \text{for } -T_s \leq t \leq T_s \\ 0 & \text{otherwise} \end{cases}$$

$$= \cos\left(\frac{\pi}{2T_s}t\right) \cdot \text{rect}\left(\frac{t}{2T_s}\right)$$



Show:  
 $|s_{BB}(t)| = \text{const}$   
 $\rightarrow$  constant envelope of  
transmit signal  $s(t)$ .

$$d_{I,n}, d_{Q,n} \in \{\pm 1\}$$

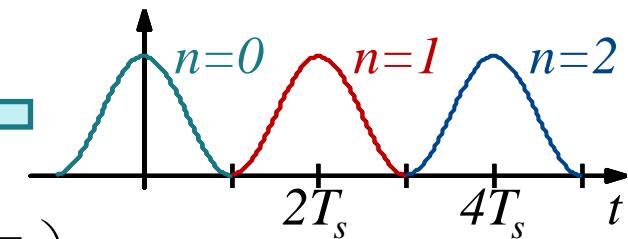
# Proof: Offset-QPSK with cos-Pulses Yields Constant Envelope of Transmit Signal (2)

$$s_{BB}(t) = \sum_n d_{I,n} g_t(t - 2nT_s) + j \sum_n d_{Q,n} g_t(t - T_s - 2nT_s)$$

$$|s_{BB}(t)|^2 = \sum_n \underbrace{d_{I,n}^2 g_t^2(t - 2nT_s)}_I + \sum_n \underbrace{d_{Q,n}^2 g_t^2(t - T_s - 2nT_s)}_I$$

duration of  $g_t(t)$  is  $2T_s \rightarrow$  no overlap of symbols within a quadrature component

$$\begin{aligned} |s_{BB}(t)|^2 &= \sum_n \cos^2\left(\frac{\pi(t - 2nT_s)}{2T_s}\right) \cdot \text{rect}\left(\frac{t - 2nT_s}{2T_s}\right) \\ &\quad + \sum_n \cos^2\left(\frac{\pi(t - T_s - 2nT_s)}{2T_s}\right) \cdot \text{rect}\left(\frac{t - T_s - 2nT_s}{2T_s}\right) \\ &= \cos^2\left(\frac{\pi t}{2T_s}\right) + \cos^2\left(\frac{\pi t}{2T_s} - \frac{\pi}{2}\right) = \cos^2\left(\frac{\pi t}{2T_s}\right) + \sin^2\left(\frac{\pi t}{2T_s}\right) \\ &\quad \uparrow \\ &\quad \cos\left(x - \frac{\pi}{2}\right) = +\sin(x) \end{aligned}$$



$$|s_{BB}(t)|^2 = 1$$

# Proof: Differential Precoding in Offset-QPSK Representation of MSK (1)

Equivalent baseband signal in interval  $nT_s \leq t \leq (n+1)T_s$ :

$$s_{BB}(t) = Ae^{j\varphi(t)} = e^{j\left[\frac{\pi h}{T_s}(t-nT_s)d_n + \varphi(nT_s)\right]} = e^{j\left[\frac{\pi h}{T_s}(t-nT_s)d_n + \pi h \sum_{m=0}^{n-1} d_m + \varphi(0)\right]} = e^{j\frac{\pi}{2}\left[\frac{1}{T_s}(t-nT_s)d_n + \sum_{m=0}^{n-1} d_m\right]}$$

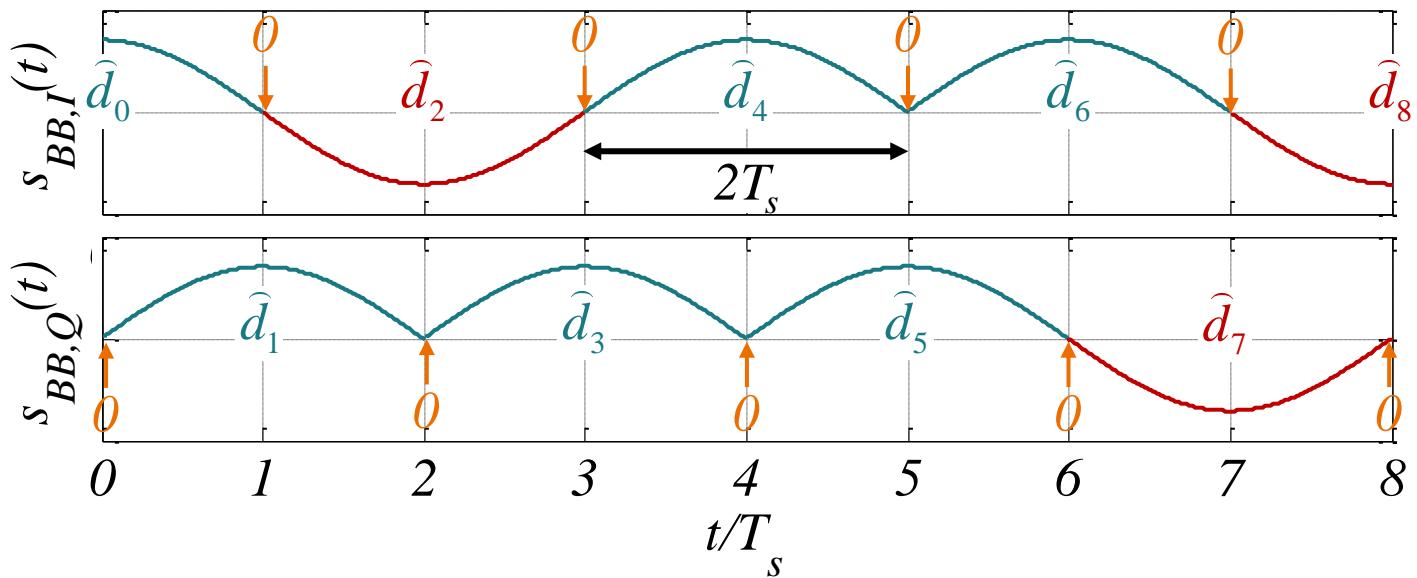
start phase:  $\varphi(0)$

MSK:  $h=1/2$

$$s_{BB}((n+1)T_s) = e^{j\frac{\pi}{2}\sum_{m=0}^n d_m} = \hat{d}_{n+1}$$

$d_n \in \{\pm 1\}$

1. Nyquist condition is met in Offset-QPSK with respect to  $2T_s$



# Proof: Differential Precoding in Offset-QPSK Representation of MSK (2)

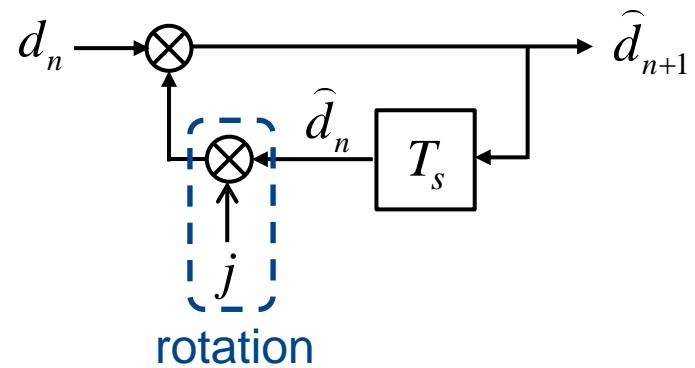
$$s_{BB}((n+1)T_s) = e^{j\frac{\pi}{2}\sum_{m=0}^n d_m} = \hat{d}_{n+1} = \underbrace{e^{j\frac{\pi}{2}d_n}}_{\mathbf{j}^d_n = \mathbf{j}d_n} e^{j\frac{\pi}{2}\sum_{m=0}^{n-1} d_m}$$

$$\hat{d}_{n+1} = jd_n \cdot \dots \cdot jd_2 \cdot jd_1 \cdot jd_0 \hat{d}_0 + 1$$

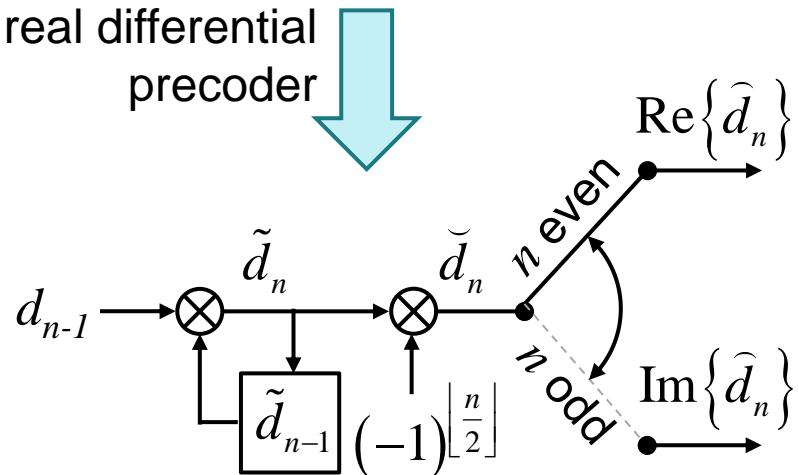
$$\hat{d}_n = j^{n+1} d_n \prod_{m=0}^{n-1} d_m \quad \tilde{d}_n$$

real differential precoder

Differential precoding in complex equivalent baseband representation:



real differential precoder



# Proof: Differential Precoding in Offset-QPSK Representation of MSK (3)

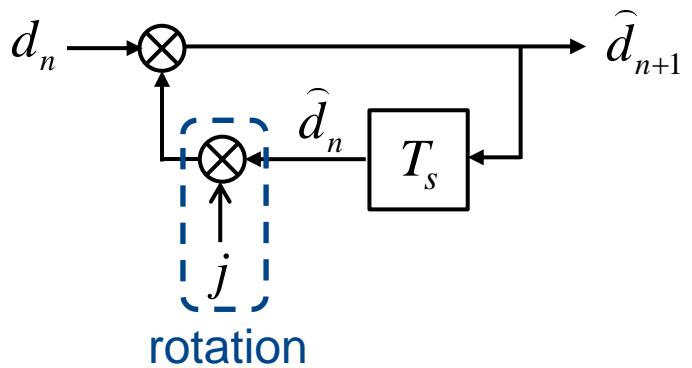
$$s_{BB}((n+1)T_s) = e^{j\frac{\pi}{2}\sum_{m=0}^n d_m} = \hat{d}_{n+1} = \underbrace{e^{j\frac{\pi}{2}d_n}}_{\mathbf{j}^{d_n}} \underbrace{e^{j\frac{\pi}{2}\sum_{m=0}^{n-1} d_m}}_{\hat{d}_n}$$

$$\hat{d}_{n+1} = j d_n \cdot \dots \cdot j d_2 \cdot j d_1 \cdot j d_0 \underbrace{\hat{d}_0}_{+1}$$

$$\hat{d}_n = j^{n+1} d_n \prod_{m=0}^{n-1} d_m \underbrace{\tilde{d}_n}_{\tilde{d}_n}$$

real differential precoder

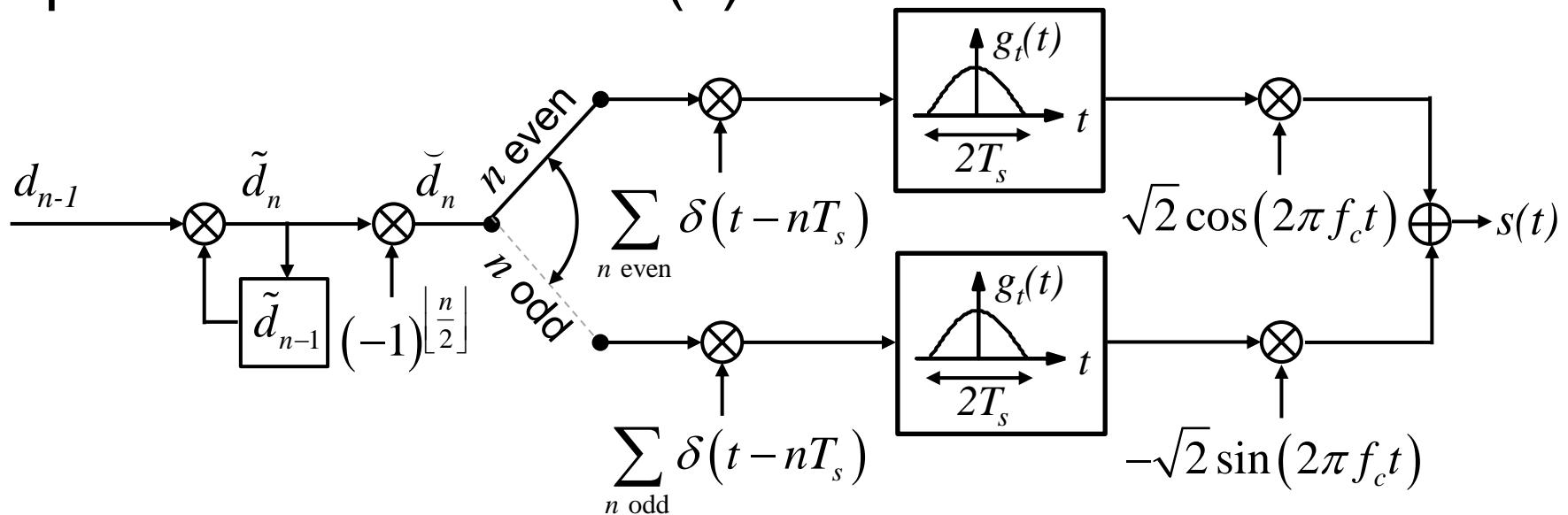
Differential precoding in complex equivalent baseband representation:



$\check{d}_n = (-1)^{\left\lfloor \frac{n}{2} \right\rfloor} d_{n-1} \tilde{d}_{n-1}; \quad d_{-1} = \tilde{d}_{-1} = +1$

rotation differential precoder

# Proof: Differential Precoding in Offset-QPSK Representation of MSK (4)

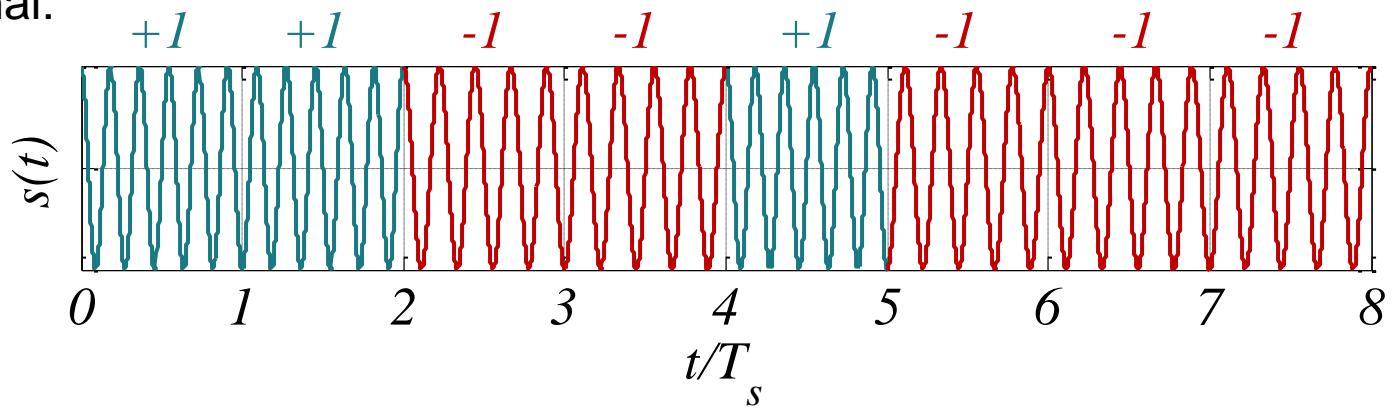


$$\tilde{d}_n = (-1)^{\left\lfloor \frac{n}{2} \right\rfloor} d_{n-1} \tilde{d}_{n-1}; \quad d_{-1} = \tilde{d}_{-1} = +1$$

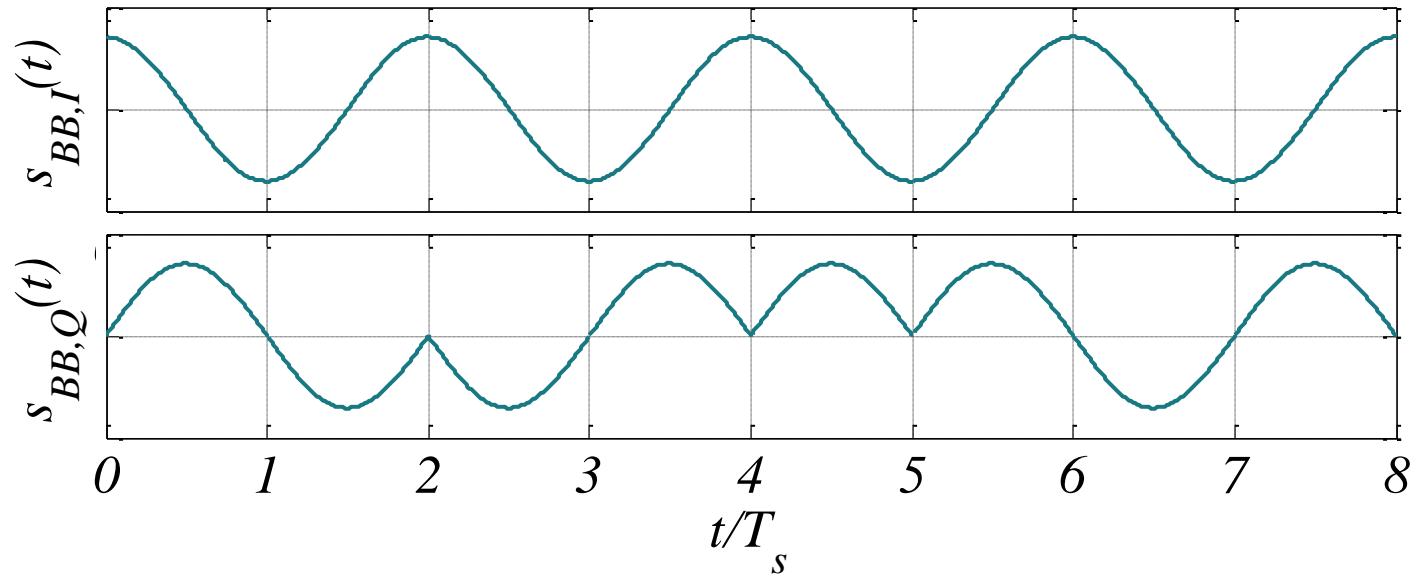
rotation differential  
precoder

# Orthogonal 2-FSK with Modulation Index $h=1$

Bandpass signal:

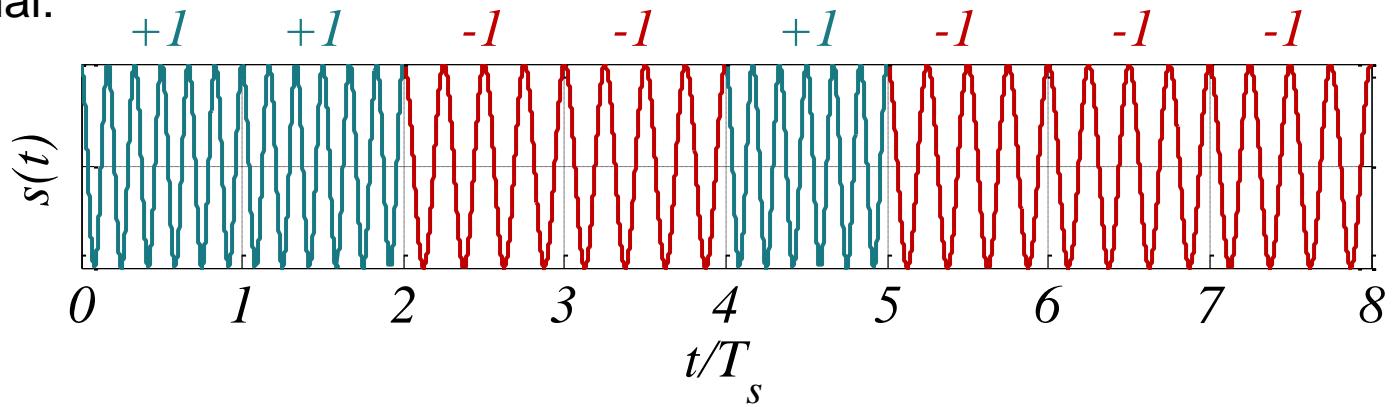


Equivalent baseband signal:

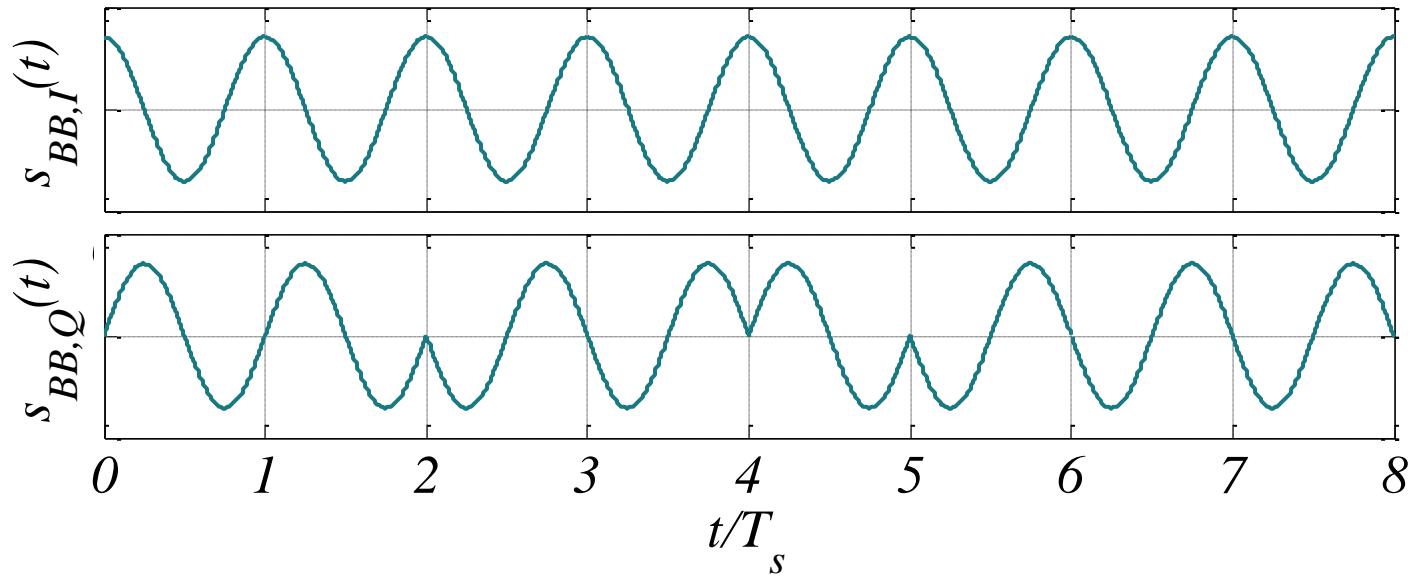


# Orthogonal 2-FSK with Modulation Index $h=2$

Bandpass signal:



Equivalent baseband signal:



# Correlation Coefficient of Elementary Signals for Binary Frequency Shift Keying (2-FSK) (1)

The waveforms transmitted for different data symbols  $d_n$  should be as unlikely as possible in order to allow good separation at the receiver and, hence, detection with low error probability. The correlation coefficient

$$\rho_{\mu\nu} = \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}^{*}(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_{\mu}^2(t) dt \cdot \int_{-\infty}^{\infty} s_{\nu}^2(t) dt}}$$

among possible transmit waveforms  $s_{\mu}(t)$  and  $s_{\nu}(t)$  is, therefore, a suitable measure for evaluating a modulation scheme. This is also motivated by the fact that a correlation receiver will correlate the received signal with all possible transmit waveforms and decide for the symbol corresponding to the waveform with highest correlation. Consequently, waveforms corresponding to different data symbols should have low correlation. Ideally, they are antipodal, i.e.  $\rho_{\mu\nu} = -1$ .

For binary FSK (2-FSK) with rectangular full response frequency pulses, there are two possible transmit waveforms: For a data symbol  $d_n = +1$ , a cosine oscillation with frequency  $f_1$  will be transmitted during a symbol period, i.e.  $s_{+1}(t) = \cos(2\pi f_1 t)$  for  $0 \leq t \leq T_s$ . For  $d_n = -1$ , a cosine oscillation with frequency  $f_2$  will be transmitted during a symbol period, i.e.  $s_{-1}(t) = \cos(2\pi f_2 t)$  for  $0 \leq t \leq T_s$ .

# Correlation Coefficient of Elementary Signals for Binary Frequency Shift Keying (2-FSK) (2)

It can be shown that the correlation coefficient of the elementary signals  $s_{+1}(t)$  and  $s_{-1}(t)$  is given by

$$\rho_{+-} = \frac{\sin(2\pi h)}{2\pi h},$$

where  $h$  is the modulation index resulting from the instantaneous frequencies  $f_1$  and  $f_2$ , i.e.

$$h = 2\Delta f T_s = (f_1 - f_2) T_s.$$

Unfortunately, antipodal elementary signals cannot be achieved with 2-FSK, as the minimum of the correlation coefficient is  $\rho_{+-} = -0.22 > -1$ . This minimum is obtained for a modulation index  $h=0.72$ . Consequently, for optimum performance in terms of bit error probability, the modulation index  $h=0.72$  should be used. However, it is often more important to reduce the bandwidth of the transmit signal, which can be achieved with smaller modulation index  $h<0.72$ .

For modulation indices which are multiples of  $1/2$ , i.e.  $h=k \cdot 1/2$ ,  $k=1, 2, \dots$ , the correlation coefficient becomes  $\rho_{+-} = 0$ , i.e. the elementary signals are orthogonal. Even though orthogonal elementary signals are not optimum in terms of bit error rate performance, they allow for a computationally efficient implementation of the detector as will be explained for orthogonal 2-FSK with modulation index  $h=1/2$  (Minimum Shift Keying (MSK)).

# Minimum Shift Keying (MSK) (1)

For a modulation index  $h=1/2$  and a rectangular full response frequency pulse, 2-FSK can be represented as a linear modulation scheme. More precisely, it is equivalent to offset-QPSK with

- effective symbol duration  $2T_s$ ,
- cosine pulses which ensure constant envelope of the transmit signal  $s(t)$ :

$$g_t(t) = \cos\left(\frac{\pi}{2T_s}t\right) \cdot \text{rect}\left(\frac{t}{2T_s}\right) \quad \bullet \quad G_t(f) = \frac{4T_s}{\pi} \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2}$$

and

- differential precoding and rotation.

Consequently, it can be demodulated in the same way as offset QPSK using a QAM demodulator and a matched filter receiver. The differential precoding and the rotation have to be reversed at the output of the matched filter.

Moreover, the average power spectral density can be determined in exactly the same way as for linear modulation schemes and is given by

$$\overline{S}_{S_{BB}S_{BB}}(f) = \frac{1}{T_s} E\left\{ |d_n|^2 \right\} |G_t(f)|^2 = \frac{16T_s^2}{\pi^2} \left( \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2} \right)^2.$$

# Minimum Shift Keying (MSK) (2)

As  $h=1/2$  is the smallest modulation index which yields orthogonal 2-FSK elementary signals, this special version of 2-FSK is called *Minimum Shift Keying (MSK)*. MSK is the only known FSK which can be represented as a linear modulation scheme.

When MSK is demodulated as linear modulation scheme using an offset-QPSK matched filter receiver, the bit error probability at the matched filter output before the differential decoding is given by

$$P_{b,\text{OQPSK}} = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_{s,\text{OQPSK}}}{2N_0}} \right),$$

where  $E_{s,\text{OQPSK}}$  is the energy per symbol of the equivalent offset-QPSK. As two bits  $\bar{d}_n$  are mapped to an offset-QPSK symbol  $d_{\text{OQPSK}} = \bar{d}_{2n} + j\bar{d}_{2n+1}$ , the energy per MSK-bit is given by

$$E_b = \frac{1}{2} E_{s,\text{OQPSK}},$$

which can be plug into the equation of the bit error probability yielding

$$P_{b,\text{OQPSK}} = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right).$$

# Minimum Shift Keying (MSK) (3)

However, the differential decoding causes double errors. Consequently, the bit error probability of MSK is given by

$$P_{b,\text{MSK}} \approx 2P_{b,\text{OQPSK}} = 2 \cdot \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right) = \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right).$$

Note, that even though MSK can be represented as offset-QPSK, MSK is a binary modulation scheme. This is reflected in the effective symbol duration of  $2T_s$  in the offset-QPSK modulator. In order to avoid the penalty of a factor 2 in the bit error probability caused by the differential detector, sometimes a differential precoding of the data according to  $d_n = u_n u_{n-1}$  is done before the MSK modulator. This reverses the effect of the MSK-inherent differential precoding already at the transmitter where no errors occur since the data is known at the transmitter. Hence, no differential detection but only the memoryless derotation has to be performed at the receiver and, hence, double errors are avoided. As a result, differentially precoded MSK achieves the same bit error rate as BPSK which is given by

$$P_b = \frac{1}{2} \operatorname{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right).$$

# Gaussian Minimum Shift Keying (GMSK)

A slight modification of MSK called *Gaussian Minimum Shift Keying (GMSK)* is applied in the GSM system: Instead of using a rectangular frequency pulse, a Gaussian-like frequency pulse is used. The GSM frequency pulse results from the convolution of a rectangular pulse and a Gaussian pulse followed by truncation of the convolution product to a length of  $5T_s$ . The Gaussian-like frequency pulse yields better spectral properties, i.e. lower bandwidth requirements compared to a rectangular frequency pulse.

Unfortunately, GMSK cannot be represented exactly as a linear modulation scheme. However, it can be well approximated as a linear modulation scheme. Therefore, most GSM modulators and detectors use this approximation in order to generate the transmit signal using a QAM modulator and a QAM demodulator at the receiver.

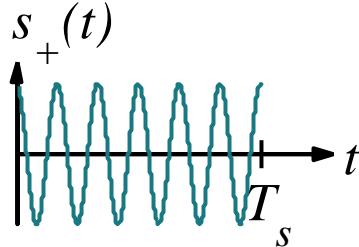
The partial response frequency pulses cause intersymbol interference which needs to be taken care of by an equalizer. The approximation of GMSK as linear modulation scheme allows to apply the computationally efficient Viterbi algorithm for equalization.

# Correlation Coefficient of Elementary Signals for Binary Frequency Shift Keying (2-FSK)

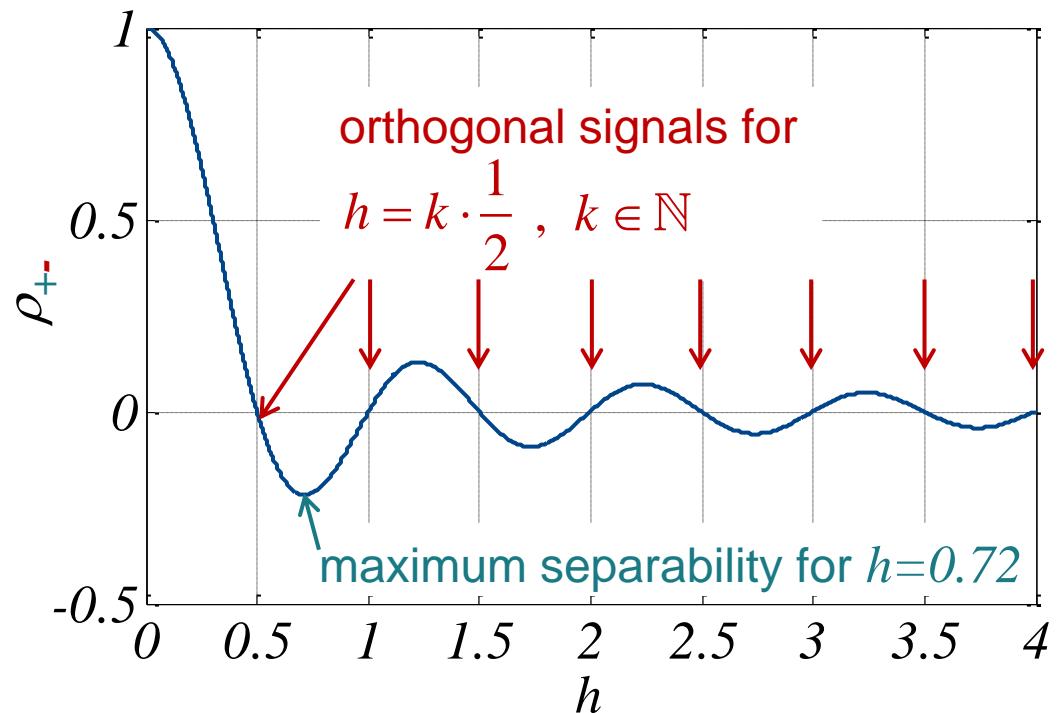
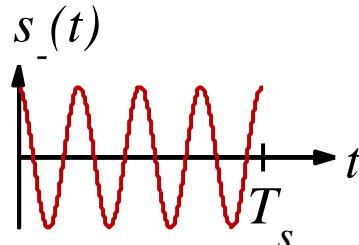
$$\rho_{\mu\nu} = \frac{\int_{-\infty}^{\infty} s_{\mu}(t) s_{\nu}^*(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_{\mu}^2(t) dt \cdot \int_{-\infty}^{\infty} s_{\nu}^2(t) dt}} = \frac{\sin(2\pi h)}{2\pi h}$$

Elementary signals:

$d=+1$ :



$d=-1$ :



# Derivation of Correlation Coefficient for Binary Frequency Shift Keying (2-FSK) (1)

$$\rho_{+-} = \frac{\int_{-\infty}^{\infty} s_+(t) s_-^*(t) dt}{\sqrt{\int_{-\infty}^{\infty} s_+^2(t) dt \cdot \int_{-\infty}^{\infty} s_-^2(t) dt}} = \frac{1}{\sqrt{E_+ E_-}} \cdot \text{Re} \left\{ \int_{-\infty}^{\infty} s_{BB,+}(t) s_{BB,-}^*(t) dt \right\}$$

see derivation for QAM

$$s_{BB,+}(t) = A \cdot e^{j \frac{\pi h}{T_s} t (+1)}, \quad 0 \leq t < T_s$$

$$s_{BB,-}(t) = A \cdot e^{j \frac{\pi h}{T_s} t (-1)}, \quad 0 \leq t < T_s$$

$$E_+ = E_- = \int_{-\infty}^{\infty} |s_{BB,+}(t)|^2 dt = \int_0^{T_s} A^2 dt = A^2 T_s$$

$$\rho_{+-} = \frac{1}{A^2 T_s} \cdot \text{Re} \left\{ \int_{-\infty}^{\infty} A \cdot e^{j \frac{\pi h}{T_s} t (+1)} \cdot A \cdot e^{-j \frac{\pi h}{T_s} t (-1)} dt \right\} = \frac{1}{T_s} \cdot \text{Re} \left\{ \int_{-\infty}^{\infty} e^{j 2 \frac{\pi h}{T_s} t} dt \right\}$$

# Derivation of Correlation Coefficient for Binary Frequency Shift Keying (2-FSK) (2)

$$\rho_{+-} = \frac{1}{T_s} \cdot \operatorname{Re} \left\{ \int_{-\infty}^{\infty} e^{j2\frac{\pi h}{T_s}t} dt \right\} = \frac{1}{T_s} \cdot \operatorname{Re} \left\{ \frac{T_s}{j2\pi h} \left[ e^{j2\frac{\pi h}{T_s}t} \right]_0^{T_s} \right\}$$

$$= \operatorname{Re} \left\{ \frac{1}{j2\pi h} \left[ e^{j2\pi h} - 1 \right] \right\}$$

$$= \operatorname{Re} \left\{ \frac{-j}{2\pi h} \left[ \cos(2\pi h) + j \sin(2\pi h) - 1 \right] \right\}$$

$$\rho_{+-} = \frac{\sin(2\pi h)}{2\pi h}$$

# General Continuous Phase Modulation (CPM)

continuous phase modulation (CPM)

$$\varphi(t) = 2\pi\Delta f T_s \int_{-\infty}^t \sum_{m=-\infty}^{\infty} d_m g_t(\tau - mT_s) d\tau + \varphi_0 = \pi h \sum_{m=-\infty}^{\infty} d_m \underbrace{\int_{-\infty}^t g_t(\tau - mT_s) d\tau}_{q_t(t-mT_s)} + \varphi_0$$

$q_t(t-mT_s)$ : phase impulse

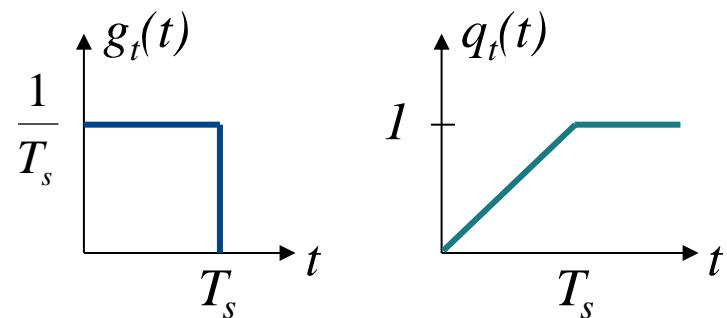
$$= \pi h \sum_{m=-\infty}^{\infty} d_m q(t - mT_s) + \varphi_0$$

**Example:** Rectangular frequency pulse  $g_t(t)$

$nT_s \leq t \leq (n+1)T_s \Rightarrow$  transmit  $d_n$

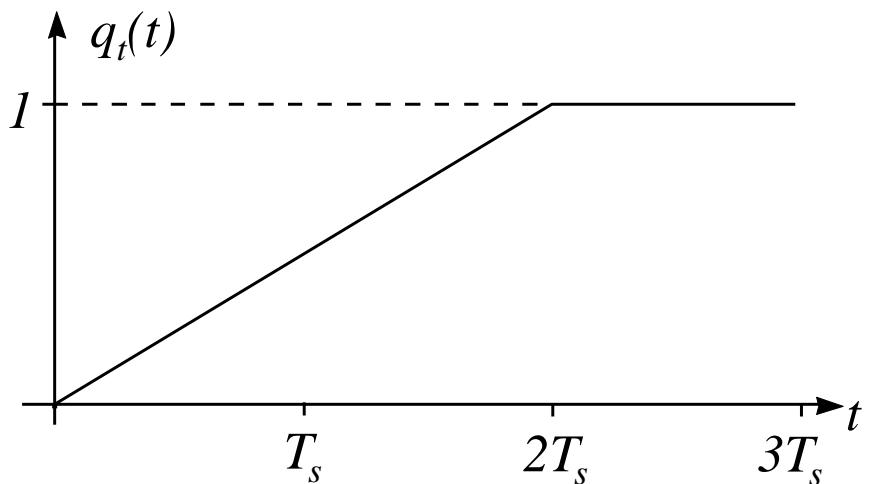
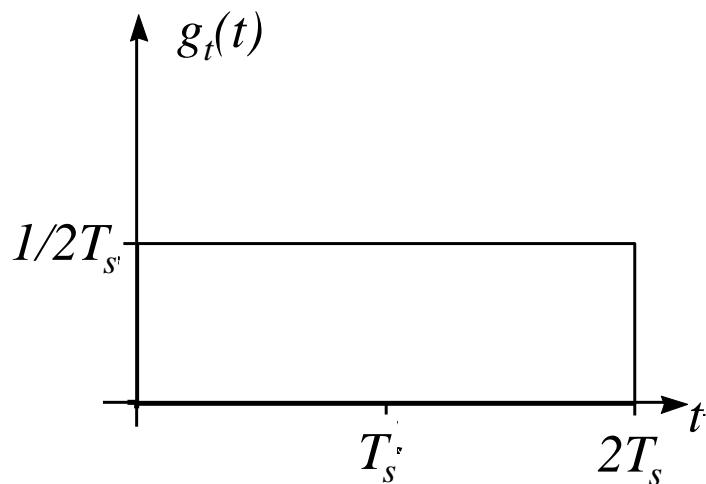
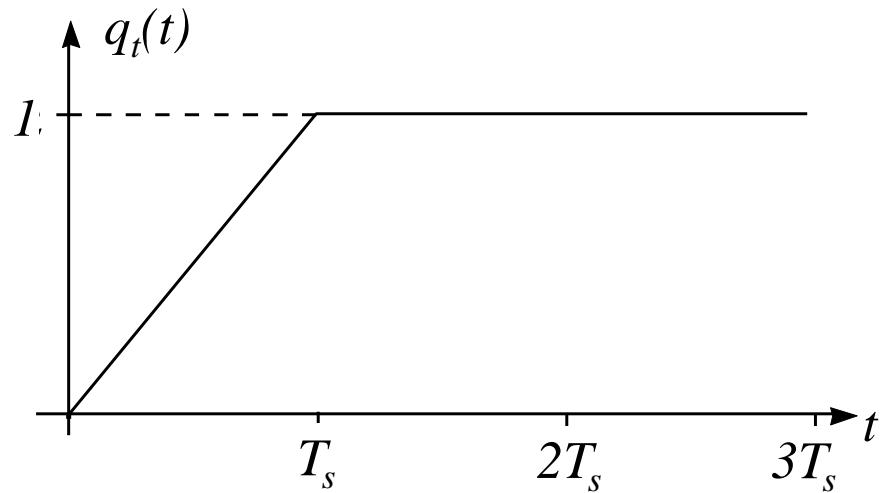
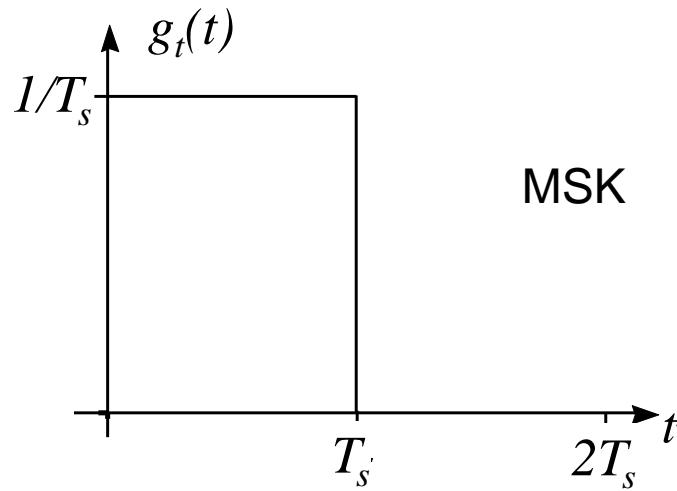
$$\varphi(t) = \pi h d_n \frac{1}{T_s} (t - nT_s) + \underbrace{\pi h \sum_{m=-\infty}^{n-1} d_m}_{\varphi(nT_s)} + \varphi_0$$

$$= \pi h d_n \left( \frac{t}{T_s} - n \right) + \varphi(nT_s)$$

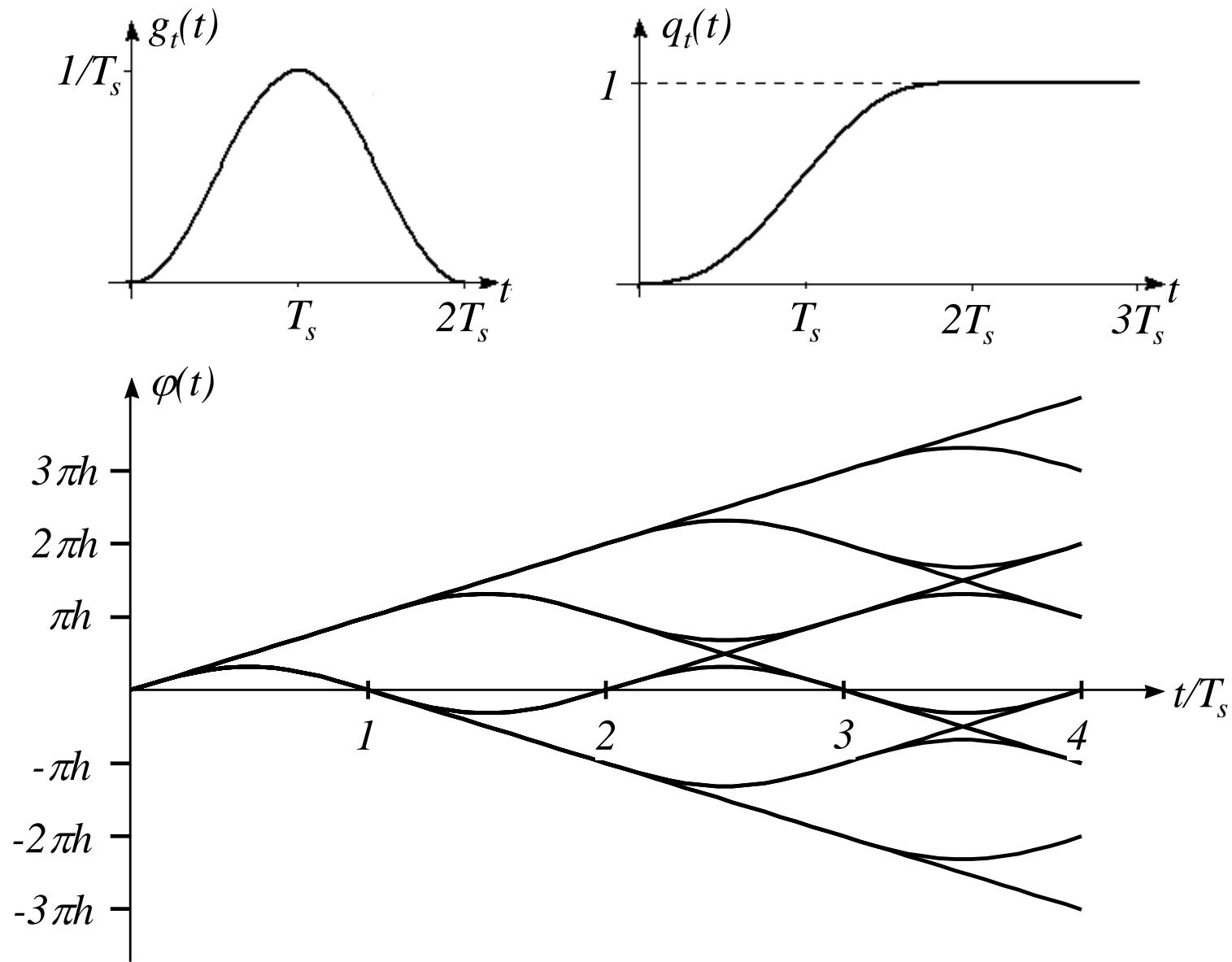


hard switching between two frequencies (frequency shift keying)

# Continuous Phase Modulation (CPM) – Rectangular Pulses



# Continuous Phase Modulation (CPM) – Raised Cosine (2RC) Pulses



# Continuous Phase Modulation (CPM) – Example

## Global System for Mobile Communication (GSM):

Gaussian Minimum Shift Keying (GMSK):

- binary FSK
- $g_t(t)$  is convolution of Gaussian and rectangular pulse (truncated to  $5T_s$ )
- $h = 0.5$
- $T_s \approx 4 \mu s$

$$\Delta f = \frac{h}{2T_s} = \frac{1/2}{2 \cdot 4 \mu s} = 0.06 \text{ MHz}$$

D-network:  $f_c = 900 \text{ MHz}$

E-network:  $f_c = 1800 \text{ MHz}$

instantaneous frequency:  $f_c \pm 0.06 \text{ MHz}$

# Power Spectral Density of MSK

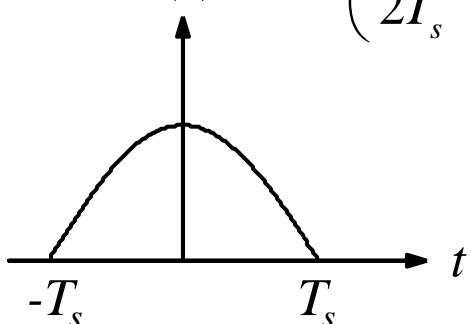
MSK can be viewed as offset-QPSK  
 → linear digital modulation scheme

Average power spectral density of equivalent baseband signal  $s_{BB}(t)$   
 for **linear** digital modulation schemes:

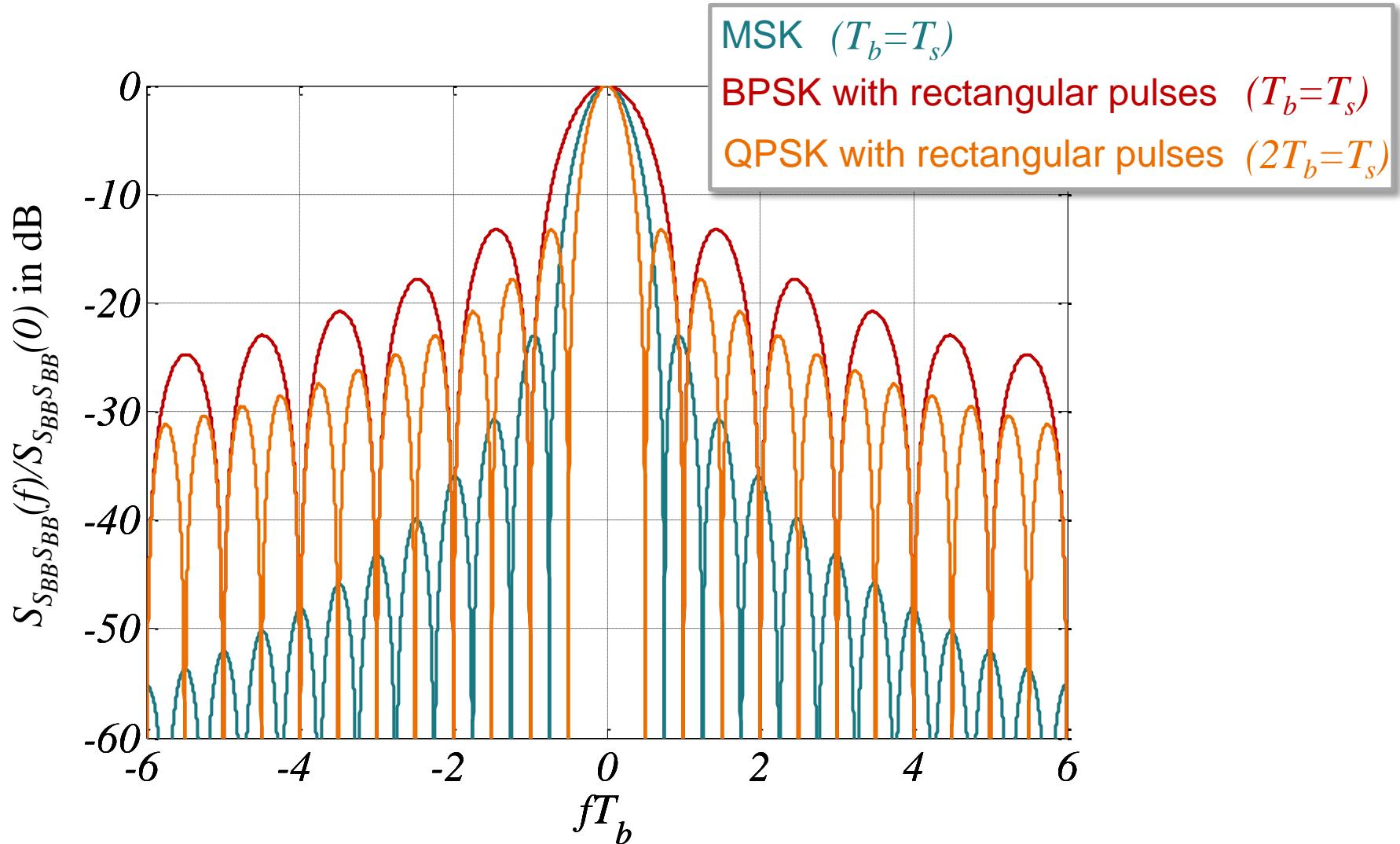
$$\bar{S}_{S_{BB}S_{BB}}(f) = \frac{1}{T_s} E\{|d_n|^2\} |G_t(f)|^2 = \frac{16T_s^2}{\pi^2} \left( \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2} \right)^2$$

$$\overbrace{\quad\quad\quad}^I$$

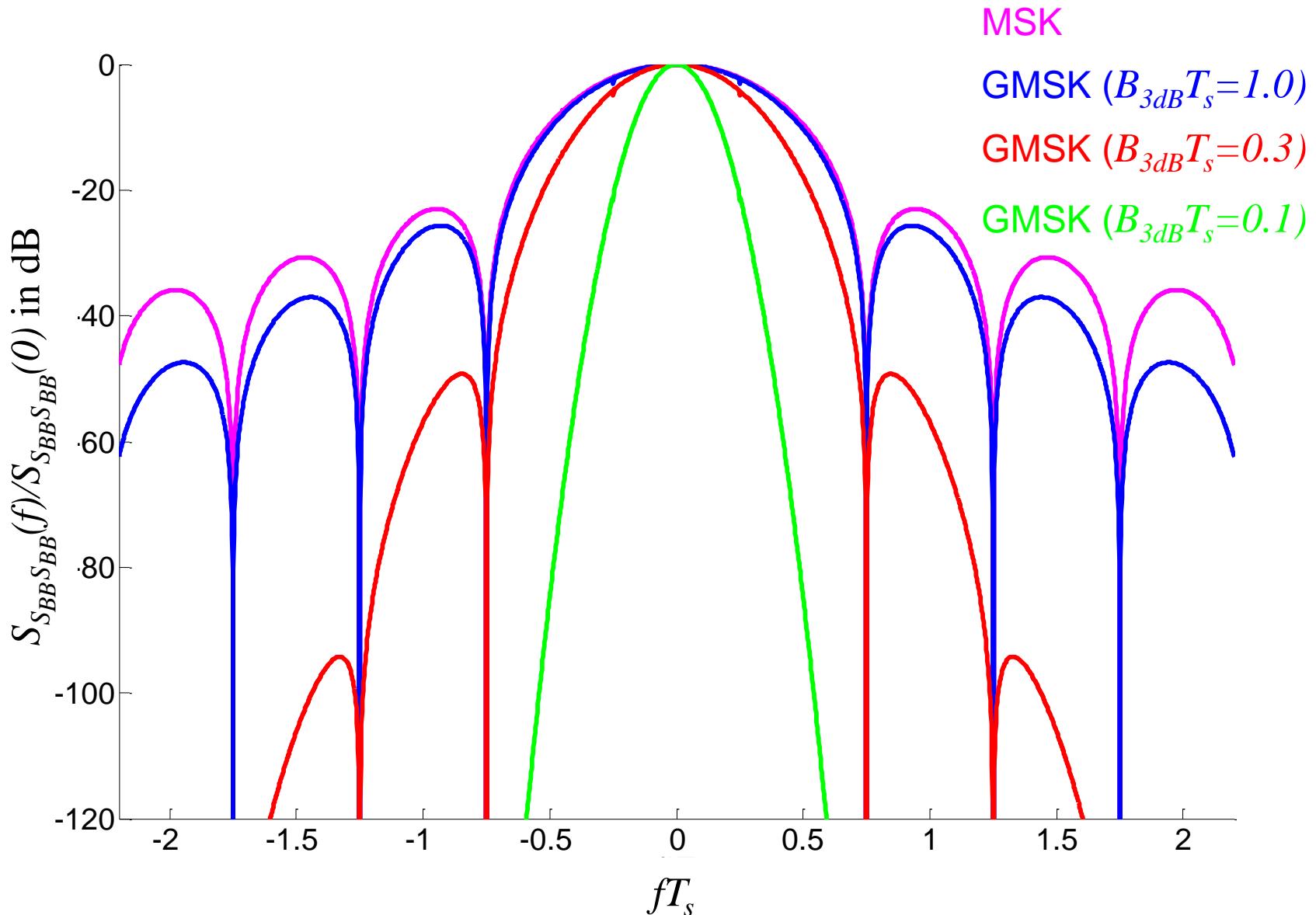
$$g_t(t) = \cos\left(\frac{\pi}{2T_s}t\right) \cdot \text{rect}\left(\frac{t}{2T_s}\right) \quad \bullet \quad G_t(f) = \frac{4T_s}{\pi} \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2}$$



# Power Spectral Density of MSK

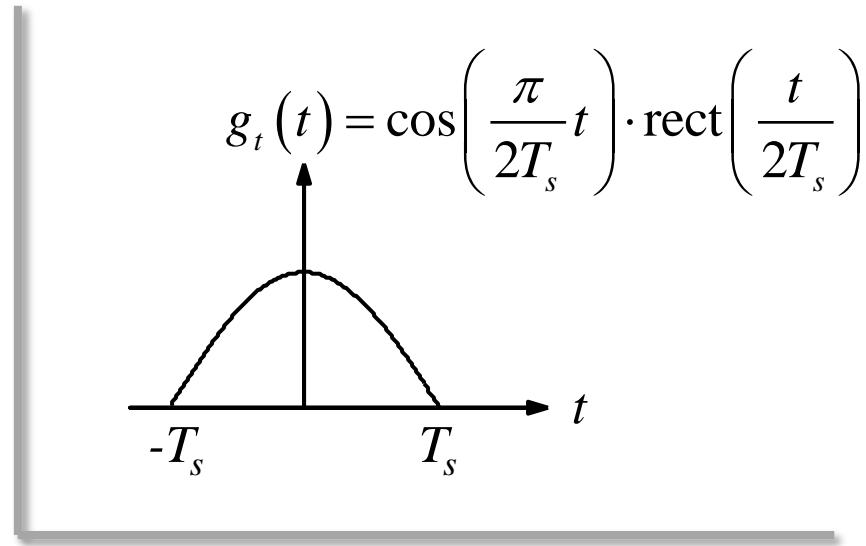


# Power Spectral Density of GMSK



# Derivation of Power Spectral Density of MSK (1)

$$\begin{aligned}
 G_t(f) &= \int_{-\infty}^{\infty} g_t(t) e^{-j2\pi ft} dt \\
 &= \int_{-T_s}^{T_s} \cos\left(\frac{\pi}{2T_s}t\right) e^{-j2\pi ft} dt \\
 &= \int_{-T_s}^{T_s} \frac{1}{2} \left( e^{j\frac{\pi}{2T_s}t} + e^{-j\frac{\pi}{2T_s}t} \right) e^{-j2\pi ft} dt \\
 &= \frac{1}{2} \int_{-T_s}^{T_s} e^{j\left(\frac{\pi}{2T_s}-2\pi f\right)t} + e^{-j\left(\frac{\pi}{2T_s}+2\pi f\right)t} dt
 \end{aligned}$$



$$= \frac{1}{2} \frac{1}{j\left(\frac{\pi}{2T_s}-2\pi f\right)} \left[ e^{j\left(\frac{\pi}{2T_s}-2\pi f\right)t} \right]_{-T_s}^{T_s} - \frac{1}{2} \frac{1}{j\left(\frac{\pi}{2T_s}+2\pi f\right)} \left[ e^{-j\left(\frac{\pi}{2T_s}+2\pi f\right)t} \right]_{-T_s}^{T_s}$$

# Derivation of Power Spectral Density of MSK (2)

$$G_t(f) = \frac{1}{2} \frac{1}{j \left( \frac{\pi}{2T_s} - 2\pi f \right)} \underbrace{\left[ e^{j\frac{\pi}{2}} e^{-j2\pi f T_s} - e^{-j\frac{\pi}{2}} e^{j2\pi f T_s} \right]}_{2j \cos(2\pi f)}$$

$$-\frac{1}{2} \frac{1}{j \left( \frac{\pi}{2T_s} + 2\pi f \right)} \underbrace{\left[ e^{-j\frac{\pi}{2}} e^{-j2\pi f T_s} - e^{j\frac{\pi}{2}} e^{j2\pi f T_s} \right]}_{-2j \cos(2\pi f)}$$

$$= \frac{1}{\left( \frac{\pi}{2T_s} - 2\pi f \right)} \cos(2\pi f T_s) + \frac{1}{\left( \frac{\pi}{2T_s} + 2\pi f \right)} \cos(2\pi f T_s)$$

# Derivation of Power Spectral Density of MSK (3)

$$G_t(f) = \frac{1}{\left(\frac{\pi}{2T_s} - 2\pi f\right)} \cos(2\pi f T_s) + \frac{1}{\left(\frac{\pi}{2T_s} + 2\pi f\right)} \cos(2\pi f T_s)$$

$$= \cos(2\pi f T_s) \frac{\frac{\pi}{2T_s} + 2\pi f + \frac{\pi}{2T_s} - 2\pi f}{\left(\frac{\pi}{2T_s} - 2\pi f\right)\left(\frac{\pi}{2T_s} + 2\pi f\right)} = \cos(2\pi f T_s) \frac{\frac{\pi}{T_s}}{\left(\frac{\pi}{2T_s}\right)^2 - (2\pi f)^2}$$

$$G_t(f) = \frac{4T_s}{\pi} \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2}$$

$$\Rightarrow |G_t(f)|^2 = \frac{16T_s^2}{\pi^2} \left( \frac{\cos(2\pi f T_s)}{1 - 16(f T_s)^2} \right)^2$$

# Coherent and Non-Coherent Detection

Coherent detection means that the downconversion of the received bandpass signal to the baseband at the receiver is done by a multiplication with a carrier oscillation  $\cos(2\pi f_c t + \varphi_c)$  and  $\sin(2\pi f_c t + \varphi_c)$ , respectively, with exactly the same frequency  $f_c$  and phase  $\varphi_c$  as the carrier oscillation of the received signal. This requires accurate knowledge of the carrier frequency and phase at the receiver, which is difficult to obtain in practice. Detectors use local oscillators for downconversion of the bandpass signal to the baseband. The frequency of the local oscillator will not be exactly the same as the frequency of the local oscillator which was used for upconversion at the transmitter. Therefore, the baseband signal will rotate in the signal space according to the frequency mismatch  $\Delta f$ . In coherent detectors, this frequency mismatch is removed by a subsequent carrier phase synchronization. However, most phase synchronisation algorithms suffer from a phase ambiguity of  $2\pi/M$ , where  $M$  is the number of discrete phase levels of the modulation scheme.

Modulation schemes which contain the information in the absolute phase of the transmit signal  $s(t)$  must be detected coherently. This applies for e.g. PSK, QAM and ASK modulation.

Differential modulation contains the information in the phase difference of successive transmit symbols rather than in the absolute phase. This allows to detect differentially modulated signals non-coherently, i.e. without explicit estimation of the carrier phase  $\varphi_c$ . Also FSK signals can be detected non-coherently as the information is contained in the instantaneous frequency of the transmit signal.

Non-coherent detection methods always contain non-linear elements such that the overall system is non-linear. As a result, even linear distortions during transmission result in non-linear distortions after non-coherent demodulation. This makes compensation of channel distortions as well as the analysis of non-coherent detection e.g. in terms of bit error rate performance in general more difficult.

# Differential Demodulation of FSK

Basically the same principle of differential demodulation as for DPSK can be applied for non-coherent demodulation of FSK signals.

The received signal  $r(t)$  is downconverted to the baseband. We assume that the frequency of the local oscillator at the receiver has a frequency offset  $\Delta f_c$  relative to the carrier of the received signal and the phase  $\varphi_c$  of the local oscillator is arbitrary, i.e. not synchronized to the received signal.

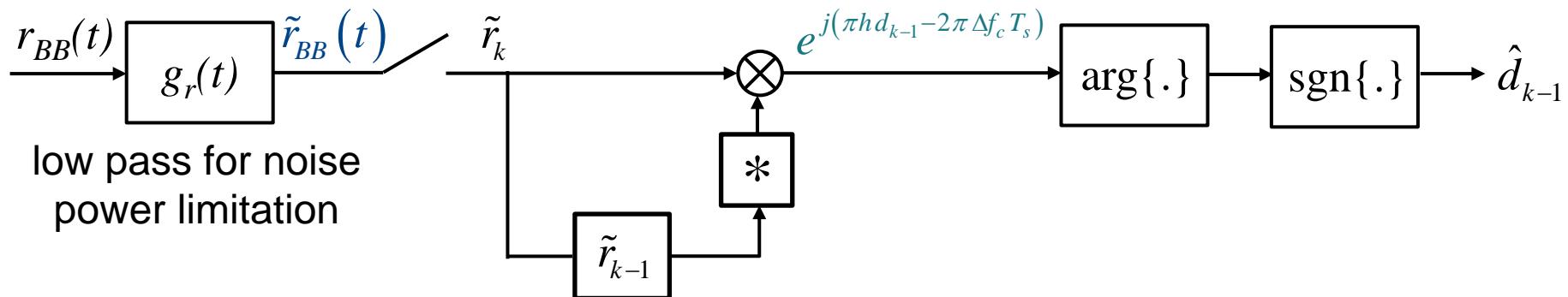
For FSK, receive filter  $g_r(t)$  is not a matched filter as in the case of DPSK, but it is a low pass filter for noise power limitation. At the receive filter output, the signal is sampled at the symbol rate  $1/T_s$ . Under the assumption of phase pulses  $q(t)$  which are limited in time and do not cause intersymbol interference at the sampling time, multiplication of each sample with the conjugate complex previous sample yields

$$\tilde{r}_{BB,k} \tilde{r}_{BB,k-1}^* = e^{j(\pi h d_{k-1} - 2\pi \Delta f_c T_s)} + \tilde{n}_k,$$

which is independent of the carrier phase offset  $\varphi_c$ . The phase of the resulting complex expression depends only on a single data symbol  $d_{k-1}$ . Hence, the data symbol  $d_{k-1}$  can be detected from the argument of  $\tilde{r}_{BB,k} \tilde{r}_{BB,k-1}^*$ .

It should be noted that the performance of the differential demodulator depends on the modulation index  $h$ . In case of binary FSK, a modulation index of  $h=1/2$  yields well distinguishable estimates for different data symbols  $d_{k-1}=+1$  and  $d_{k-1}=-1$ . Therefore, the detector is robust to additive noise and to an unknown frequency offset  $\Delta f_c$ . On the other hand, a modulation index  $h=1$  yields the same phase of  $\tilde{r}_{BB,k} \tilde{r}_{BB,k-1}^*$  for  $d_{k-1}=+1$  and  $d_{k-1}=-1$  such that the respective elementary signals cannot be distinguished. Consequently, the differential demodulator is not applicable for  $h=1$ .

# Differential Demodulation of FSK (1)



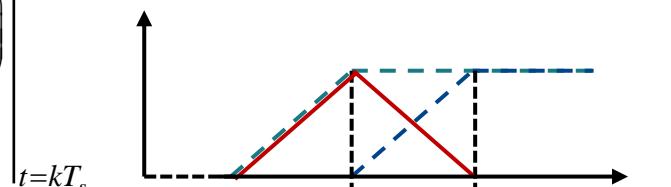
$$\tilde{r}_{BB}(t) = e^{j\left(\pi h \sum_{n=0}^{\infty} d_n q(t-nT_s) + \varphi_c - 2\pi \Delta f_c t\right)}$$

carrier phase and frequency offset

(AWGN neglected)

$$\tilde{r}_{BB}(t) \tilde{r}_{BB}^*(t - T_s) \Big|_{t=kT_s} = e^{j\left(\pi h \sum_{n=0}^{\infty} d_n q(t-nT_s) + \varphi_c - 2\pi \Delta f_c t\right)} e^{-j\left(\pi h \sum_{n=0}^{\infty} d_n q(t-nT_s - T_s) + \varphi_c - 2\pi \Delta f_c (t-T_s)\right)} \Big|_{t=kT_s}$$

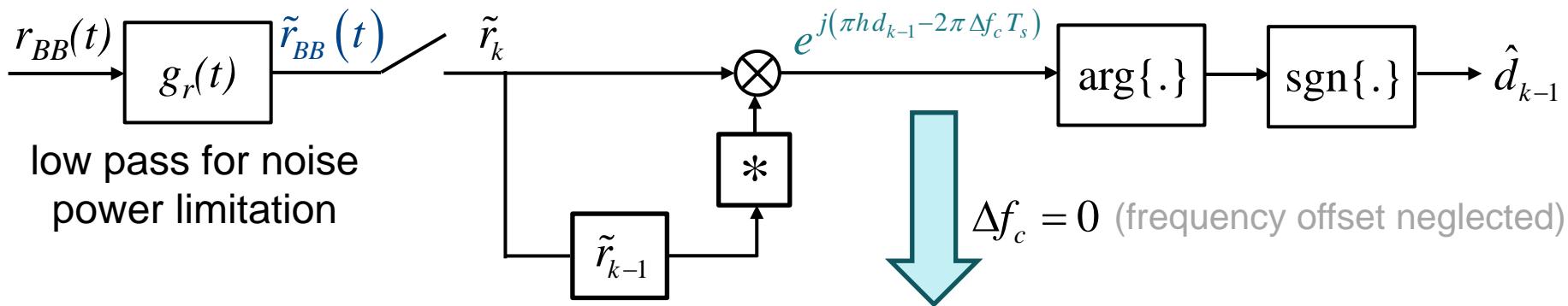
$$= e^{j\left(\pi h \sum_{n=0}^{\infty} d_n [q(t-nT_s) - q(t-nT_s - T_s)] - 2\pi \Delta f_c T_s\right)}$$



assumption: no ISI

$$= e^{j(\pi h d_{k-1} - 2\pi \Delta f_c T_s)}$$

# Differential Demodulation of FSK (2)



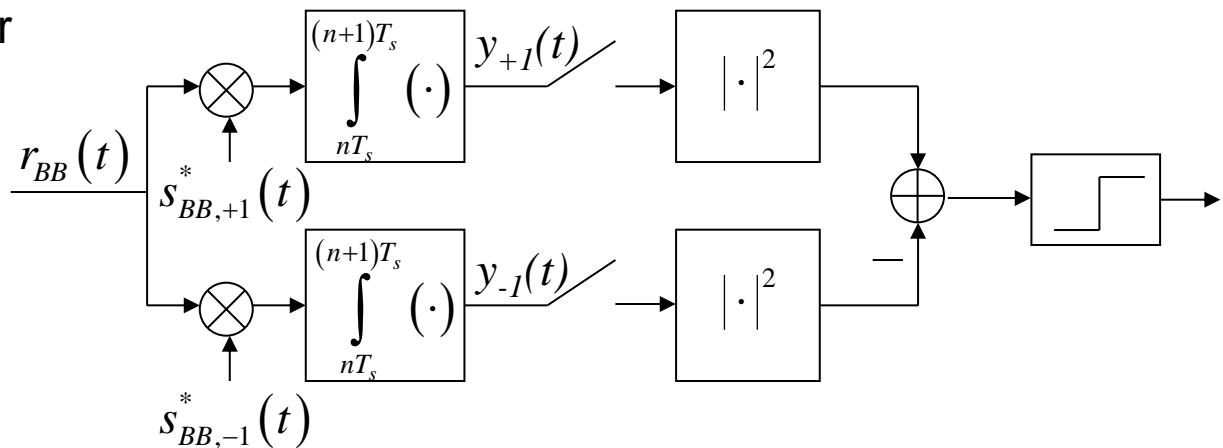
modulation index	$h = \frac{1}{2}$	$h = 1$
$d_{k-1} = +1:$	$e^{j\frac{\pi}{2}(+1)} = j$	$e^{j\pi(+1)} = -1$
$d_{k-1} = -1:$	$e^{j\frac{\pi}{2}(-1)} = -j$	$e^{j\pi(-1)} = -1$

good separability      elementary signals cannot be distinguished

The differential modulator works best for a modulation index  $h=1/2$ . It is not suitable for a modulation index  $h=1$ .

# Non-Coherent FSK Detection

Binary FSK with rectangular frequency pulses.



Possible equivalent baseband signals ( $0 \leq t \leq T_s$ ):

$$s_{BB,+1}(t) = e^{j\pi h \frac{t}{T_s} (+1)} \text{ for } d_n = +1, \quad s_{BB,-1}(t) = e^{j\pi h \frac{t}{T_s} (-1)} \text{ for } d_n = -1$$

$d_n = +1 \Rightarrow$  Received equivalent baseband signal:  $r_{BB}(t) = e^{j\pi h \frac{t-nT_s}{T_s} (+1)} \cdot e^{j\varphi_0} + n_{BB}(t)$

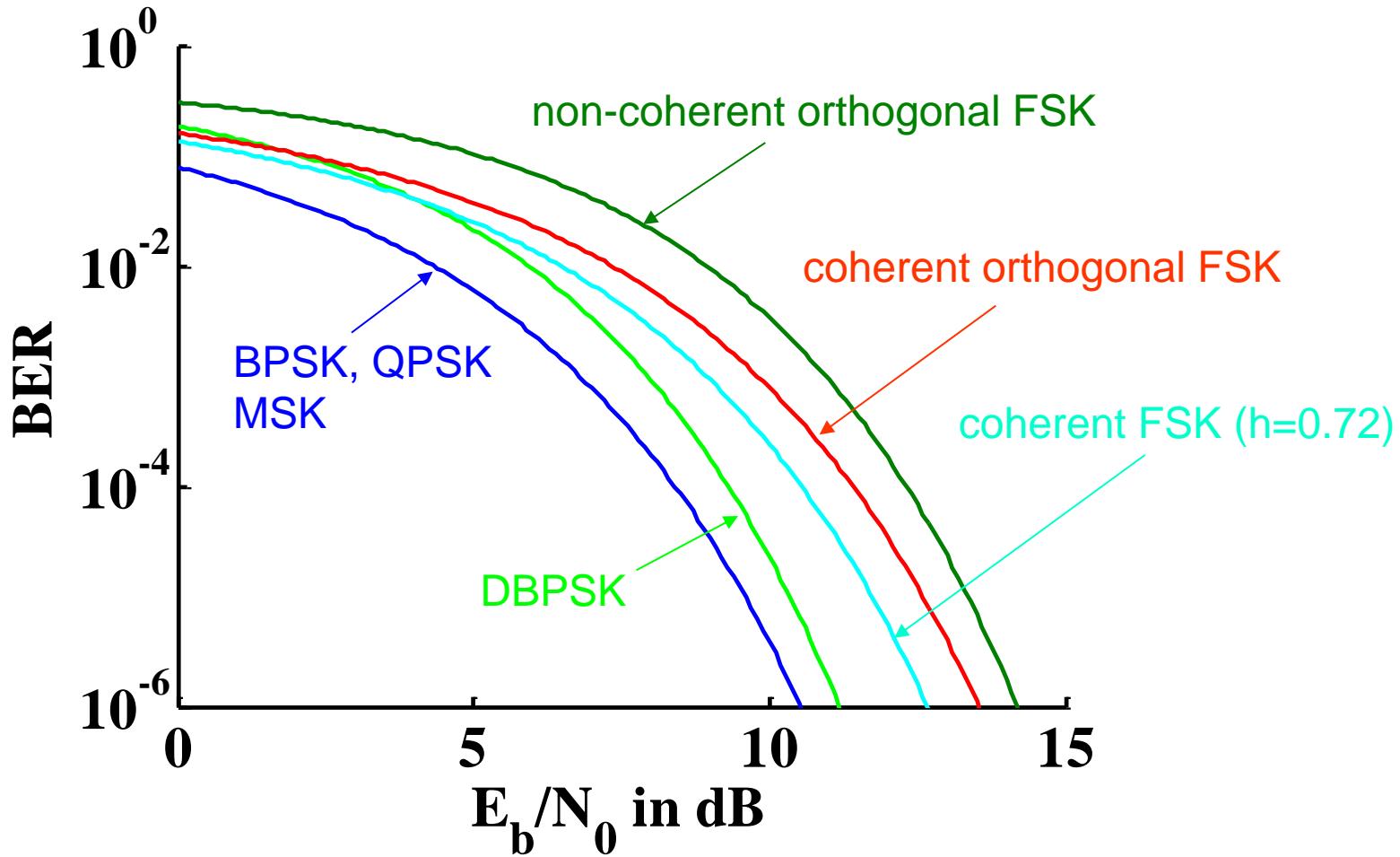
$$y_{+1}(t) = \int_{nT_s}^{(n+1)T_s} r_{BB}(t) s_{BB,+1}^*(t - nT_s) dt = \int_{nT_s}^{(n+1)T_s} e^{j\varphi_0} e^{j\pi h \frac{t-nT_s}{T_s}} e^{-j\pi h \frac{t-nT_s}{T_s}} + \tilde{n}_1(t) = T_s e^{j\varphi_0} + \tilde{n}_1(t)$$

$$y_{-1}(t) = \int_{nT_s}^{(n+1)T_s} r_{BB}(t) s_{BB,-1}^*(t - nT_s) dt = \int_{nT_s}^{(n+1)T_s} e^{j\varphi_0} e^{j\pi h \frac{t-nT_s}{T_s}} e^{j\pi h \frac{t-nT_s}{T_s}} + \tilde{n}_2(t) = e^{j\varphi_0} \underbrace{\frac{e^{j2\pi h} - 1}{j2\pi h / T_s}}_{=0 \text{ for } h=1, 2, \dots} + \tilde{n}_2(t)$$

# Bit Error Probability of Digital Modulation Methods

method	$P_b$	
coherent BPSK	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$	$E_s = E_b$
coherent QPSK	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$	$E_s = 2E_b$
coherent $M$ -PSK	$\approx \frac{1}{\log_2 M} \operatorname{erfc}\left(\sqrt{\frac{E_s}{N_0}} \sin \frac{\pi}{M}\right)$	$E_s = E_b \log_2 M$
non-coherent DBPSK	$\frac{1}{2} e^{-E_b / N_0}$	$E_s = E_b$
coherent MSK	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$	$E_s = E_b$
coherent orthogonal FSK ( $h=1, 1.5, 2 \dots$ )	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{2N_0}}\right)$	$E_s = E_b$
coherent FSK ( $h=0.72$ )	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{E_b}{2N_0}} 1.22\right)$	$E_s = E_b$
non-coherent orthogonal FSK ( $h=1, 2 \dots$ )	$\frac{1}{2} e^{-E_b / (2N_0)}$	$E_s = E_b$

# Bit Error Probability Comparison of Digital Modulation Methods



# Frequency-Selective Channels (1)

Frequency-selective channels are characterized by a frequency response  $H(f)$  which is not constant within the passband of the transmission system. A frequency-selective frequency response corresponds to a channel impulse response  $h(\tau)$  which is not just a Dirac impulse but which shows some dispersion in time. The length of the impulse response is called the maximum channel delay  $\tau_{\max}$ . Consequently, each received sample will be affected by not only a single transmit symbol but by several successively transmitted symbols. This effect is called *intersymbol interference (ISI)*.

Intersymbol interference is caused by multipath propagation which to some extend occurs in most transmission channels. The most intuitive explanation of multipath propagation is for wireless channels: Here, the transmitted signal is reflected at objects such as buildings, mountains, cars. As a result, the receiver observes the superposition of all those multipath components which have different delay.

However, multipath propagation also occurs in other media such as optical communications or wireline communications. Impedance mismatch causing refelections is e.g. a reason for multipath propagation in wireline transmission.

The impact of multipath components on the received signal depends on the ratio of multipath delays and symbol duration  $T_s$  as well as on the power of the multipath components.

A multipath component with long delay relative to the first arriving multipath component may have negligible impact, if it is very weak. On the other hand, strong late arriving multipath components will have detrimental effect and require compensation at the receiver.

Detection stages for elimination of intersymbol interference at the receiver are called *equalizers*.

# Frequency-Selective Channels (2)

The amount of intersymbol interference (frequency-selectivity) and, consequently, the need for equalization, can be characterized by *the root mean square (rms) delay spread*  $\tau_{\text{rms}}$  of the channel. Unlike the maximum channel delay  $\tau_{\text{max}}$ , the rms delay spread  $\tau_{\text{rms}}$  takes the strength of the multipath components at different delay  $\tau$  into account.

The rms delay spread of a channel with impulse response  $h(\tau)$  is defined by

$$\tau_{\text{rms}} = \sqrt{\frac{\int_0^{\infty} (\tau - \bar{\tau})^2 |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}},$$

where

$$\bar{\tau} = \frac{\int_0^{\infty} \tau |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}$$

is the mean delay of the channel.

# Frequency-Selective Channels (3)

If the symbol duration  $T_s$  is much longer than the rms delay spread  $\tau_{\text{rms}}$ , i.e.  $T_s \gg \tau_{\text{rms}}$ , the magnitude  $|H(f)|$  of the channel transfer function can be considered to be about constant within the bandwidth of the transmit signal and the channel may be considered to be frequency-flat. In case the symbol duration is significantly smaller than the rms delay spread, i.e.  $T_s \ll \tau_{\text{rms}}$ , the channel cannot be considered to be about constant within the bandwidth of the transmit signal. Hence, the channel is frequency-selective.

As the bandwidth needed to transmit at a Baud rate of  $1/T_s$  is related to  $1/T_s$ , we call a channel a narrowband channel, if  $1/T_s \ll 1/\tau_{\text{rms}}$ . Otherwise, the channel is called a broadband channel.

As a rule of thumb, a channel is considered to be frequency-selective and, consequently, needs equalization, if the rms delay spread is larger than 10% of the symbol duration  $T_s$ , i.e. if  $\tau_{\text{rms}} > 0.1 T_s$ .

# Receive Filter Design for Frequency-Selective Channels (1)

In an AWGN channel, the optimum receive filter which maximizes the SNR at the receive filter output is the matched filter. In frequency-flat channels, where the impulse response of the channel is just a Dirac impulse, i.e.  $h_c(t) = \delta(t)$ , the impulse response  $g_r(t)$  of the matched filter is determined by the impulse response of the pulse shaping filter at the transmitter, i.e.  $g_r(t) = g_t^*(-t)$ . The matched filter may be implemented as analog filter. In case it is implemented as a digital filter, the received signal is sampled at a high sampling rate  $1/T_a$  which meets the sampling theorem. At the output of the matched filter, sampling at the symbol rate  $1/T_s$  yields a so-called set of sufficient statistics, i.e. no information is lost.

Note, that sampling at the symbol rate  $1/T_s$  does not meet the sampling theorem. However, this is not a problem since we are not interested in reconstructing the analog transmit signal  $s(t)$  but only in detecting the data symbols  $d_n$ .

If the impulse response  $g_t(t) * g_r(t)$  of the total transmission system including transmit filter  $g_t(t)$  and receive filter  $g_r(t)$  meets the first Nyquist condition, the samples at the matched filter output are free of intersymbol interference (ISI).

As the received signal does not only contain the message signal but also additive noise, the noise is also passed through the receive filter. Filtering of white Gaussian noise in general yields coloured noise. However, it is desirable to obtain white, i.e. uncorrelated noise also at the filter output as it usually simplifies the successive detection stages.

If the matched filter has a square-root Nyquist characteristic and the channel noise is additive white Gaussian noise (AWGN), the symbol-spaced samples at the matched filter output will be corrupted by additive white Gaussian noise.

# Receive Filter Design for Frequency-Selective Channels (2)

In frequency-selective channels, the received signal was filtered not only by the transmit filter  $g_t(t)$  but also by the ISI channel with impulse response  $h_c(t)$ . Consequently, the matched filter  $g_r(t)$  needs to be adapted not only to the transmit filter  $g_t(t)$  but to the total impulse response resulting from the convolution  $h_t(t) = g_t(t) * h_c(t)$ , i.e. the impulse response of the matched filter is given by  $g_r(t) = h_t^*(-t)$ . Unfortunately, the overall system impulse response  $h_t(t) * g_r(t)$  will not meet the first Nyquist condition and, hence, the samples at the receive filter output will be corrupted by intersymbol interference (ISI). Therefore, a successive detection stage called equalizer is needed in order to compensate for the detrimental effect of ISI.

Moreover, the matched filter  $g_r(t) = h_t^*(-t)$  does not have a square-root Nyquist characteristic anymore and, hence, the noise in the samples at the matched filter output will be coloured. Another problem with adapting the receive filter to the total impulse response

$h_t(t) = g_t(t) * h_c(t)$  is that the channel impulse response  $h_c(t)$  needs to be estimated at a high sampling rate  $1/T_a$ , which is computationally expensive and requires transmission of a significant amount of known pilot symbols for channel estimation. For successive detection stages after the receive filter, channel estimation is only required at the lower symbol rate  $1/T_s$ .

Moreover, the channel  $h_c(t)$  may change in time which requires highly adaptive channel estimation and frequent updates. This is e.g. the case in mobile communications, where the terminals move around and, hence, experience different channels in each location.

In order to avoid the aforementioned problems with the matched filter, a suboptimum fixed receive filter  $g_r(t)$  is usually applied.

# Receive Filter Design for Frequency-Selective Channels (3)

One option is to match the receive filter to the transmit filter only, i.e.  $g_r(t) = g_t^*(-t)$ , ignoring the impact of the ISI channel  $h_c(t)$ . If  $g_r(t)$  has a square-root Nyquist characteristic, at least the noise remains white and Gaussian after sampling at the symbol rate  $1/T_s$ . However, the total system impulse response  $h_t(t) * g_r(t)$  will still not meet the first Nyquist condition and, hence, an equalizer is needed in order to combat the ISI.

Some detection stages such as fractionally spaced equalizers operate at a multiple of the symbol rate  $1/T_s$ . In this case, the receive filter output is not sampled at the symbol rate  $1/T_s$  but at a multiple  $c/T_s$  of the symbol rate resulting in a sample spacing of  $T_s/c$ . As the sample spacing is now a fraction of the symbol duration  $T_s$ , the samples are said to be fractionally spaced. Consequently, the noise in the samples will be coloured even if the receive filter has a square-root Nyquist characteristic. This has to be taken into account in the design of the successive detection stages.

Anyway, in many applications, other criteria have to be taken into consideration in the receive filter design. E.g. it is often most important to suppress co-channel interference. In this case, receive filters with strong stopband suppression are applied which may neither meet the matched filter condition nor provide white Gaussian noise after sampling at the receive filter output. However, the system performance may be better using such a compromise receive filter.

Despite the actually coloured noise, successive detection stages may then just operate under the (wrong) assumption of white noise or apply special means of signal processing in order to decorrelate the noise.

# Equalization (1)

It is the task of an equalizer to eliminate the intersymbol interference. Basically, there are two fundamental approaches to solve the problem of equalization: The signal processing based approach aims at designing a receive filter which compensates for the intersymbol interference. In the time domain, this can be viewed as a filter  $g_e(t)$  which meets

$$h(t) * g_e(t) = \delta(t),$$

i.e. the overall impulse response of the concatenation of channel and equalizer yields a Dirac impulse. Hence, the overall system is free of intersymbol interference. Since a Dirac impulse in the frequency domain corresponds to a constant frequency response, the equalizer should invert the frequency response of the channel, i.e.

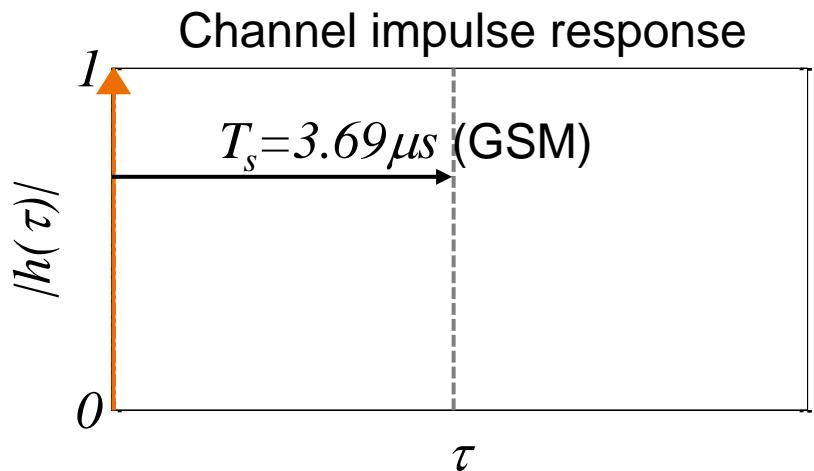
$$H(f) G_e(f) = \text{const} \Rightarrow G_e(f) \sim 1/H(f).$$

Variants of the aforementioned filtering approach can be implemented with manageable complexity in many scenarios.

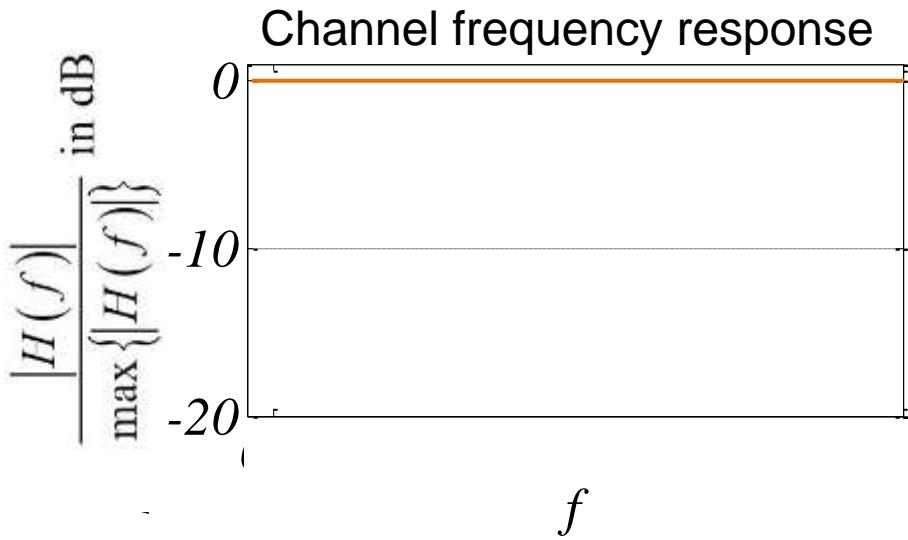
# Equalization

However, from an error rate performance point of view, the filtering approach is suboptimum in noisy channels. The optimum solution is a maximum likelihood or maximum a posteriori sequence estimator, which estimates the most likely transmitted sequence given the sequence of received samples. The Viterbi algorithm provides a method for implementation of the optimum equalizer. The major achievement of the Viterbi algorithm is that its computational complexity grows only linearly with the length of the transmitted sequence of data symbols. However, it grows exponentially with the length of the channel impulse response making the Viterbi algorithm prohibitively complex for scenarios with significant time dispersion, i.e. long channel impulse response.

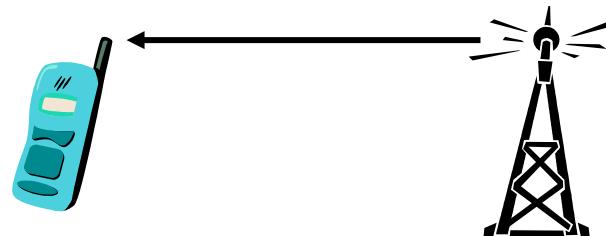
# Frequency-Flat Channels



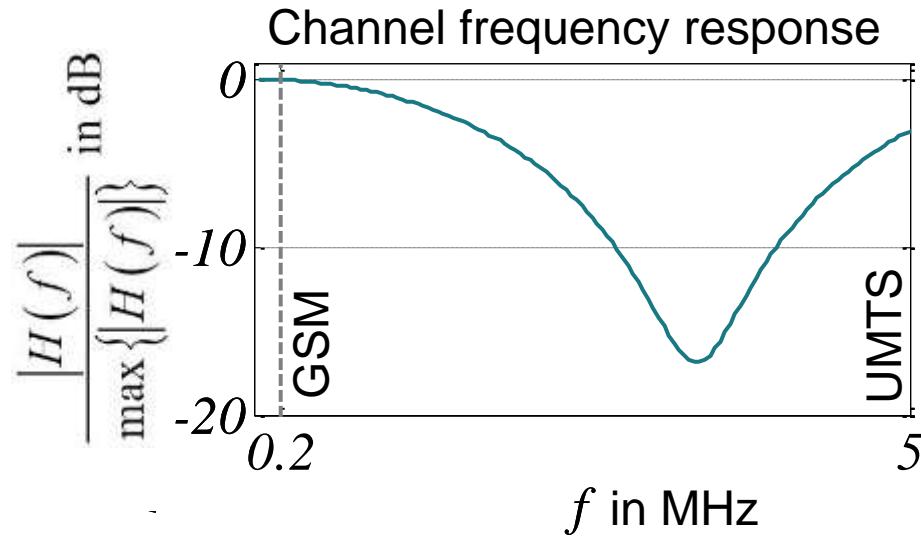
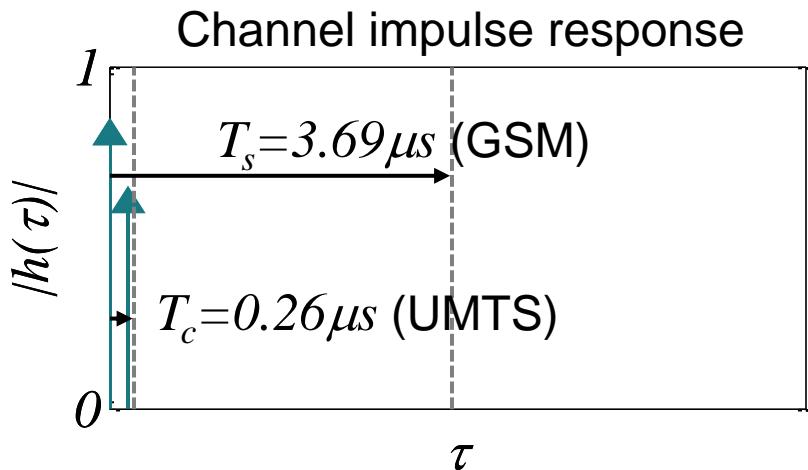
no intersymbol interference (ISI)



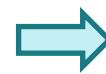
frequency-flat channel



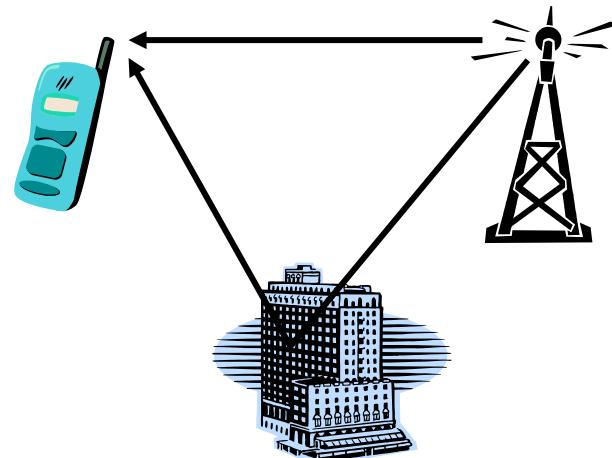
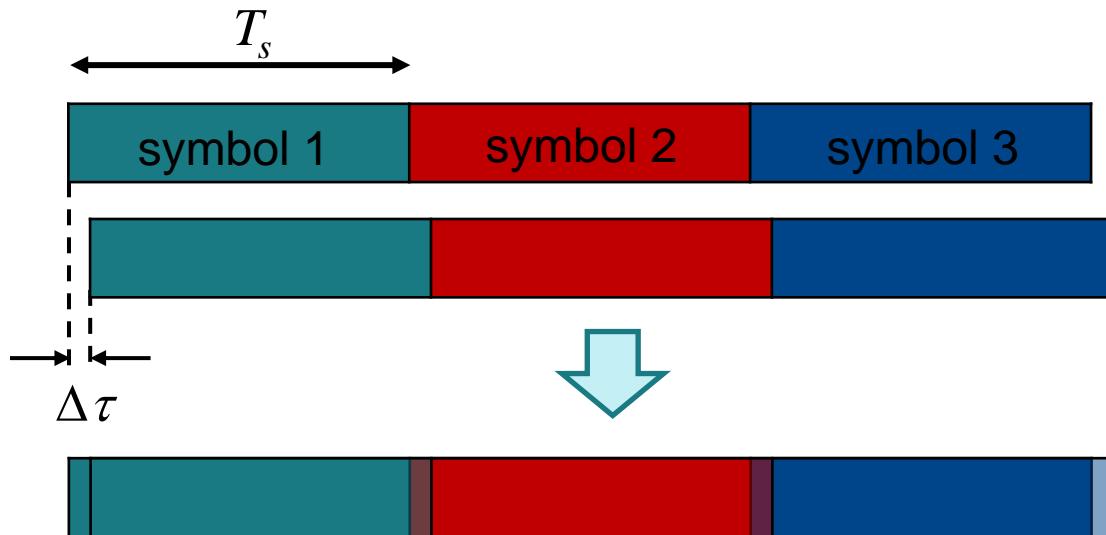
# Frequency-Selective Channels (1)



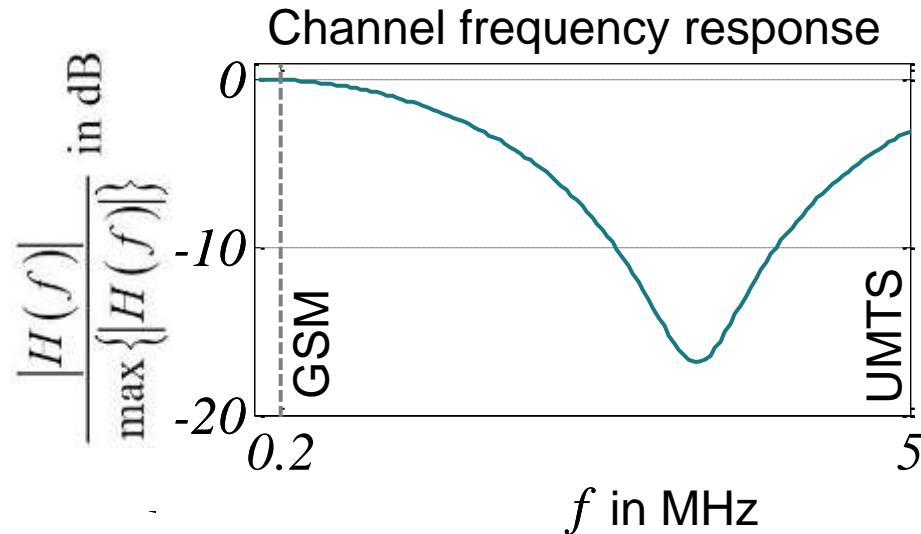
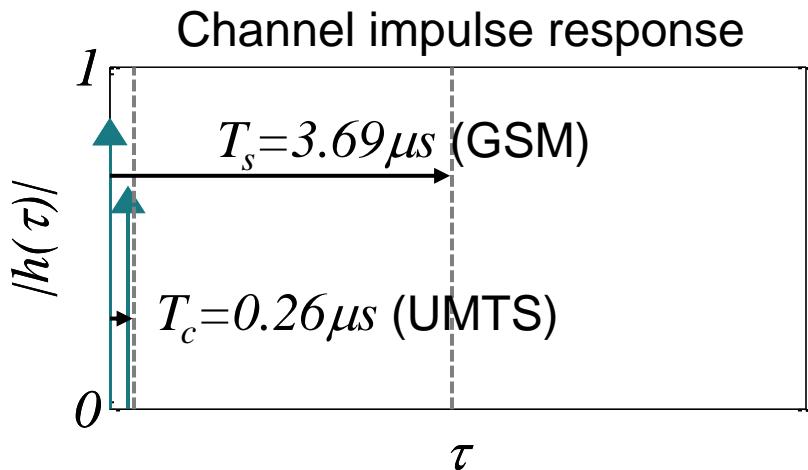
weak intersymbol interference (ISI)



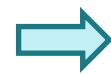
almost frequency-flat  
narrowband channel



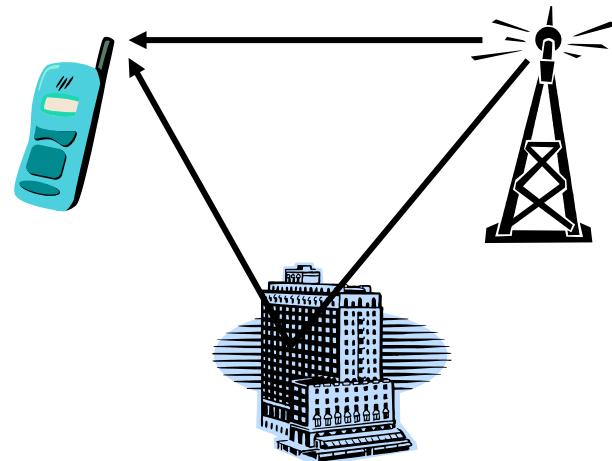
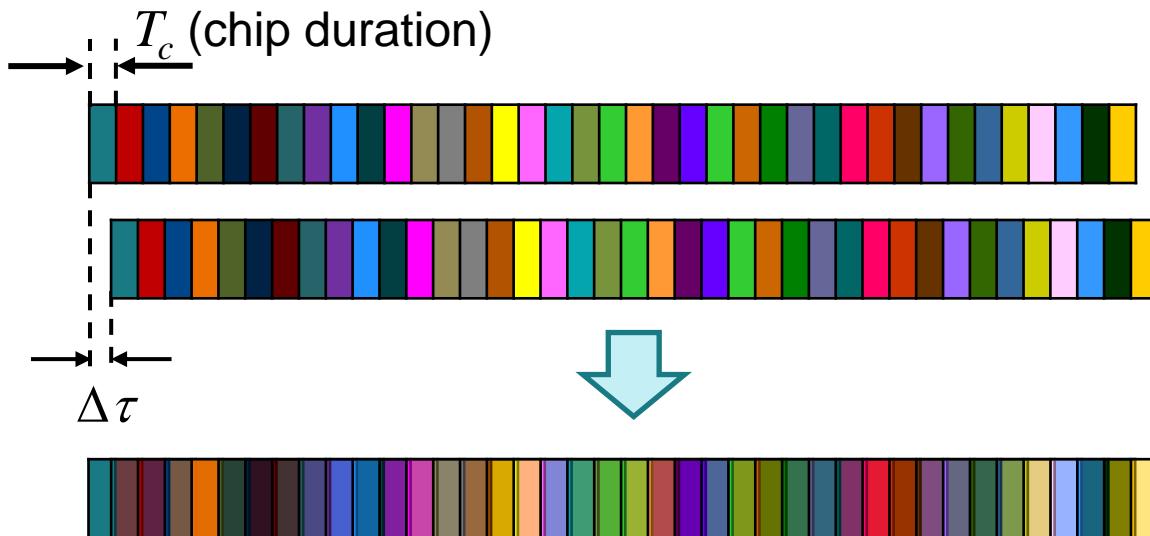
# Frequency-Selective Channels (2)



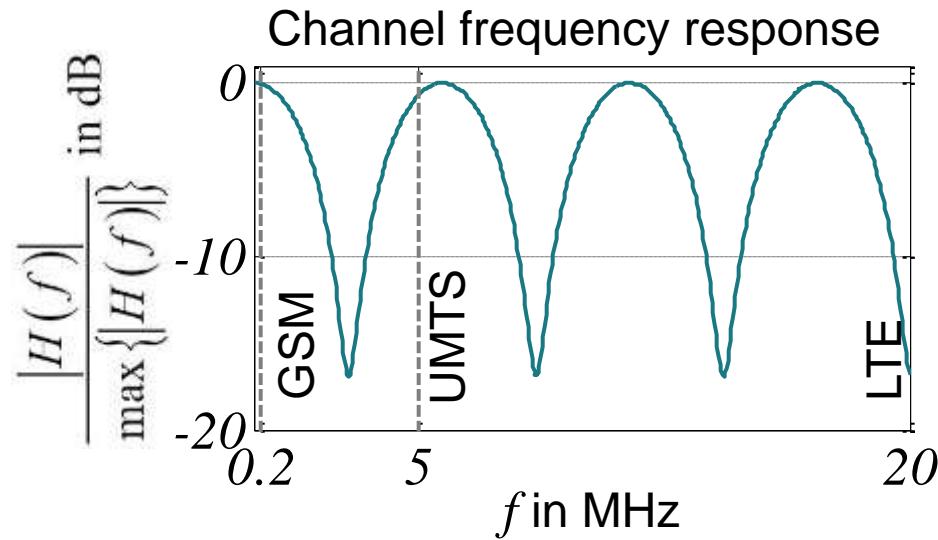
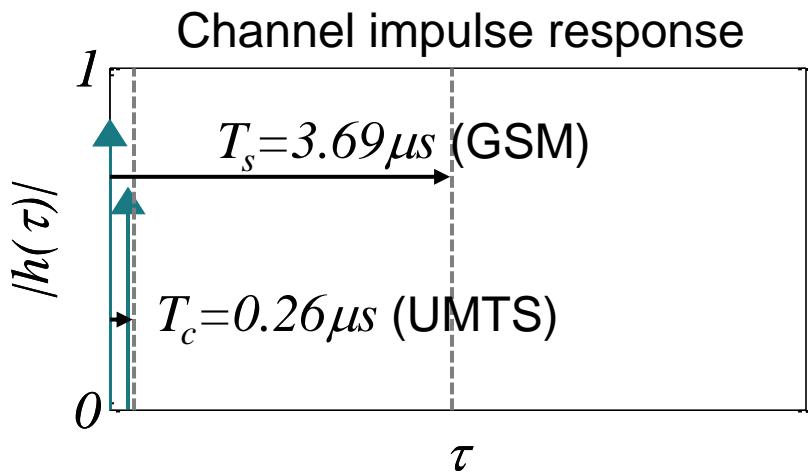
strong intersymbol interference (ISI)



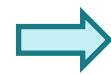
frequency-selective  
wideband channel



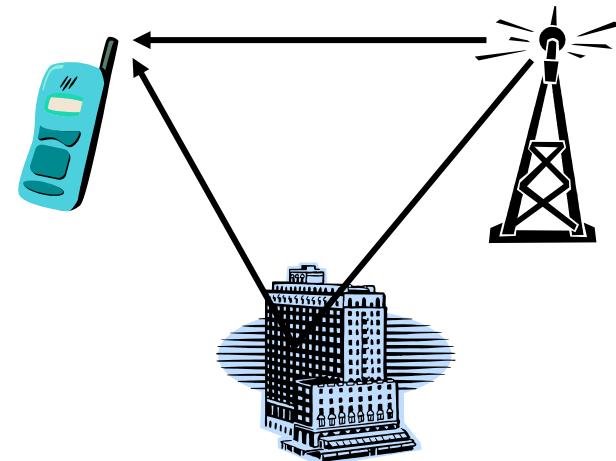
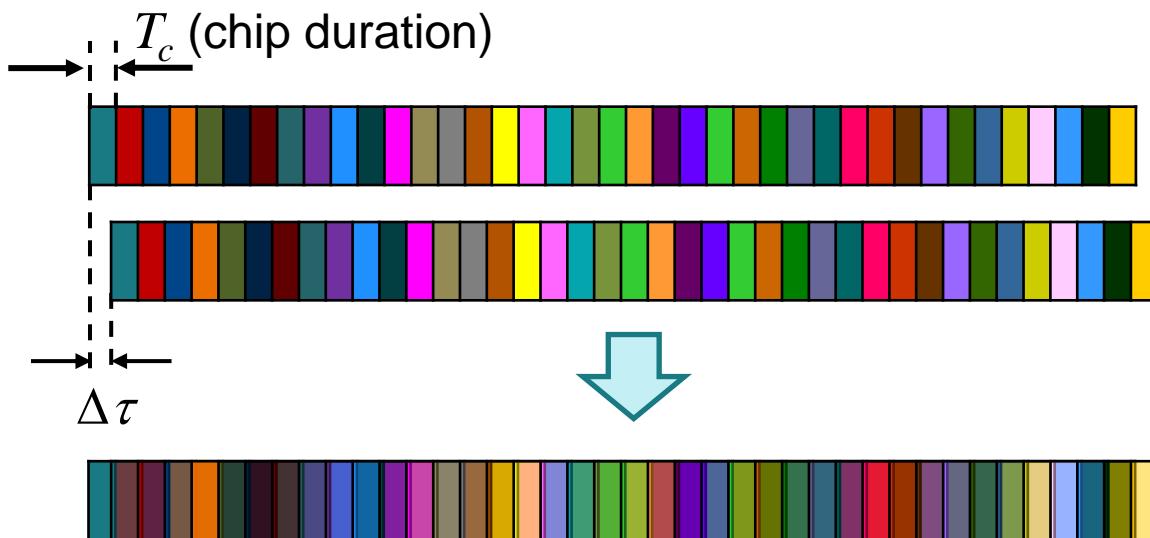
# Frequency-Selective Channels (3)



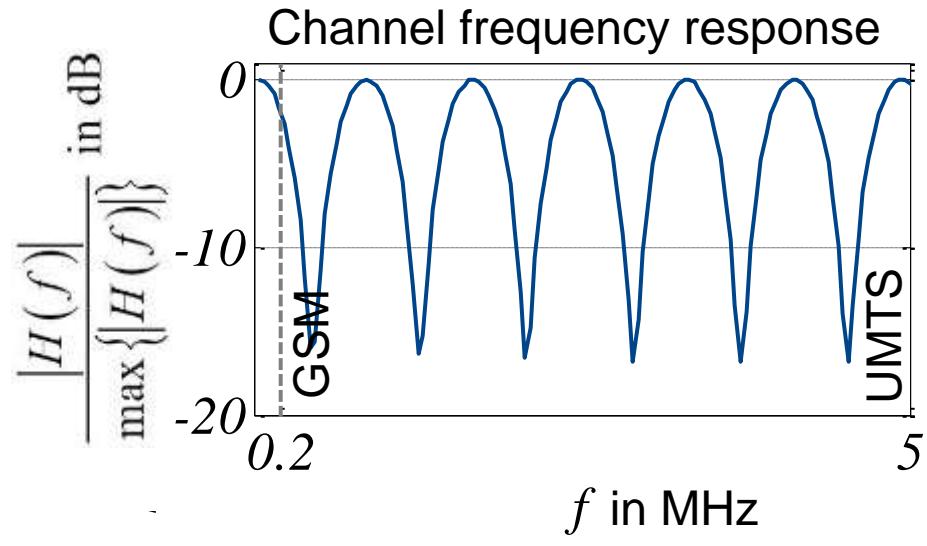
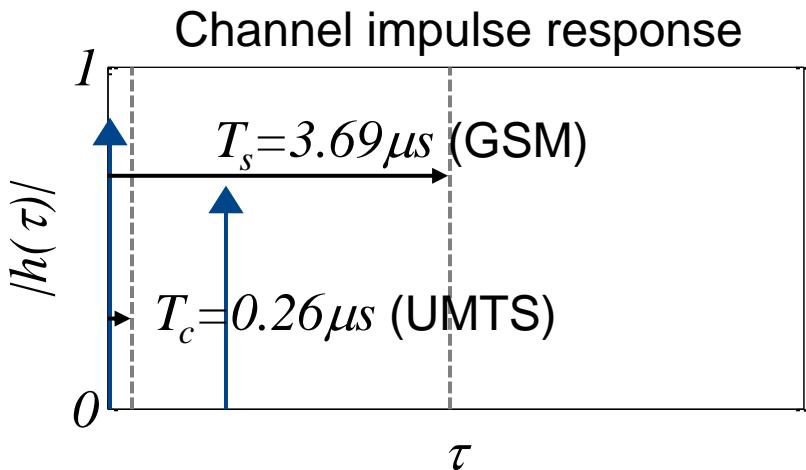
strong intersymbol interference (ISI)



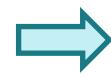
frequency-selective  
broadband channel



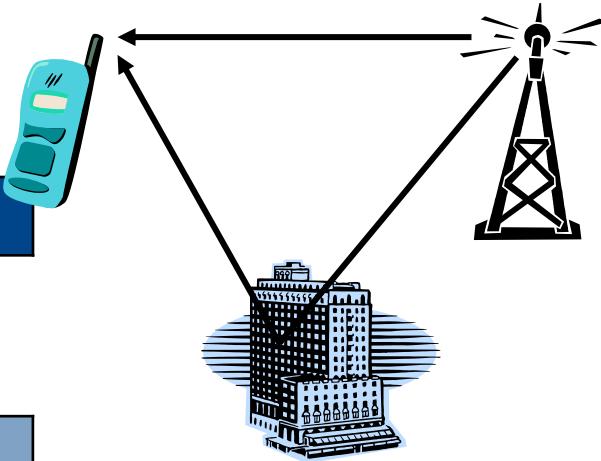
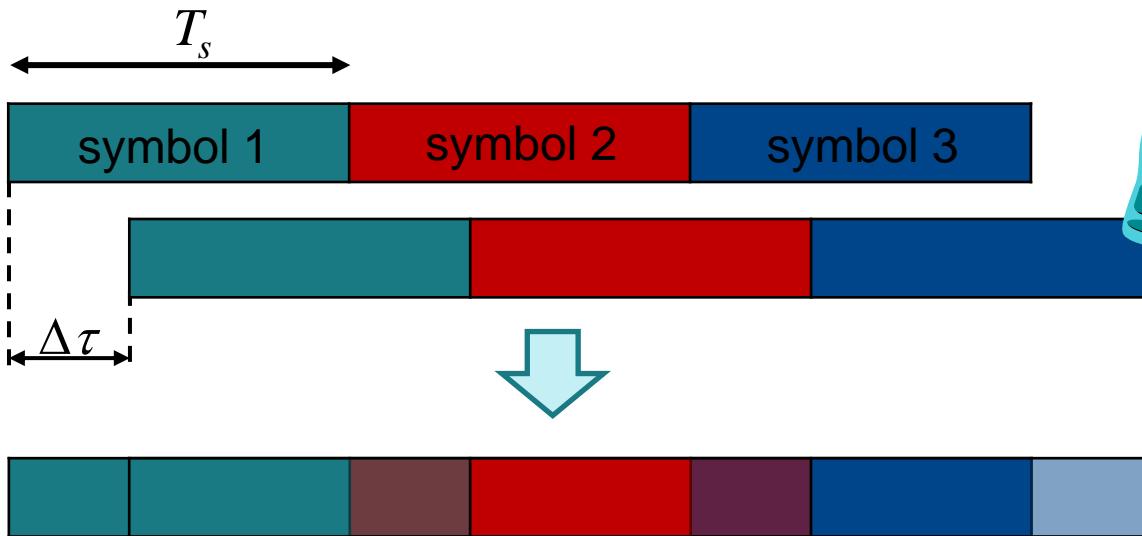
# Frequency-Selective Channels (4)



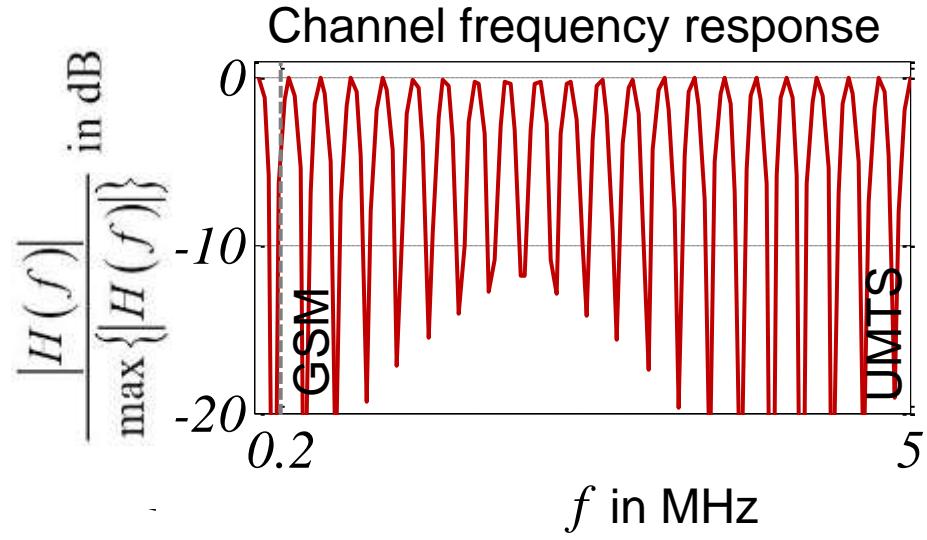
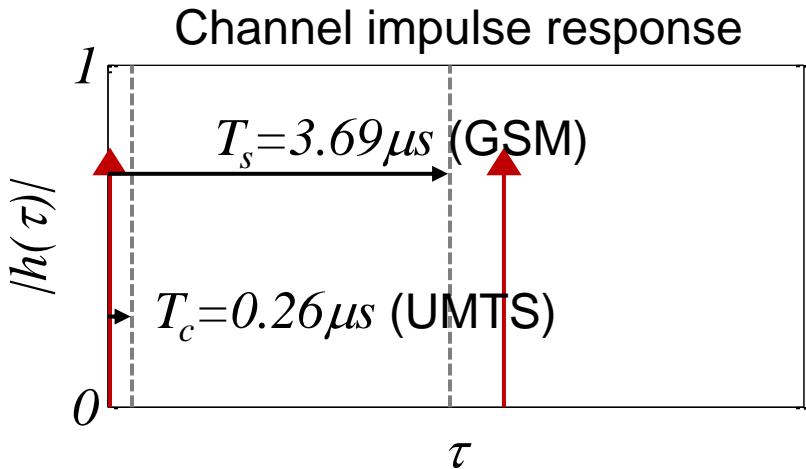
strong intersymbol interference (ISI)



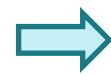
frequency-selective channel



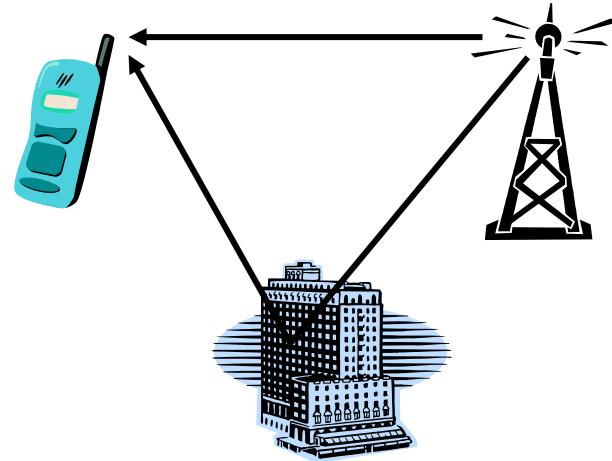
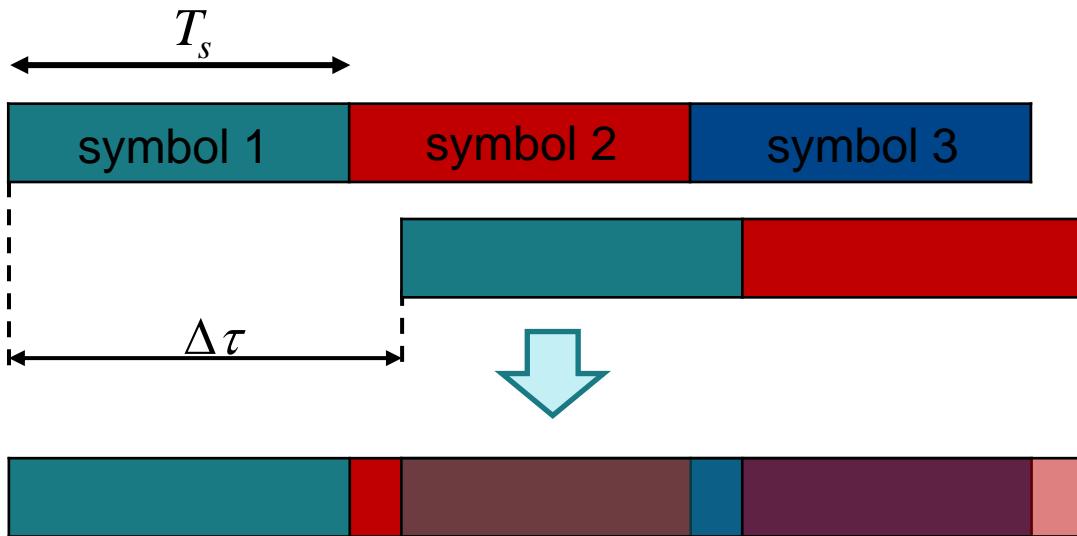
# Frequency-Selective Channels (5)



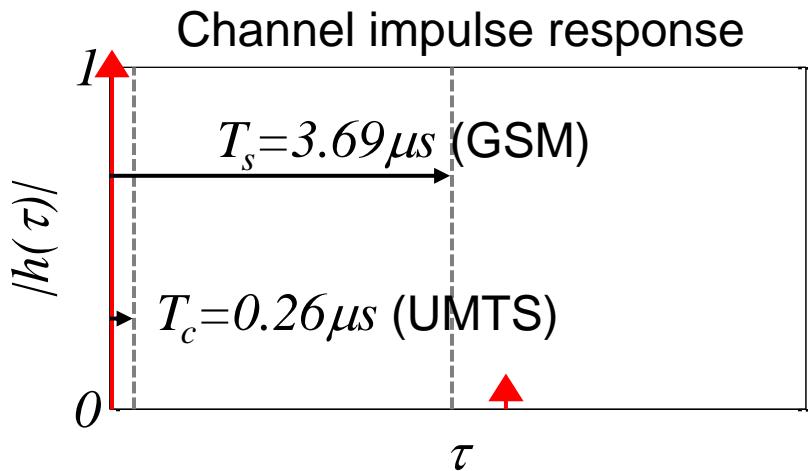
strong intersymbol interference (ISI)



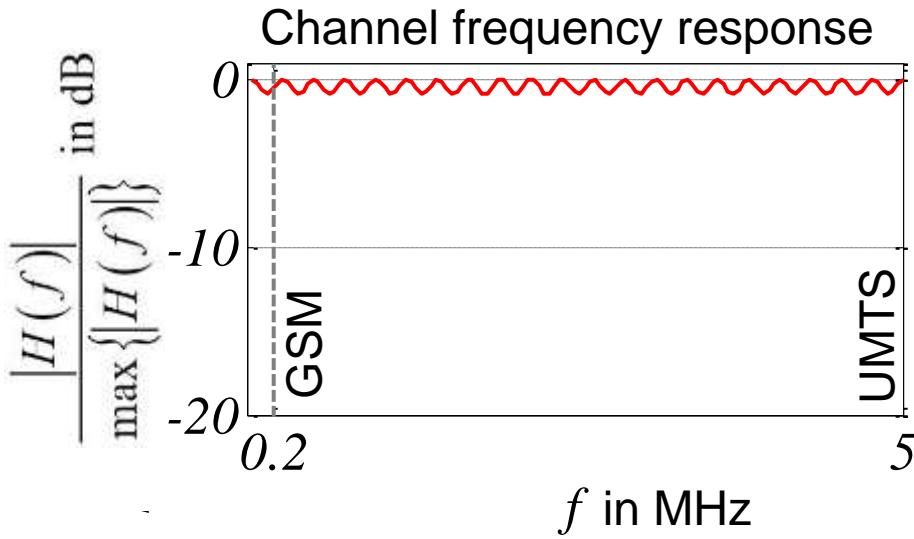
frequency-selective channel



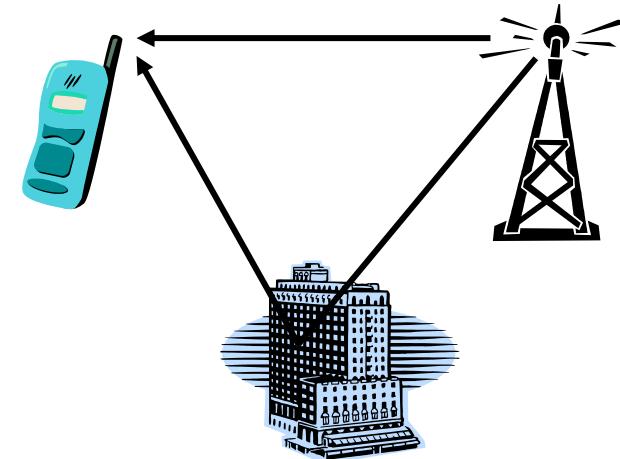
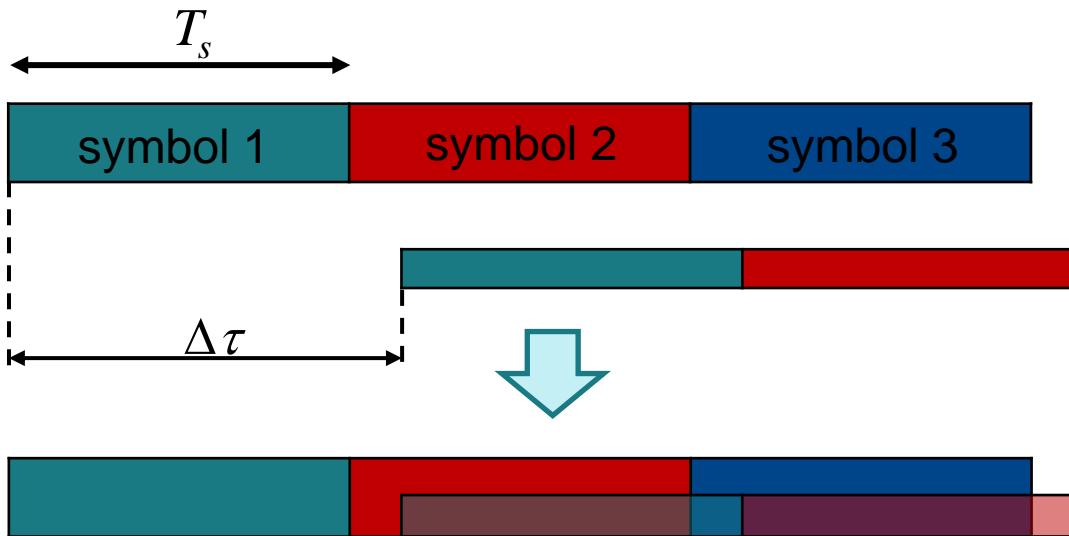
# Frequency-Selective Channels (6)



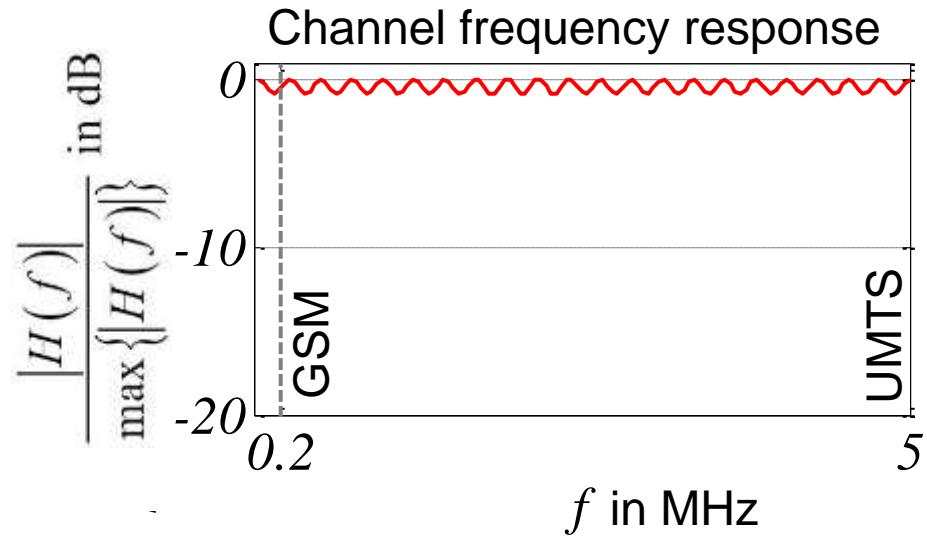
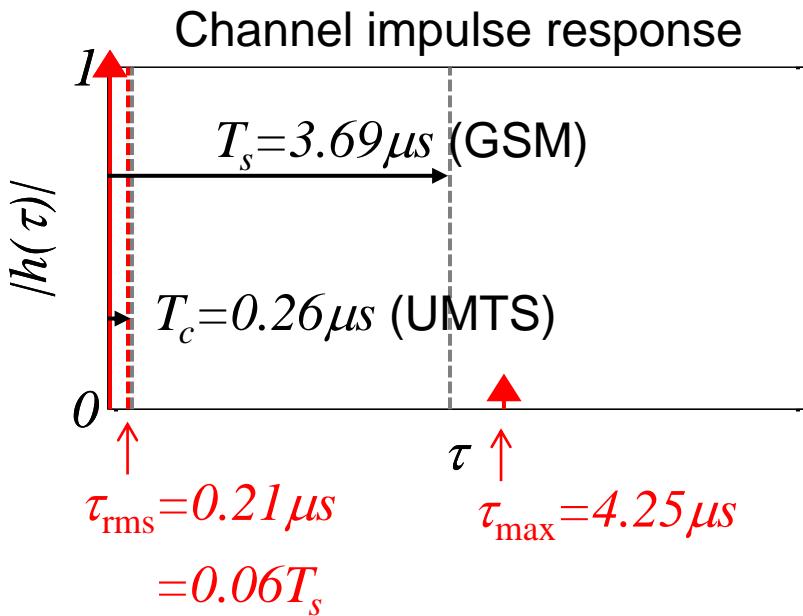
weak intersymbol interference (ISI)



almost frequency-flat channel



# Root Mean Square (RMS) Delay Spread (1)



RMS delay spread:

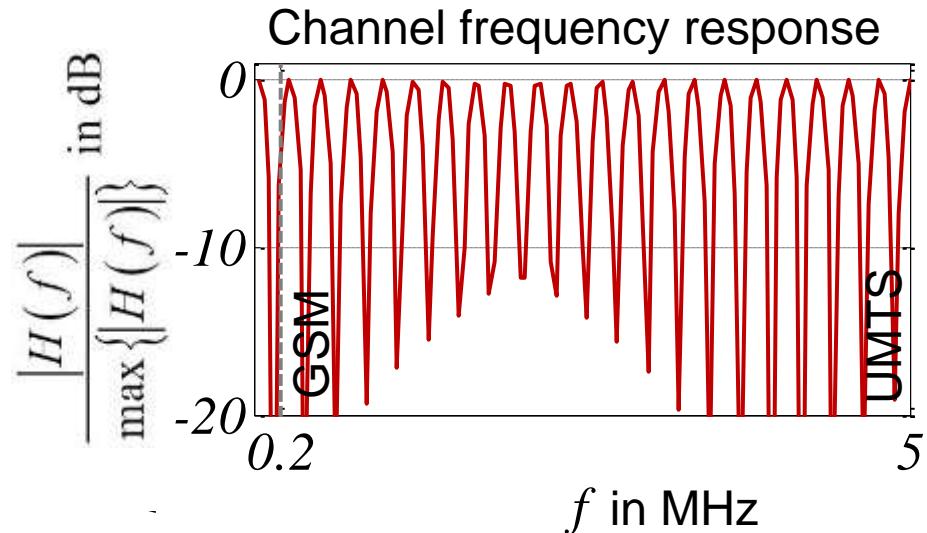
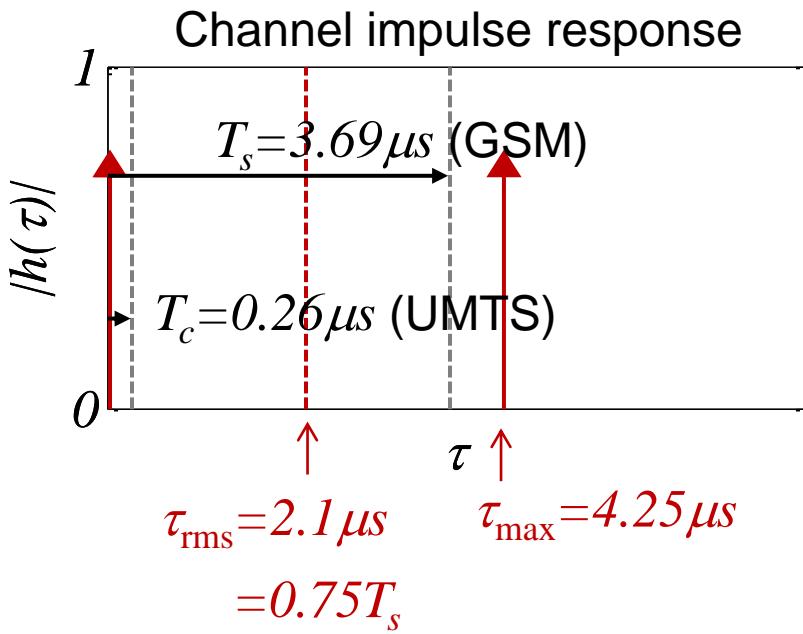
$$\tau_{rms} = \sqrt{\frac{\int_0^{\infty} (\tau - \bar{\tau})^2 |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}}$$

Mean channel delay:

$$\bar{\tau} = \frac{\int_0^{\infty} \tau |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}$$

The channel is considered to be frequency-selective, if  $T_s \ll \tau_{rms}$ .

# Root Mean Square (RMS) Delay Spread (2)



RMS delay spread:

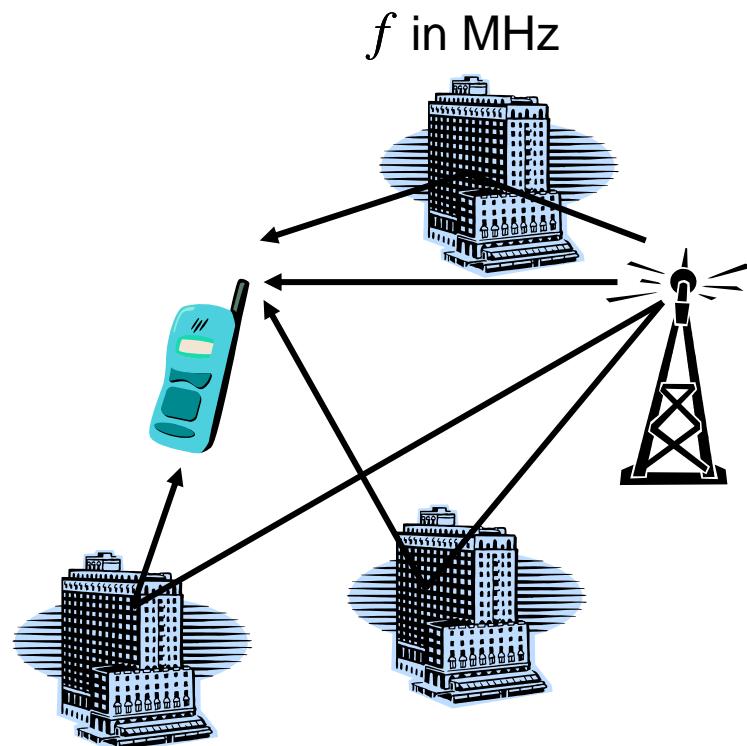
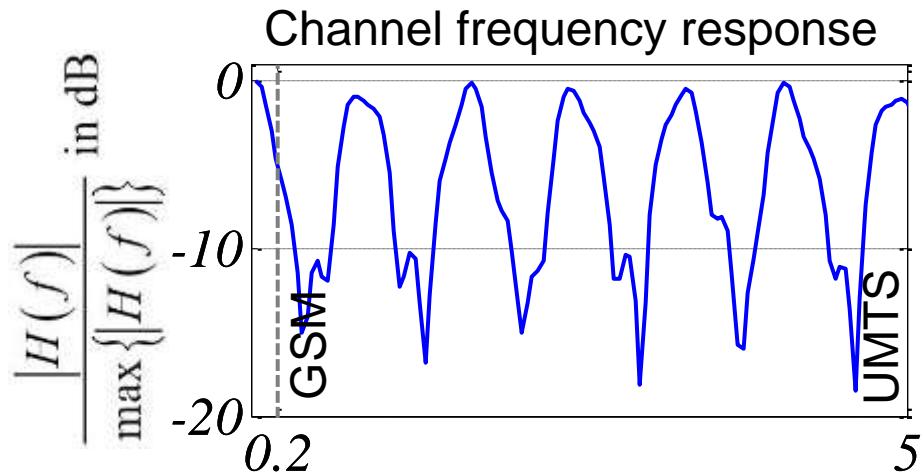
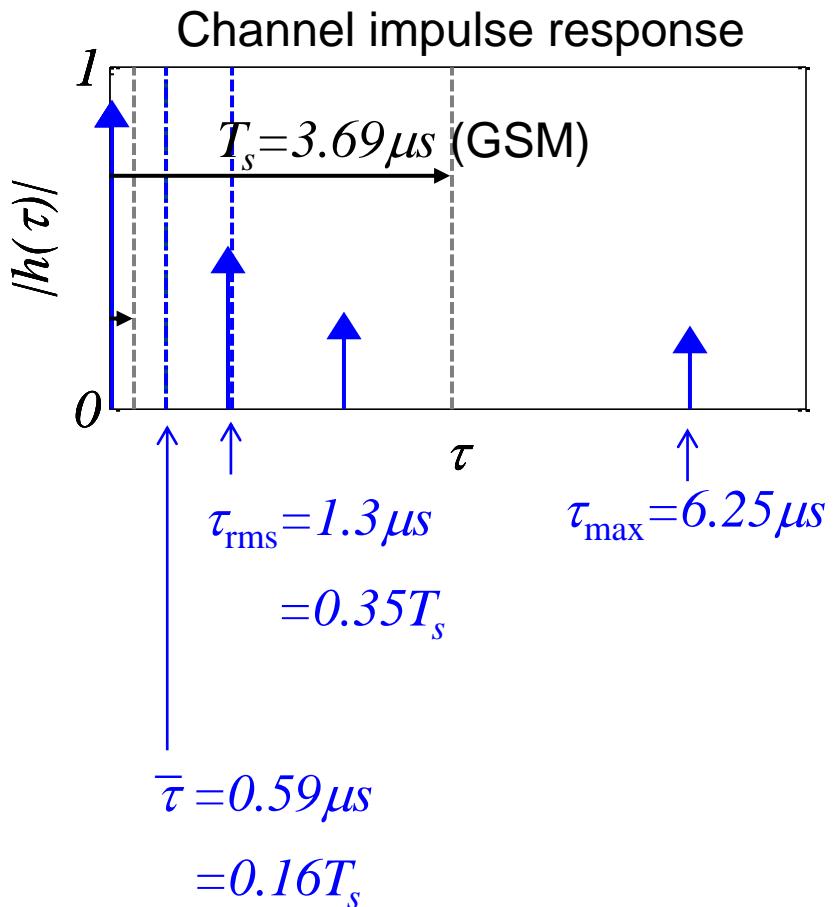
$$\tau_{rms} = \sqrt{\frac{\int_0^{\infty} (\tau - \bar{\tau})^2 |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}}$$

Mean channel delay:

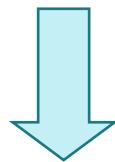
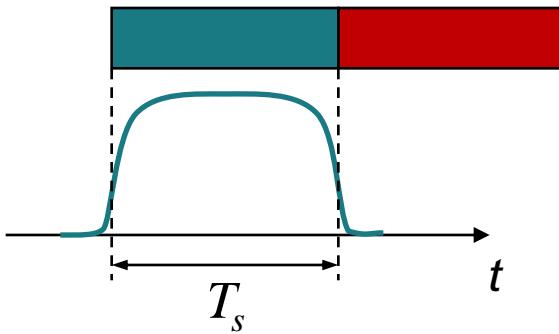
$$\bar{\tau} = \frac{\int_0^{\infty} \tau |h(\tau)|^2 d\tau}{\int_0^{\infty} |h(\tau)|^2 d\tau}$$

The channel is considered to be frequency-selective, if  $T_s \ll \tau_{rms}$ .

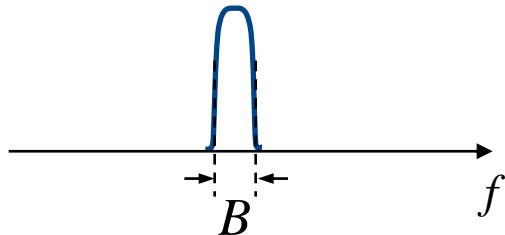
# Frequency-Selective Channels (7)



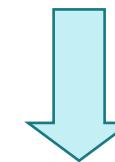
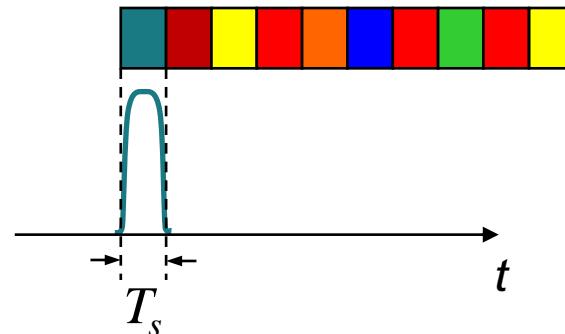
# Narrowband and Broadband Channels



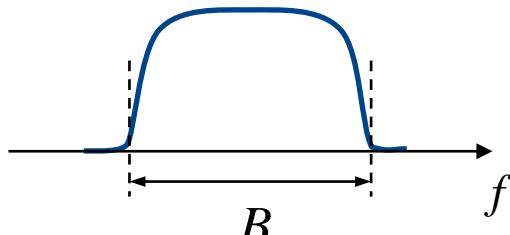
$$\tau_{\text{rms}} \ll T_s$$



narrowband channel



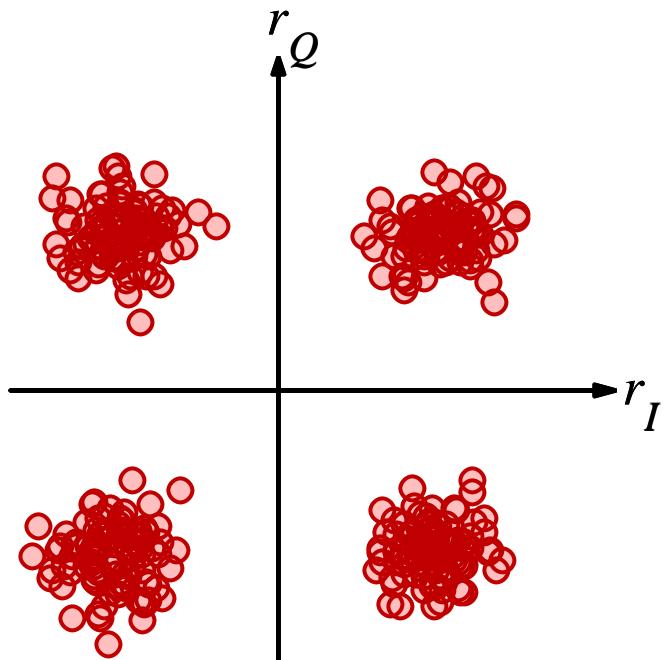
$$\tau_{\text{rms}} \gg T_s$$



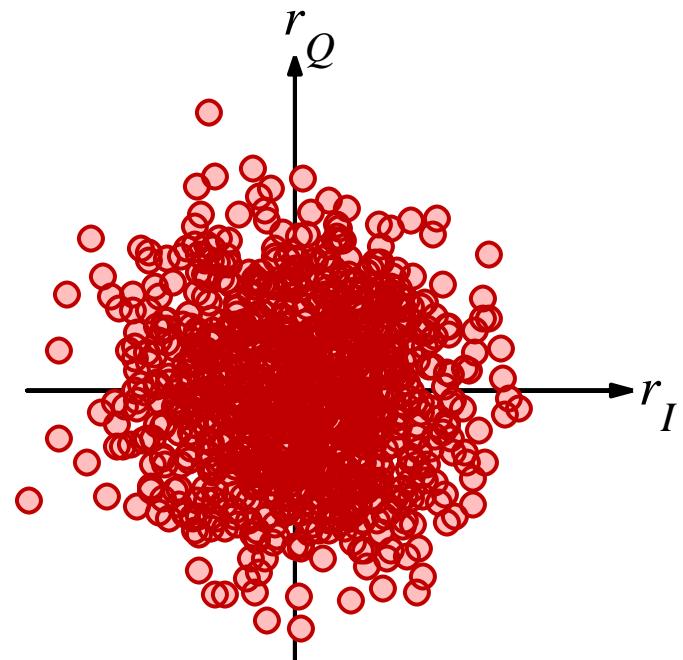
broadband channel

# QPSK Scatter Plot at Receiver with Intersymbol Interference (ISI)

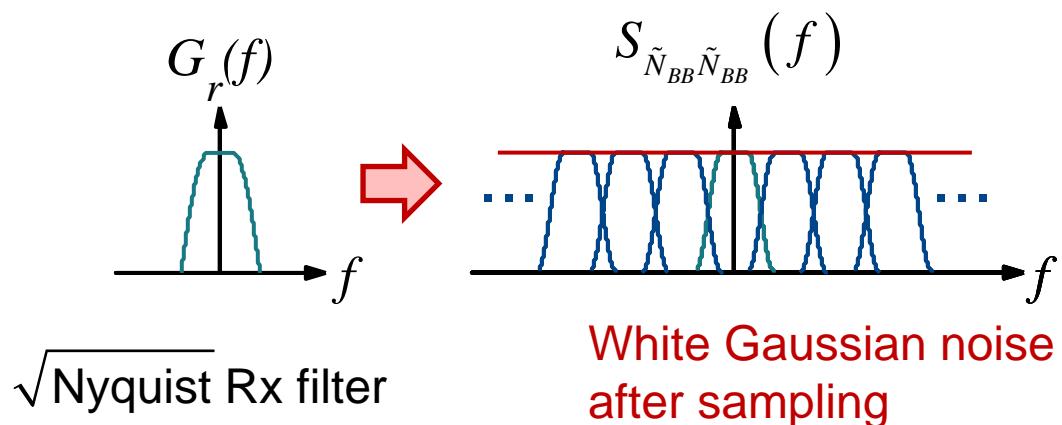
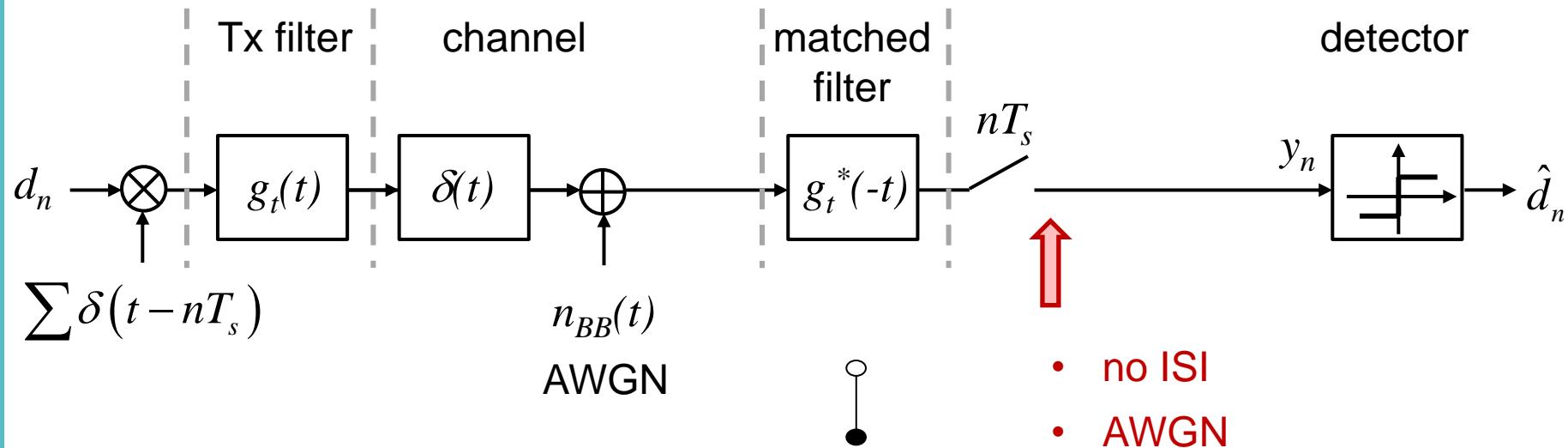
Frequency-flat channel  
(no ISI)



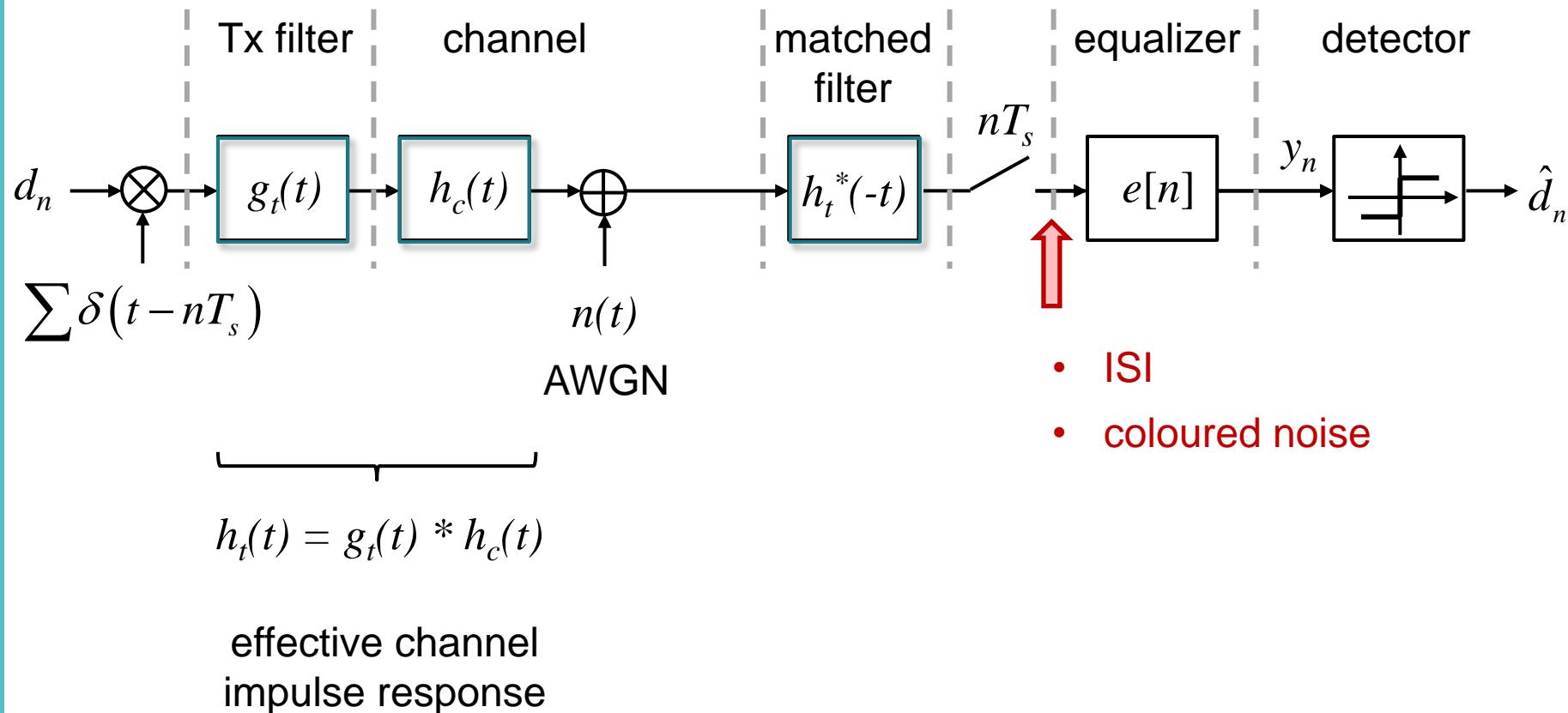
Frequency-selective channel  
(ISI channel)



# Equivalent Baseband Transmission Model- Frequency-Flat AWGN Channel

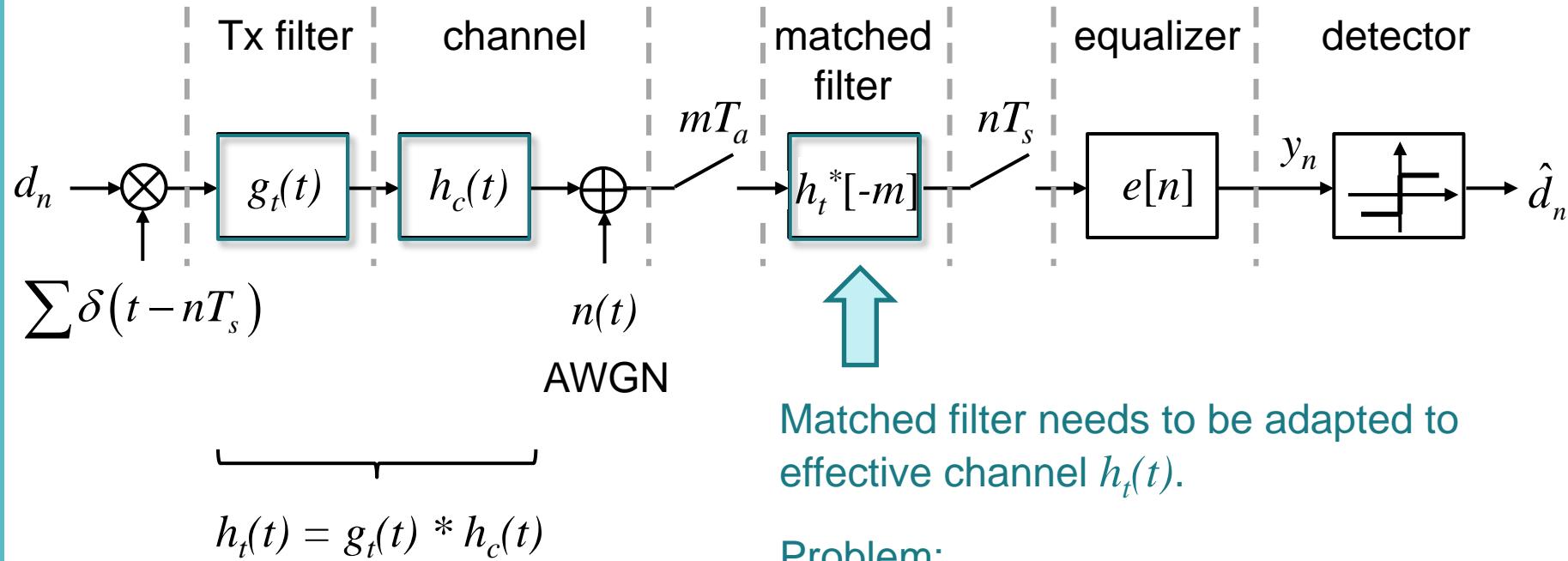


# Equivalent Baseband Transmission Model- Frequency-Selective AWGN Channel



An equalizer is a device which aims at removing the detrimental effect of intersymbol interference (ISI).

# Frequency-Selective AWGN Channel-Receive Filter Design (1)

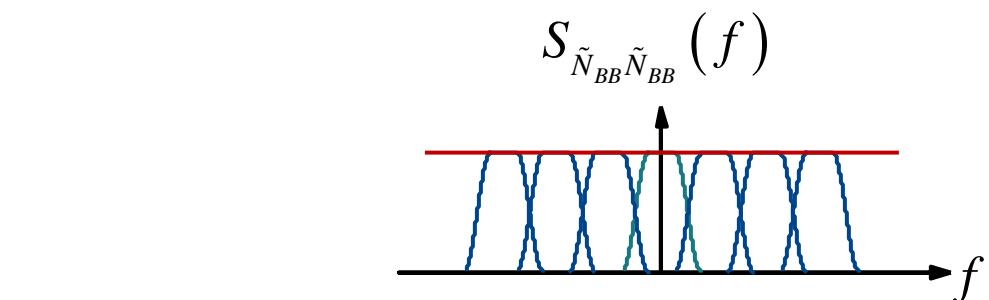
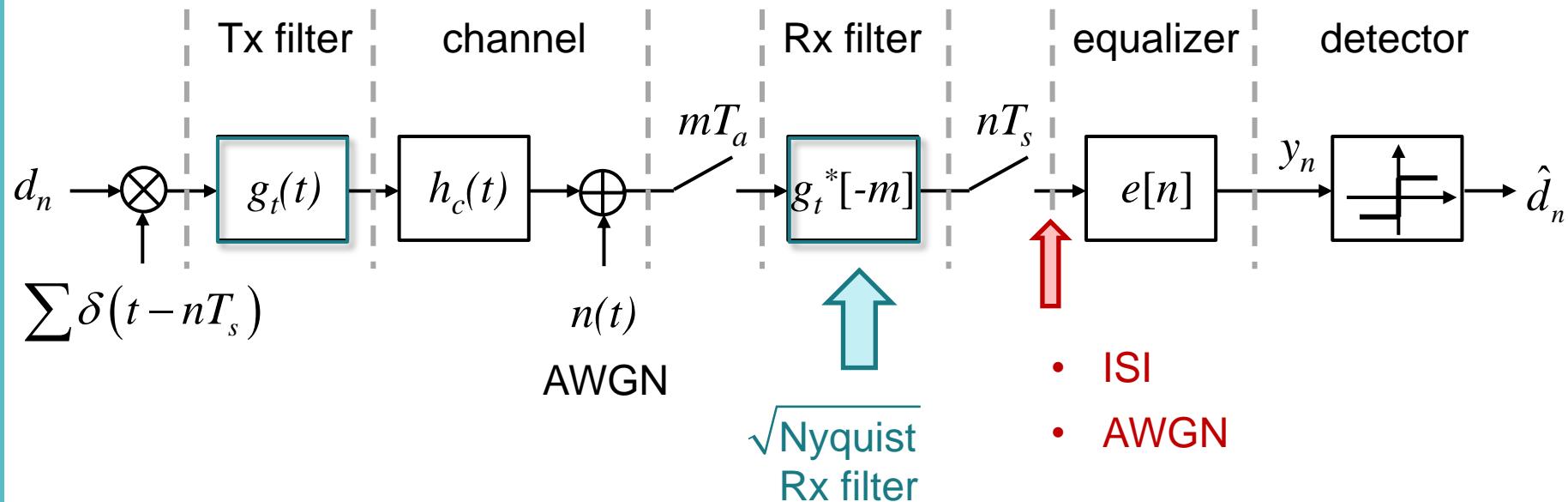


Problem:

Channel  $h_c(t)$  may be time-varying.

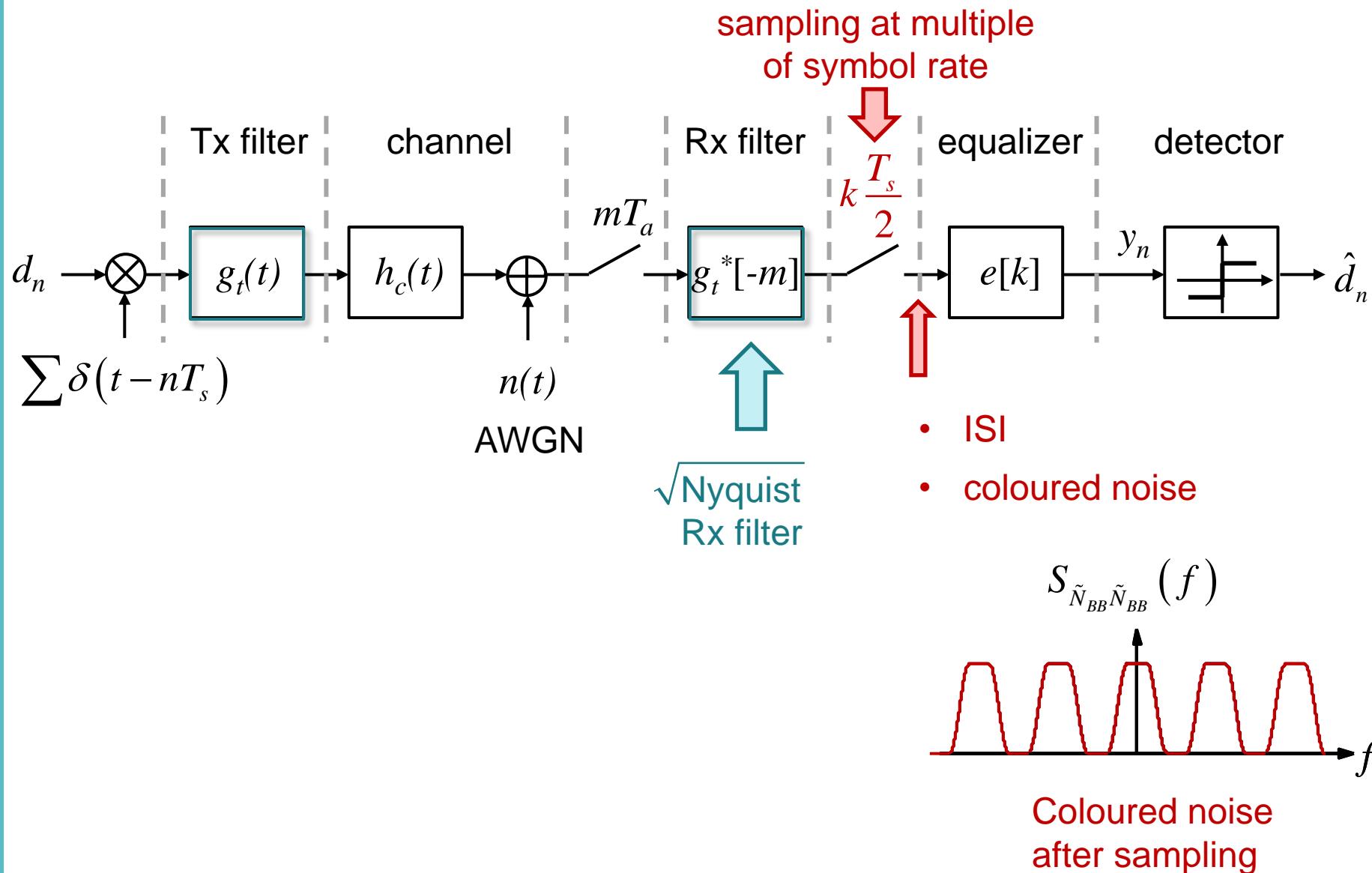
- Computationally expensive channel estimation at high sampling rate  $1/T_a$  which meets the sampling theorem.
- Compromise: Adapt Rx filter  $g_r(t)$  only to transmit filter  $g_t(t)$ .

# Frequency-Selective AWGN Channel-Receive Filter Design (2)

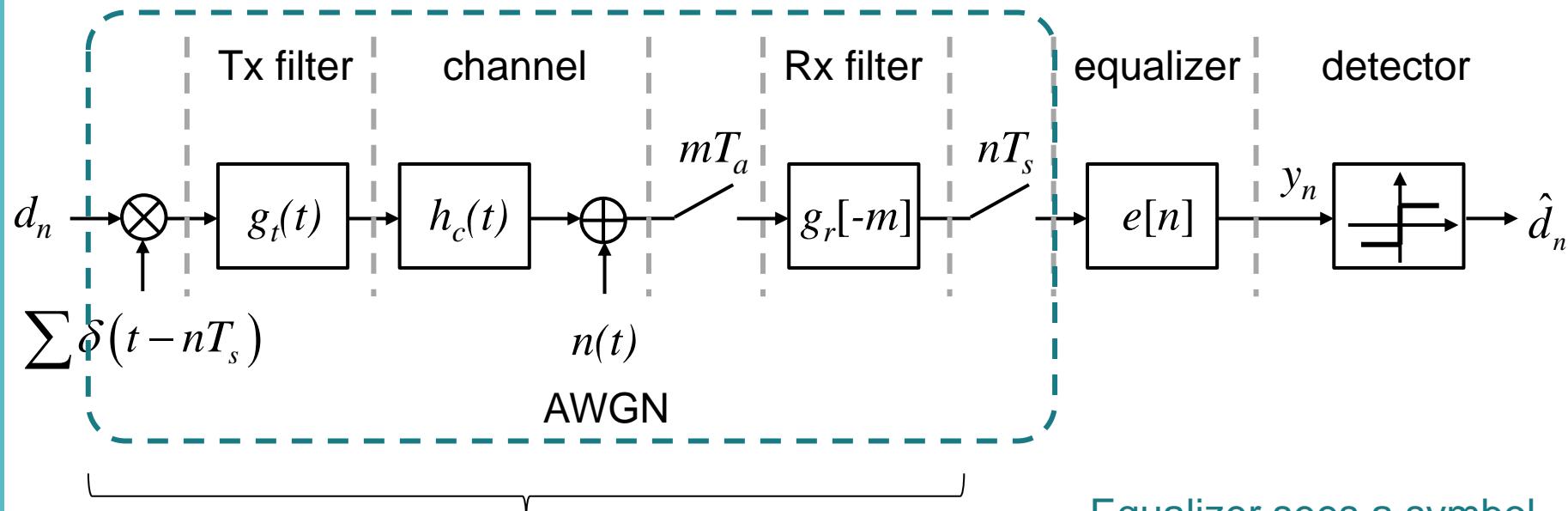


White Gaussian noise  
after sampling

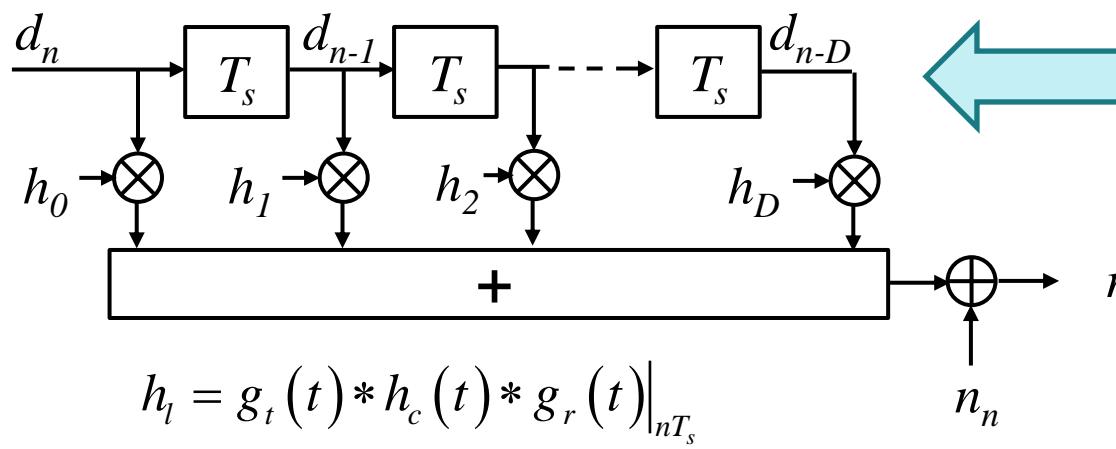
# Fractionally Spaced Equalizers



# Symbol-Spaced Discrete-Time Equivalent Baseband Transmission Model



Equalizer sees a symbol ( $T_s$ )-spaced discrete-time AWGN channel with ISI.

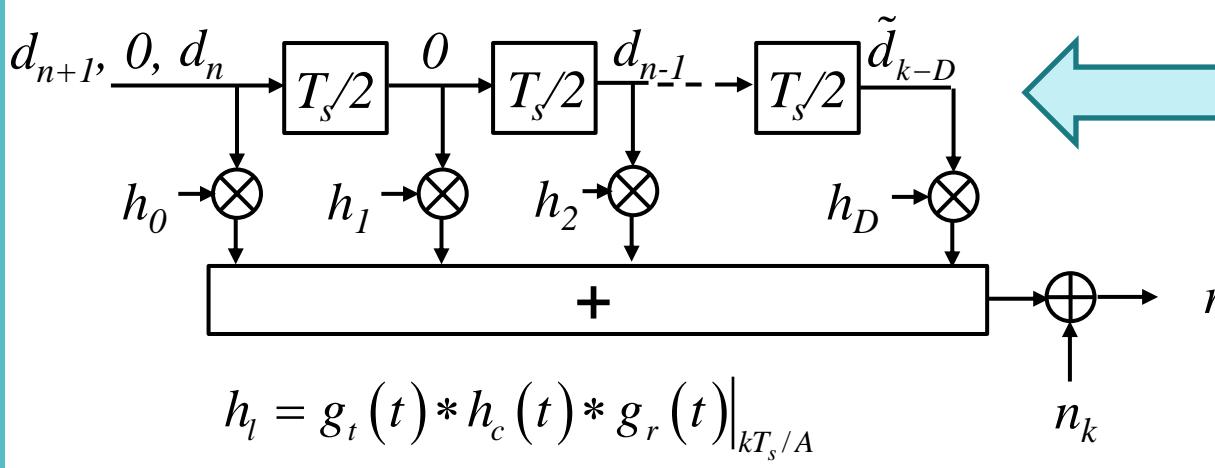
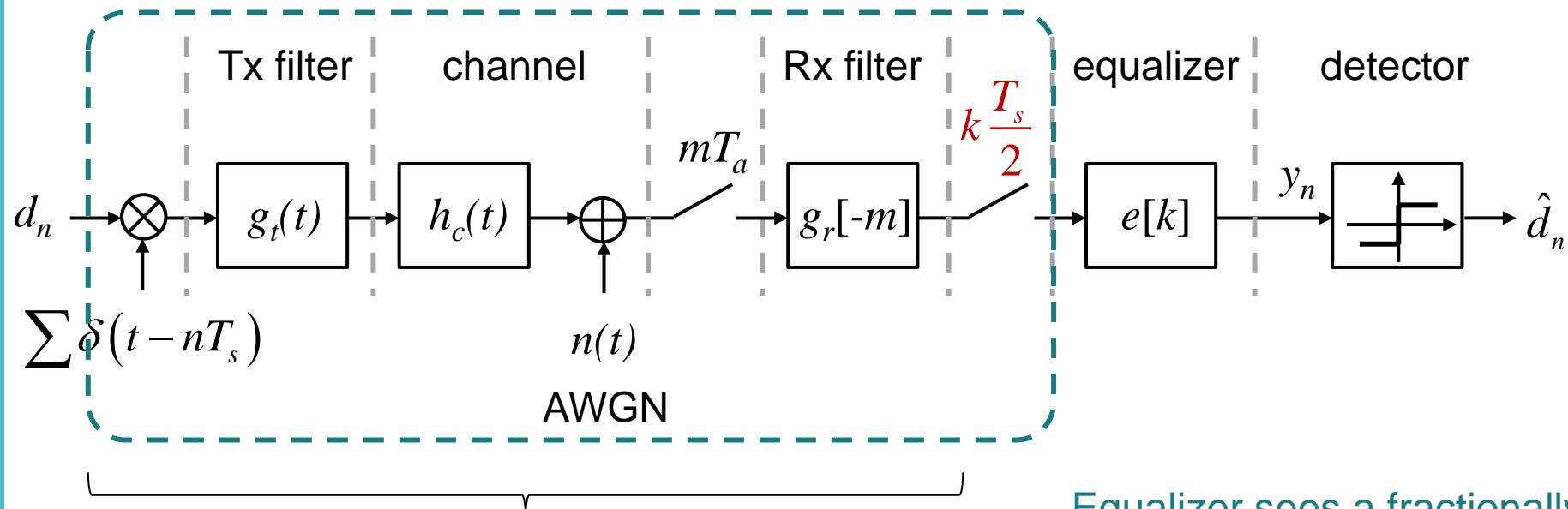


$$r_n = \left( \sum_{l=0}^D h_l d_{n-l} \right) + n_n$$

$$h_l = g_t(t) * h_c(t) * g_r(t) \Big|_{nT_s}$$

D: channel memory

# Fractionally Spaced Discrete-Time Equivalent Baseband Transmission Model



Equalizer sees a fractionally (e.g.  $T_s/2$ )-spaced discrete-time AWGN-ISI channel.

$$r_k = \left( \sum_{l=0}^D h_l d_{k-l} \right) + n_k$$

$$h_l = g_t(t) * h_c(t) * g_r(t) \Big|_{kT_s/A}$$

D: channel memory

# Equalization: Inverse System

For perfect compensation of linear channel distortions, the transfer function of the equalizer should invert the transfer function of the channel.

The transfer function of a discrete-time channel or equalizer, respectively, can be characterized by the pole-zero plot in the complex z-plane.

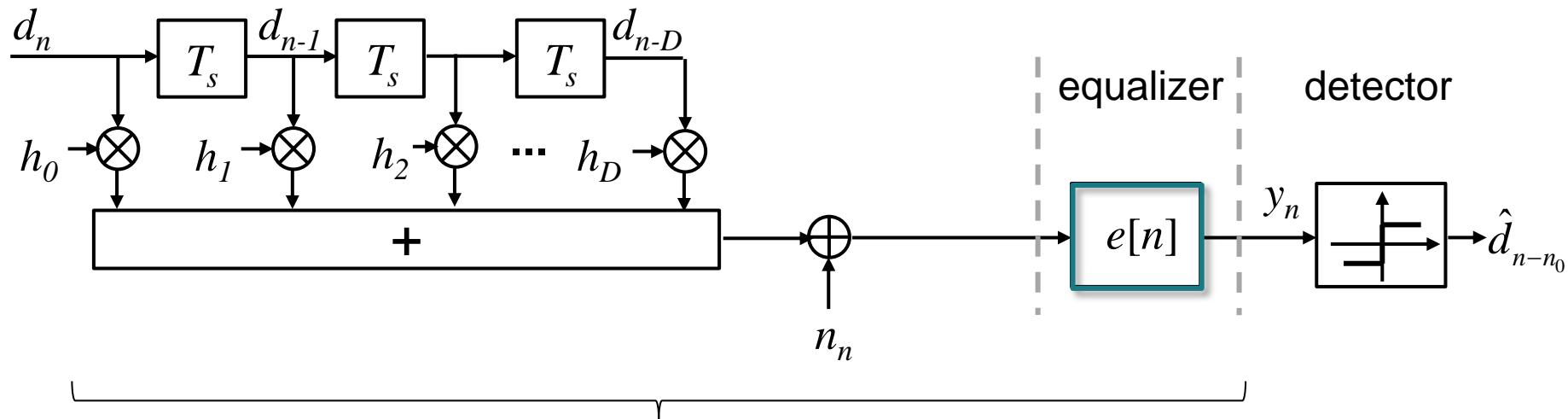
The equalizer should compensate the zeros and poles of the channel except for those in the origin of the z-plane as poles and zeros in the origin correspond to a delay only and make sure that the system is causal.

Hence, the equalizer transfer function should have zeros at the positions, where the channel transfer function has a zero. Analogously, it should have poles at the positions, where the channel transfer function has zeros.

Most channels can be modelled as non-recursive system and, therefore, have poles only in the origin of the z-plane which do not need to be eliminated.

Compensating zeros of the channel transfer function by placing poles of the equalizer transfer function at the same positions is critical since poles outside the unit circle cause instability of the system. Therefore, only minimum phase channels, which have zeros only inside the unit circle, can be inverted by a stable equalizer. However, the channel transfer function cannot be arbitrarily influenced by the system designer and, hence, we cannot assume a minimum phase channel in general. Therefore, the perfect inverse system needs to be approximated by a non-recursive system, i.e. by a finite impulse response (FIR) filter. As FIR filters have poles in the origin of the z-plane only, they are always stable. However, a perfect equalization is not possible in general using a non-recursive (FIR) equalizer of finite length.

# Equalization: Inverse System (1)

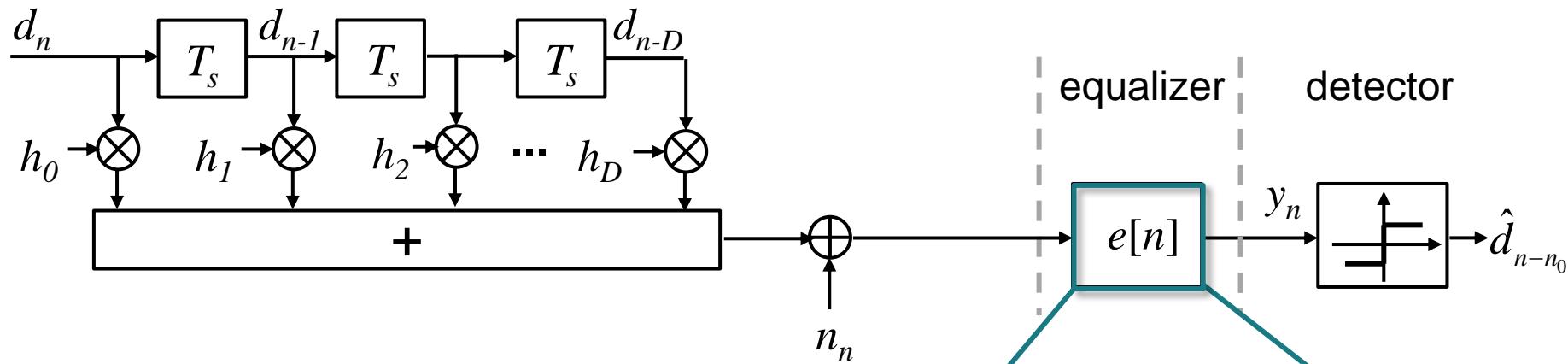


$$\text{Target: } h[n] * e[n] = \delta[n - n_0]$$

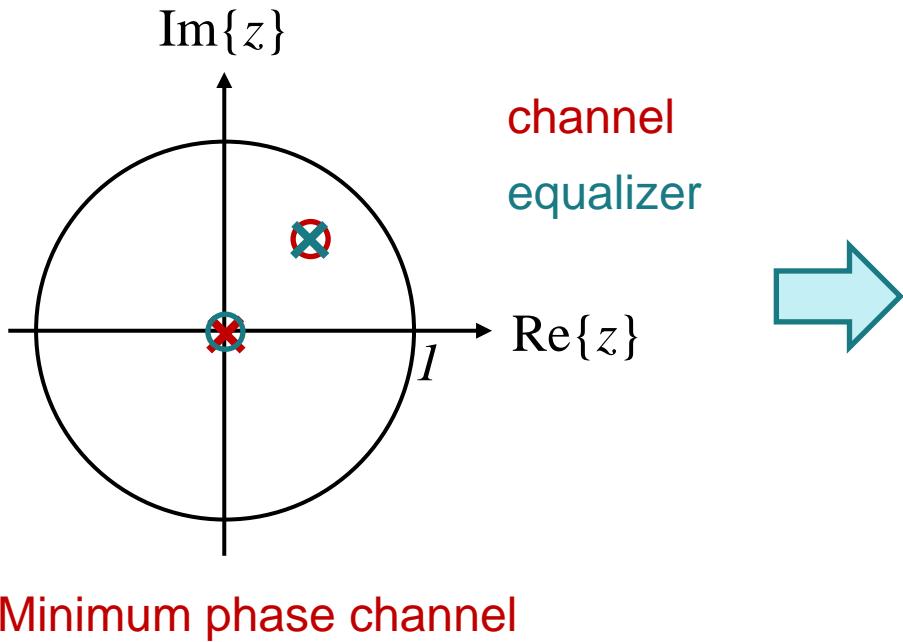


equalization processing  
delay is allowed

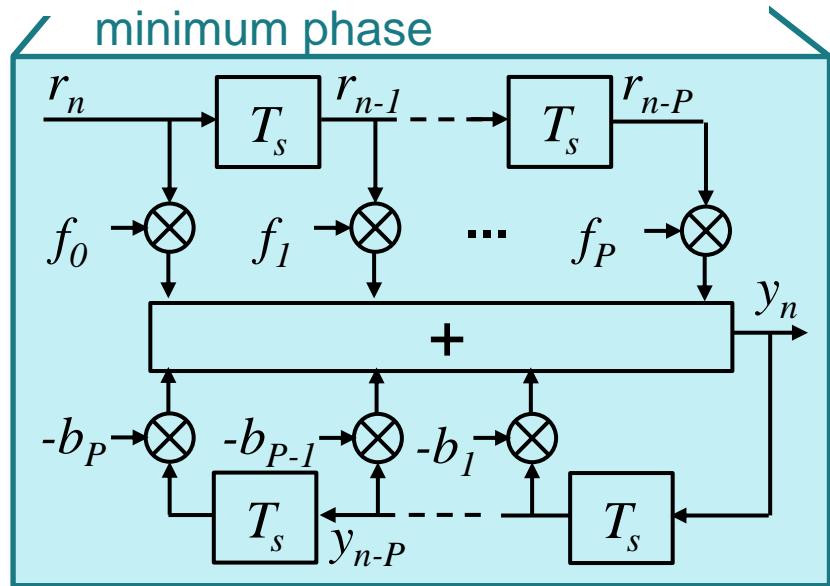
# Equalization: Inverse System (2)



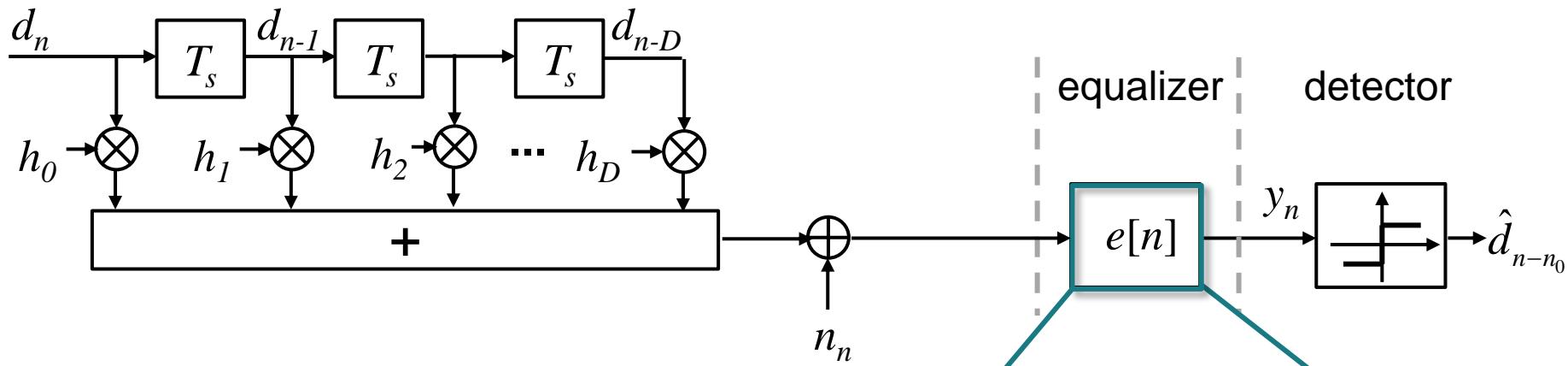
Pole-zero plot:



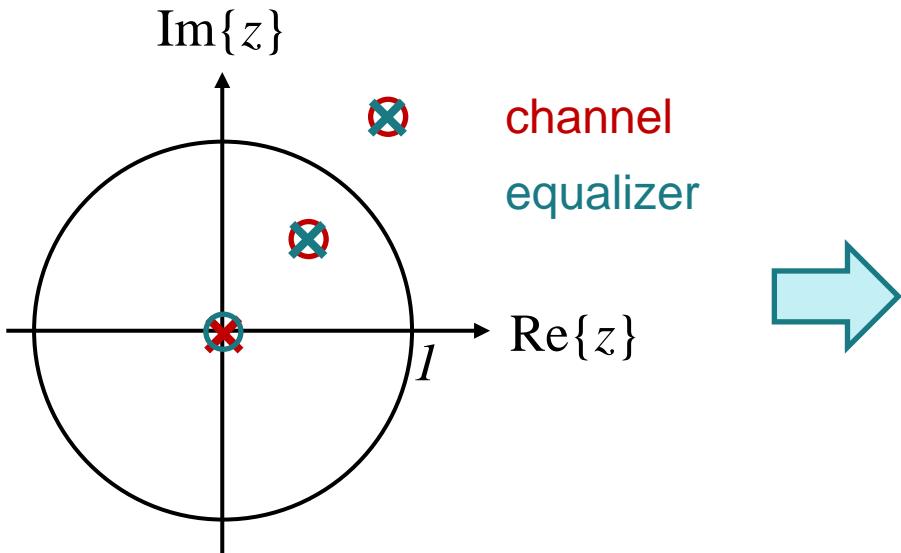
Equalizer must be a recursive system  
→ stability problem if channel is not  
minimum phase



# Equalization: Inverse System (3)

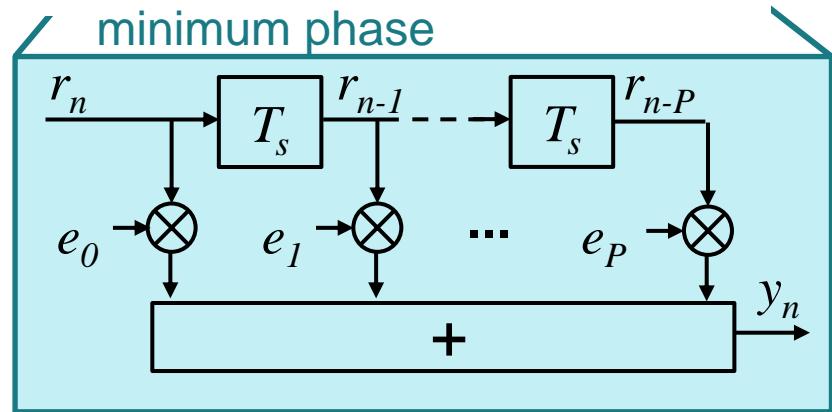


Pole-zero plot:



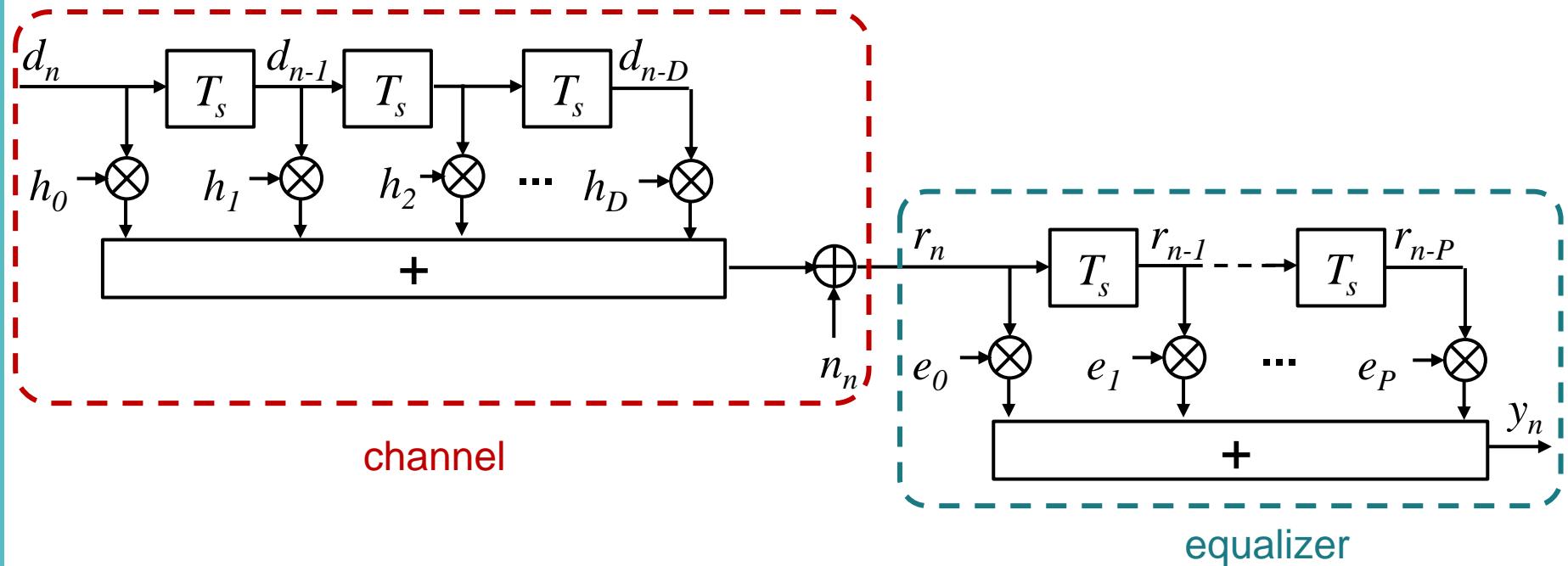
channel  
equalizer

Equalizer must be a recursive system  
→ stability problem if channel is not  
minimum phase



→ approximation by non-recursive  
(FIR) equalizer

# Non-Recursive Linear Equalizer (1)



# Possibility of Perfect Equalization with Fractionally Spaced Non-Recursive Linear Equalizers (1)

The target of the equalizer is to compensate for linear distortions which occur during transmission. We consider a non-recursive linear equalizer, i.e. an equalizer which can be implemented as an FIR filter of order  $P$ . The total impulse response of the discrete-time equivalent baseband system including transmit and receive filters, channel and equalizer is given by the discrete convolution  $h[k] * e[k]$ , where the impulse response  $h[k]$  of length  $D+1$  models the concatenation of transmit filter, channel and receive filter, while  $e[k]$  is the equalizer impulse response of length  $P+1$ .

The equalizer may be a fractionally spaced equalizer which operates on the sampling rate  $1/T_a$  which is a multiple of the symbol rate  $1/T_s = 1/(AT_a)$ . As we are interested only in interference-free data symbols at the equalizer output, the equalizer output is downsampled by a factor  $A$  to the symbol rate. The downsampled equalizer output should ideally be free of intersymbol interference, i.e. the design target for the equalizer is given by

$$h[k]*e[k]\Big|_{k=nA} = \delta[n - n_0],$$

where we allow for an equalizer processing delay  $n_0$  in order to relax the equalization problem.

The result of the discrete convolution  $h[k] * e[k]$  has length  $D+P+1$ . Hence, after downsampling by a factor of  $A$  at the equalizer output, we obtain  $(D+P+1)/A$  samples, which should be free of intersymbol interference. These conditions yield a set of equations for determination of the  $P+1$  equalizer coefficients.

# Possibility of Perfect Equalization with Fractionally Spaced Non-Recursive Linear Equalizers (2)

In order to avoid an overdetermined system of equations,

$$\frac{D + P + 1}{A} = P + 1$$

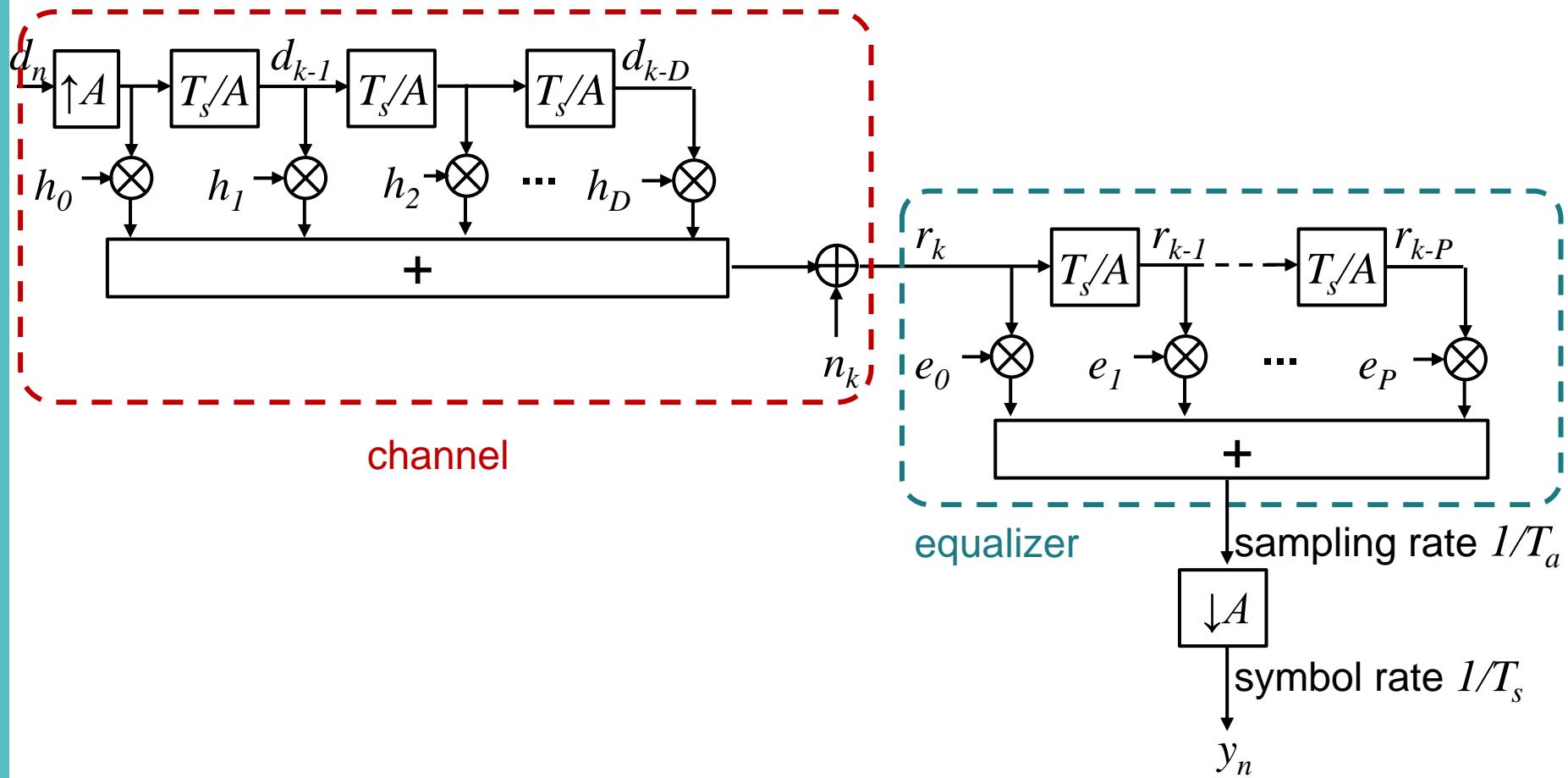
must hold. Consequently, the number of required equalizer coefficients is given by

$$P = \frac{D + 1 - A}{A - 1}.$$

It can be concluded, that perfect equalization with a symbol-spaced ( $A=1$ ) non-recursive linear equalizer requires in general an infinite equalizer order  $P$  and, consequently, cannot be realized. On the other hand, a fractionally spaced ( $A>1$ ) equalizer can in principle yield perfect equalization. This can be explained by the fact that a fractionally spaced equalizer has more degrees of freedom, as only the remaining equalizer outputs after downsampling determine the equalization condition, while the other equalizer output samples before downsampling can have arbitrary values. Hence, the equalizer does not need to invert perfectly the fractionally spaced channel.

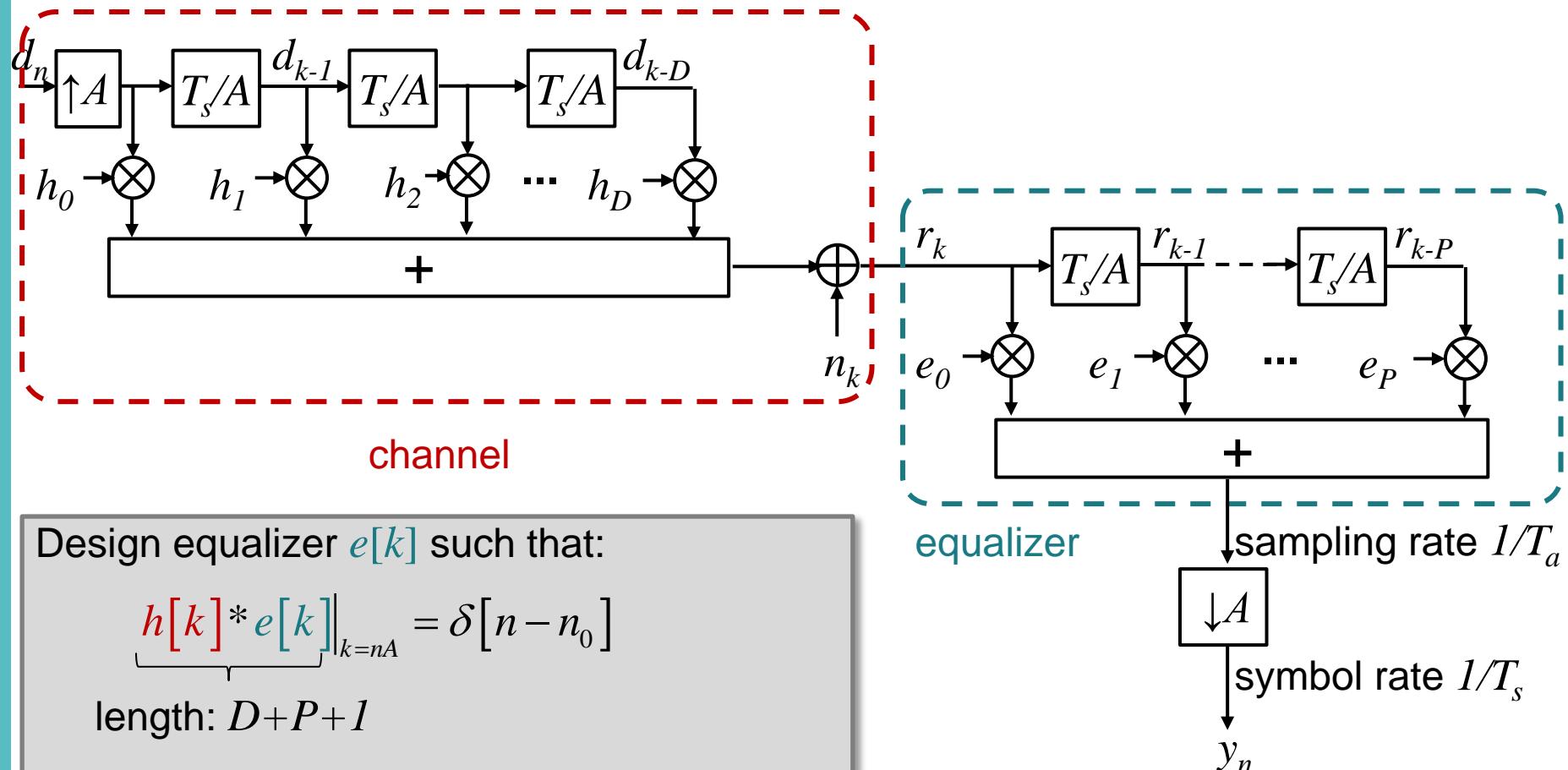
A disadvantage of fractionally spaced equalizers is that more signal processing effort is required as it operates on a higher sampling rate. Moreover, the channel coefficients  $h_k$  need to be estimated at a higher sampling rate compared to a symbol-spaced equalizer. Therefore, in practice symbol-spaced equalizers are more common even though they do not allow for perfect equalization.

# Possibility of Perfect Equalization with Fractionally Spaced Non-Recursive Linear Equalizers (1)



- A perfect symbol-spaced ( $A=I$ ) non-recursive linear equalizer would require an infinite equalizer order  $P$ .
- Perfect equalization is in principle possible with a fractionally spaced equalizer ( $A>I$ ).

# Possibility of Perfect Equalization with Fractionally Spaced Non-Recursive Linear Equalizers- Derivation



Design equalizer  $e[k]$  such that:

$$\underbrace{h[k]^* e[k]}_{k=nA} = \delta[n - n_0]$$

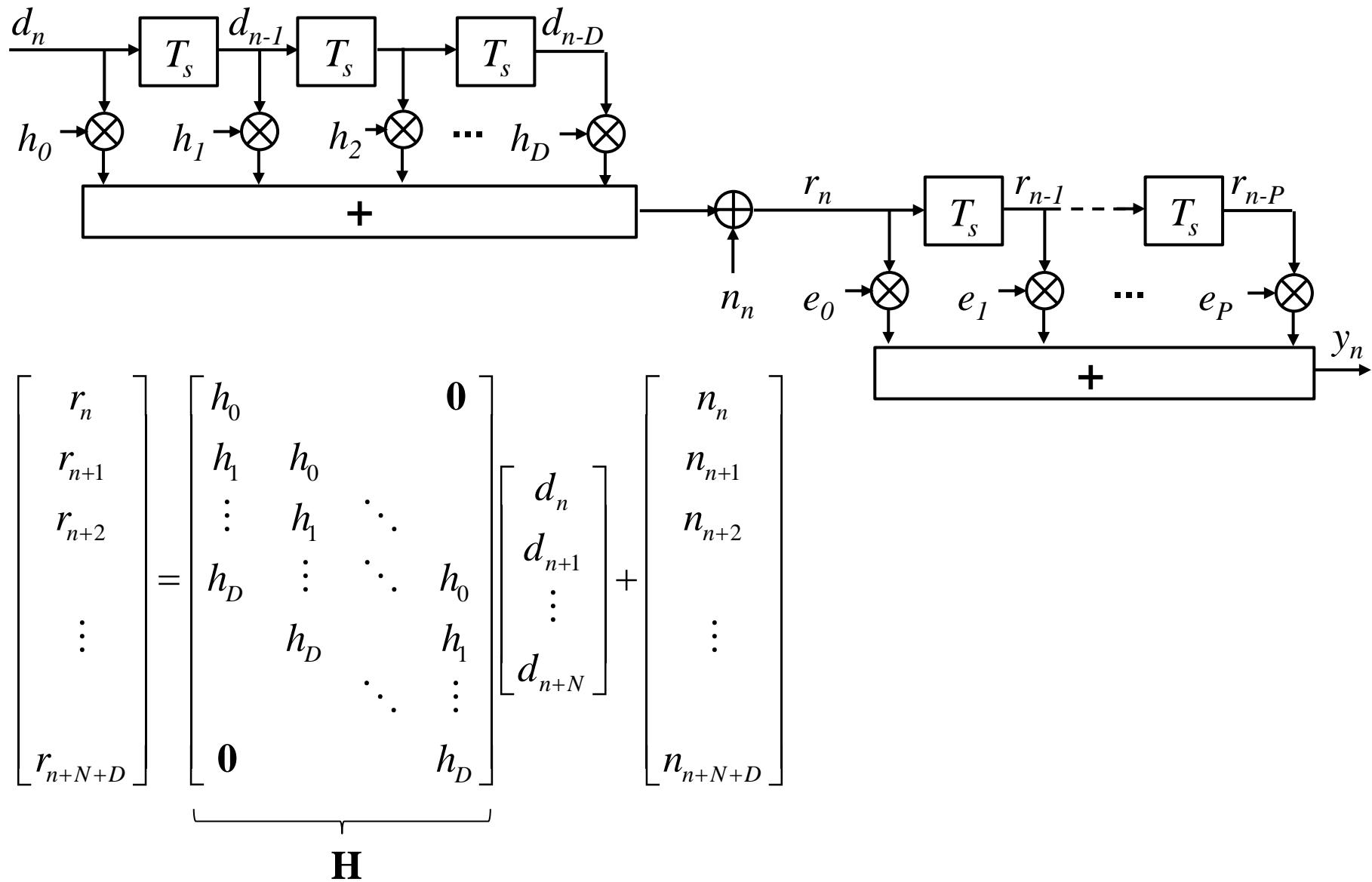
length:  $D+P+1$

$\rightarrow \frac{D+1+P}{A}$  conditions for determination  
of  $P+1$  equalizer coefficients

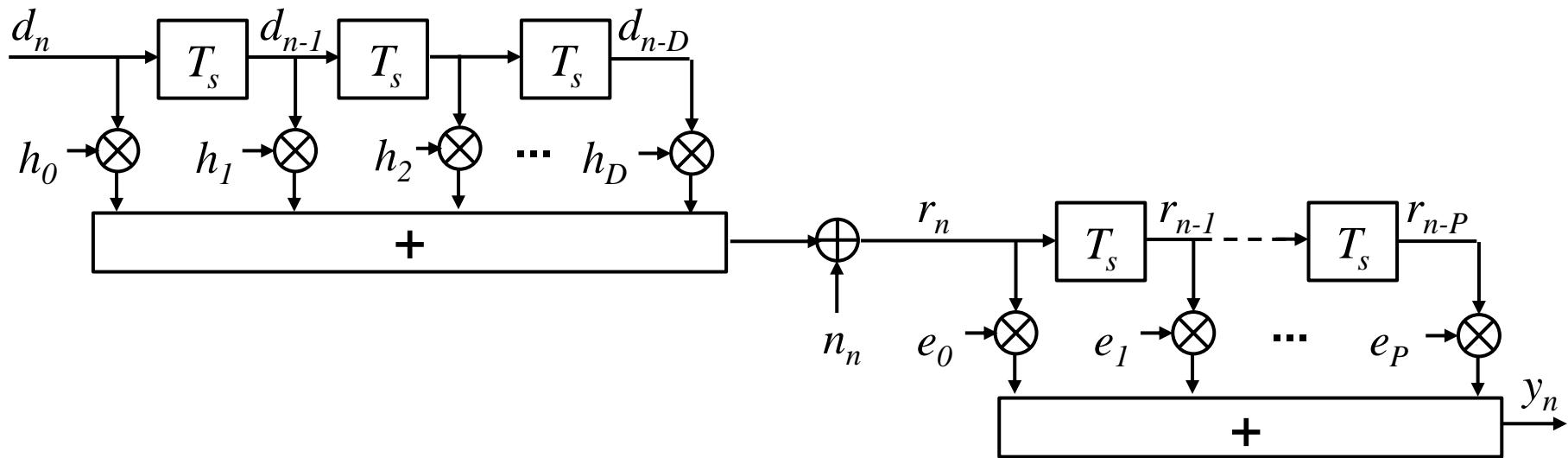
$$\rightarrow \frac{D+1+P}{A} = P+1 \quad \Rightarrow P = \frac{D+1-A}{A-1}$$

$$\Rightarrow \begin{cases} P \rightarrow \infty & \text{for } A = 1 \\ P < \infty & \text{for } A > 1 \end{cases}$$

# Non-Recursive Linear Equalizer-Convolution Matrix (1)



# Non-Recursive Linear Equalizer-Convolution Matrix (2)



$$\begin{bmatrix} r_n \\ r_{n-1} \\ \vdots \\ r_{n-P} \end{bmatrix} = \underbrace{\begin{bmatrix} h_0 & h_1 & \cdots & h_D \\ h_0 & h_1 & \cdots & h_D \\ \ddots & \ddots & \ddots & \ddots \\ h_0 & h_1 & \cdots & h_D \end{bmatrix}}_{\mathbf{H}^T} \underbrace{\begin{bmatrix} d_n \\ d_{n-1} \\ \vdots \\ d_{n-D-P} \end{bmatrix}}_{\mathbf{d}} + \underbrace{\begin{bmatrix} n_n \\ n_{n-1} \\ \vdots \\ n_{n-P} \end{bmatrix}}_{\mathbf{n}}$$

$\mathbf{r}$                $\mathbf{H}^T$                $\mathbf{d}$                $\mathbf{n}$

# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (1)

A zero forcing equalizer aims at eliminating the intersymbol interference completely, i.e. it forces the intersymbol interference to zero. The resulting impulse response of the overall system including transmit filter, channel, receive filter and equalizer should yield

$$h[n]^* e[n] = \delta[n - n_0],$$

where  $n_0$  is the processing delay which can be chosen such that the best equalizer performance is obtained. The equalization problem can be formulated in vector-matrix notation as

$$\mathbf{H}\mathbf{e} = \mathbf{i},$$

where  $\mathbf{H}$  denotes the convolution matrix of the channel, the vector  $\mathbf{e}$  contains the equalizer coefficients and  $\mathbf{i}$  is a vector which contains an entry „1“ in the  $n_0$ -th row while all other entries are zero.

As it is impossible to perfectly force the interference to zero using a symbol-spaced non-recursive linear equalizer of finite length, we approximate a zero-forcing equalizer according to the least squares criterion: We allow for an equalization error

$$\Delta = \mathbf{H}\mathbf{e} - \mathbf{i}$$

and design the equalizer taps  $\mathbf{e}$  such that the energy  $\Delta^H \Delta$  of the equalization error is minimized.

# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (2)

The least squares solution yields the expression

$$\mathbf{e}_{ZF} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}$$

for the equalizer taps e. i.e., the equalizer is determined by the Moore-Penrose pseudo inverse of the channel matrix  $\mathbf{H}$ . The Moore-Penrose pseudo inverse allows to invert non-square matrices. The vector  $\mathbf{i}$  chooses a particular column of the Moore-Penrose pseudo inverse. It is useful to choose the row which results in the smallest equalization error energy  $\Delta^H \Delta$ .

A zero-forcing equalizer tries to invert the channel transfer function as good as possible given a certain equalizer order  $P$ . However, it does not take the additive noise into account.

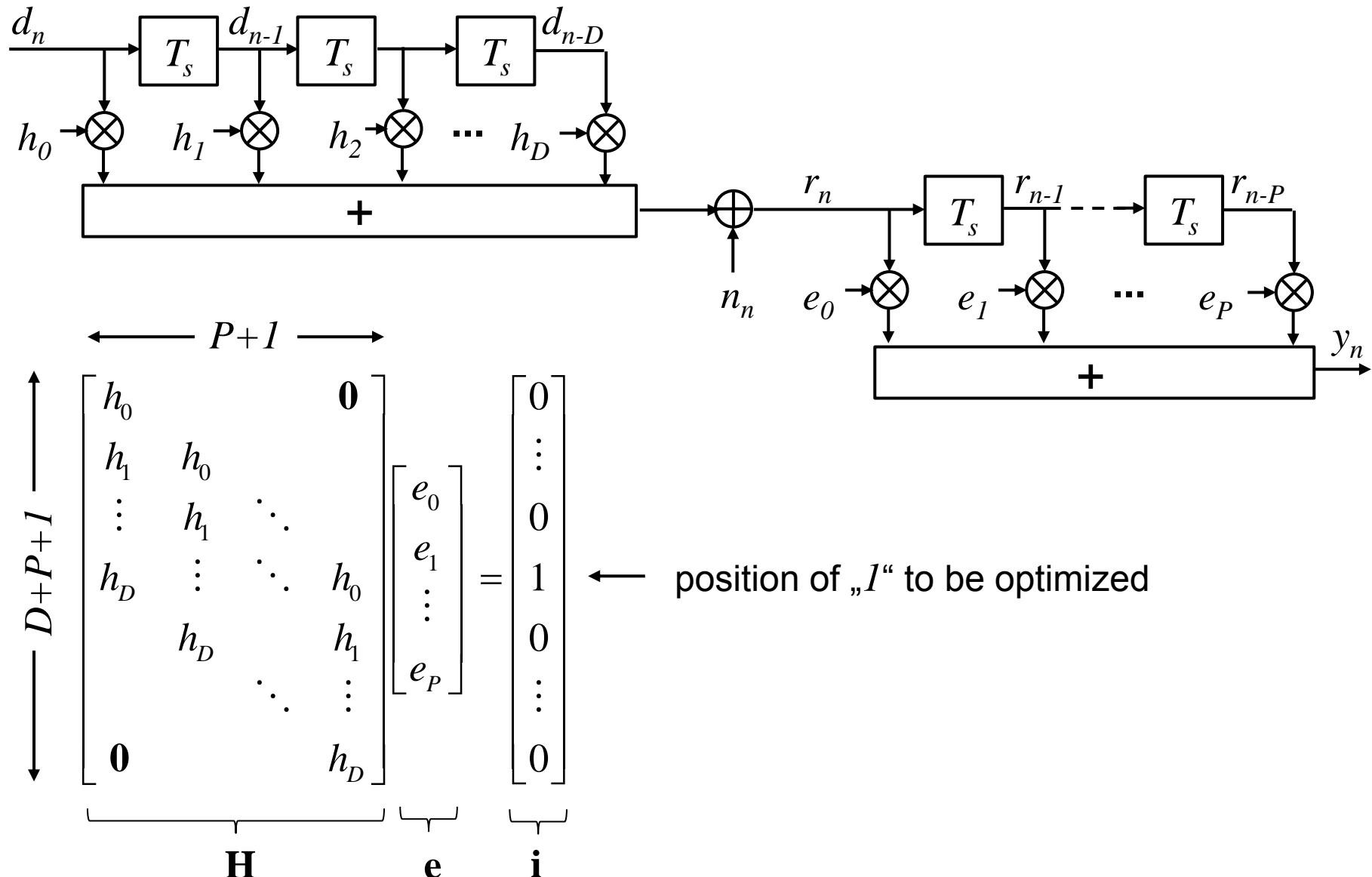
The major problem of zero-forcing equalization in noisy channels is that it typically results in significant noise enhancement. Especially in channels with a strong attenuation of the transfer function at certain frequencies (node frequencies), inverting the channel transfer function requires strong amplification of the received signal at those frequencies by the equalizer. However, this not only amplifies the desired signal but also the additive noise. As a result, it may not be possible to detect the data signal at the equalizer output due to detrimental noise enhancement.

Therefore, often another equalizer design criterion, the minimum mean squared error (MMSE) criterion, is applied, which aims at an optimized compromise between perfect equalization and noise enhancement. At low SNR, an MMSE equalizer mainly reduces noise enhancement while a significant amount of intersymbol interference (ISI) remains at the equalizer output. However, the remaining ISI is less harmful than the noise enhancement of a zero-forcing equalizer would be.

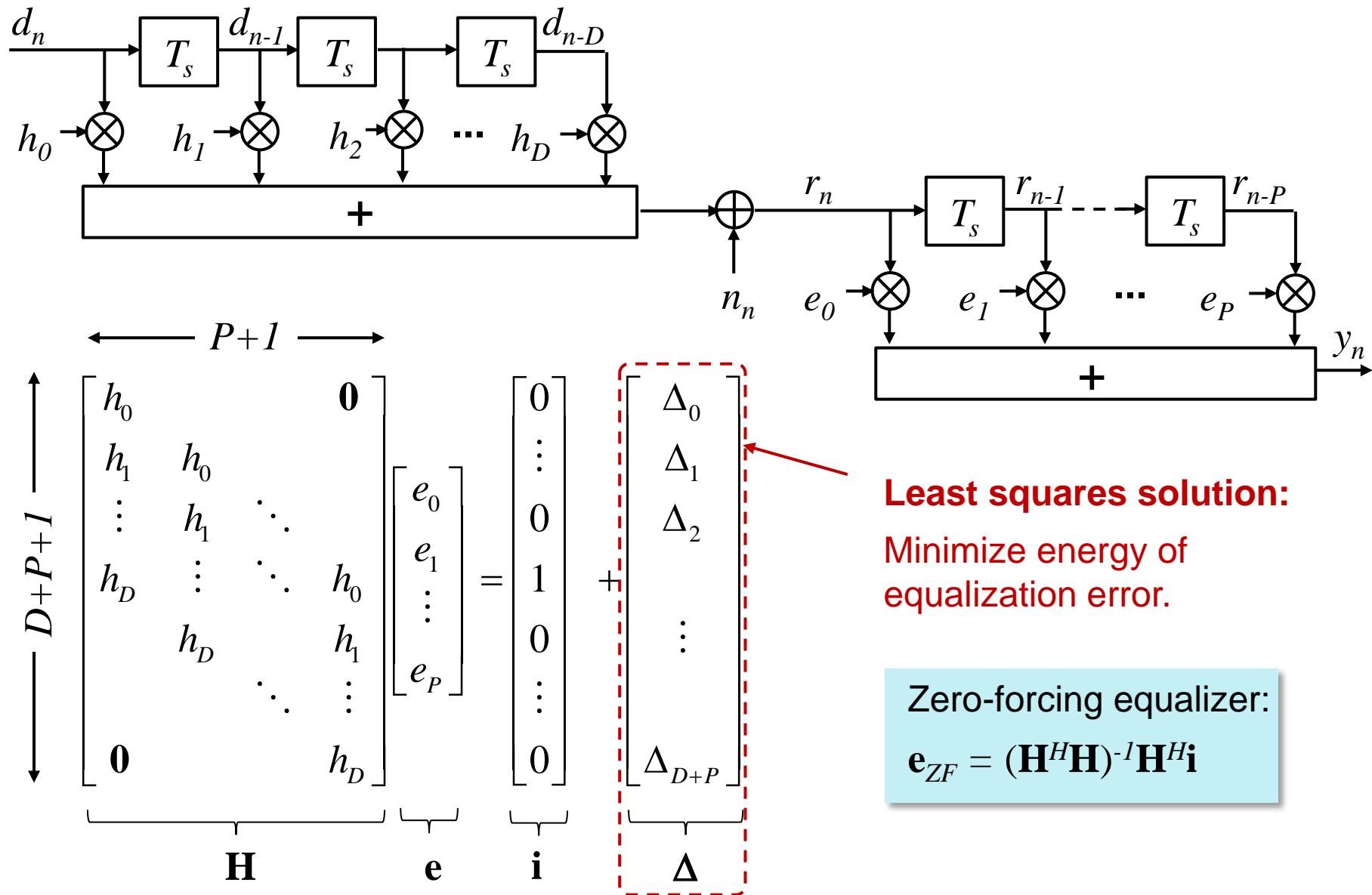
# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (3)

At high SNR, where the noise power is low, an MMSE equalizer will mainly suppress ISI as noise enhancement is less critical. Therefore, at high SNR, zero-forcing and MMSE converge to the same solution.

# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (1)



# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (2)



**Least squares solution:**  
Minimize energy of  
equalization error.

$$\mathbf{e}_{ZF} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}$$

# Non-Recursive

## Linear Zero-Forcing Equalizer - Derivation

Target:  $\mathbf{H} \mathbf{e} = \mathbf{i} + \Delta$

Equalization error:  $\Delta = \mathbf{H} \mathbf{e} - \mathbf{i}$

**Least squares solution:** Minimize energy of equalization error:

$$\mathbf{e}_{ZF} = \arg \min_{\mathbf{e}} \left\{ \Delta^H \Delta \right\} = \arg \min_{\mathbf{e}} \left\{ \begin{bmatrix} \Delta_0^* & \cdots & \Delta_{D+P}^* \end{bmatrix} \begin{bmatrix} \Delta_0 \\ \vdots \\ \Delta_{D+P} \end{bmatrix} \right\} = \arg \min_{\mathbf{e}} \left\{ \sum_{k=0}^{D+P} |\Delta_k|^2 \right\}$$

$$\Delta^H \Delta = (\mathbf{e}^H \mathbf{H}^H - \mathbf{i}^H) (\mathbf{H} \mathbf{e} - \mathbf{i}) = \mathbf{e}^H \mathbf{H}^H \mathbf{H} \mathbf{e} - \mathbf{e}^H \mathbf{H}^H \mathbf{i} - \mathbf{i}^H \mathbf{H} \mathbf{e} + \underbrace{\mathbf{i}^H \mathbf{i}}_I$$

$$\frac{d\Delta^H \Delta}{d\mathbf{e}^H} = \mathbf{H}^H \mathbf{H} \mathbf{e} - \mathbf{H}^H \mathbf{i} = \mathbf{0}$$

↑

Wirtinger calculus

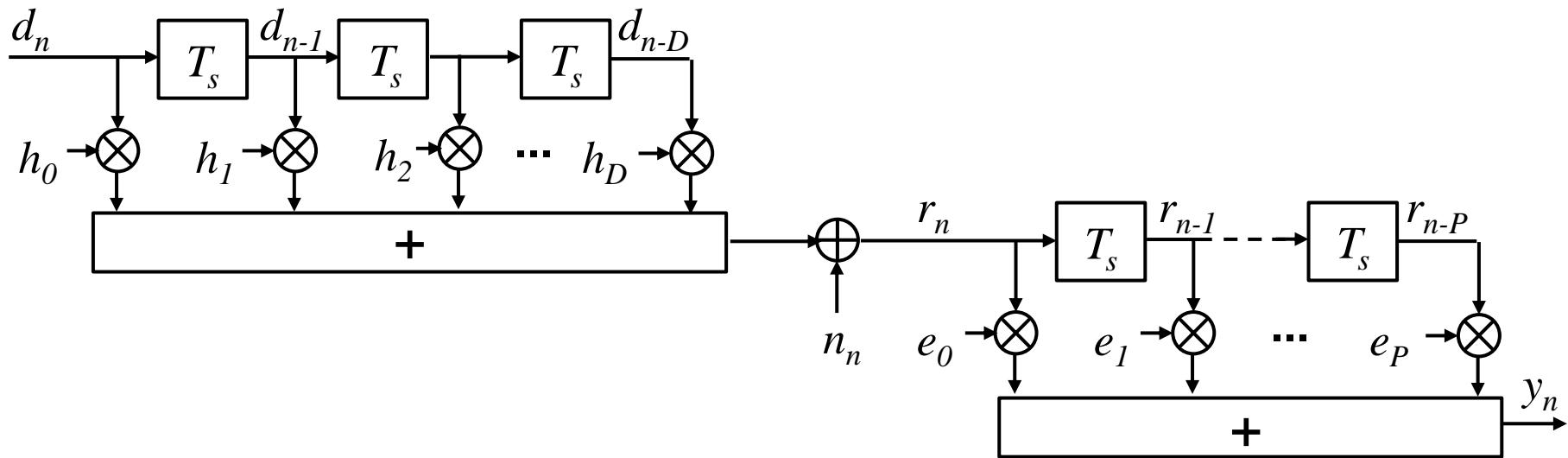
$$\underbrace{(\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{H} \mathbf{e}}_{\mathbf{I}_{P+1}} = \underbrace{(\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}}$$

Moore-Penrose  
pseudo inverse



Zero-forcing equalizer:  
 $\mathbf{e}_{ZF} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}$

# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (3)



$$\mathbf{H}\mathbf{e}_{ZF} = \underbrace{\mathbf{H}(\mathbf{H}^H\mathbf{H})^{-1}}_{\neq \mathbf{I}} \mathbf{H}^H \mathbf{i}$$

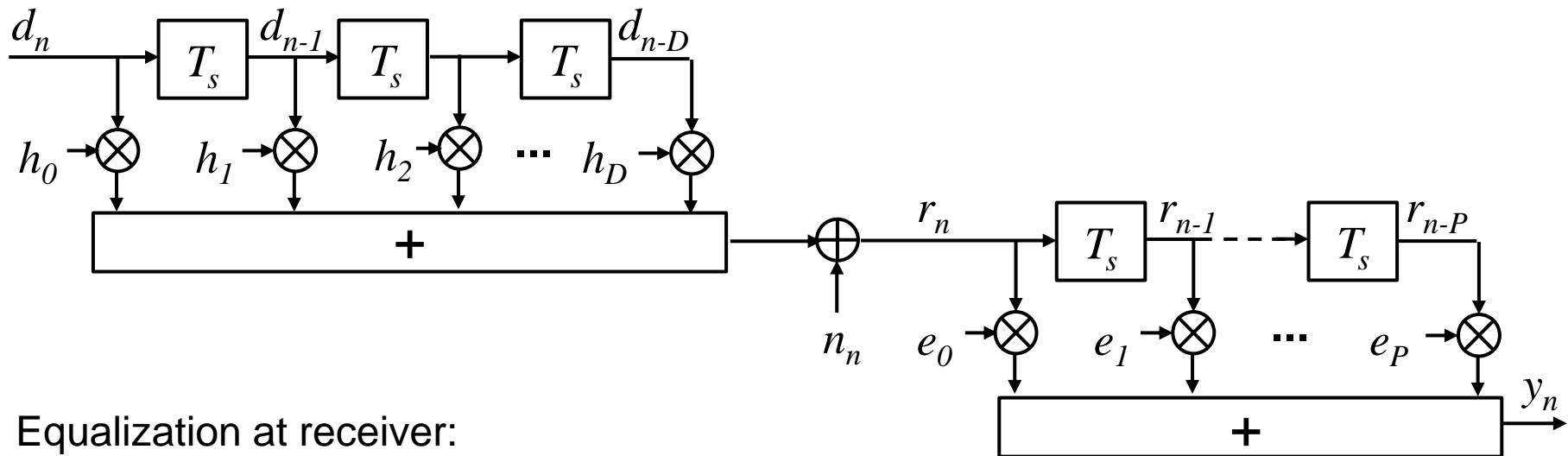
→ remaining  
equalization  
error  $\Delta$ .

noise enhancement  
→ major problem of  
ZF equalization!

Zero-forcing equalizer:  

$$\mathbf{e}_{ZF} = \boxed{(\mathbf{H}^H\mathbf{H})^{-1}} \mathbf{H}^H \mathbf{i}$$

# Non-Recursive Linear Zero-Forcing (ZF) Equalizer (4)



Equalization at receiver:

$$\begin{aligned}
 y_n &= \mathbf{e}_{ZF}^T \mathbf{r} \\
 &= \mathbf{e}_{ZF}^T \mathbf{H}^T \mathbf{d} + \mathbf{e}_{ZF}^T \mathbf{n} \\
 &= \underbrace{\mathbf{i}^T \mathbf{H}^* (\mathbf{H}^T \mathbf{H})^{-1}}_{\neq \mathbf{I}} \mathbf{H}^T \mathbf{d} + \underbrace{\mathbf{i}^T \mathbf{H}^* (\mathbf{H}^T \mathbf{H})^{-1}}_{\text{noise enhancement}} \mathbf{n} \\
 &\rightarrow \text{remaining equalization error } \Delta.
 \end{aligned}$$

Zero-forcing equalizer:

$$\mathbf{e}_{ZF} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{i}$$

# Differentiation of Complex Functions: Wirtinger Calculus

$$x = x_R + jx_I$$

$$\frac{\partial f(x)}{\partial x} = \frac{1}{2} \left[ \frac{\partial f(x)}{\partial x_R} - j \frac{\partial f(x)}{\partial x_I} \right]$$

The Wirtinger Calculus can be used for differentiation with respect to a complex variable  $x$ .

Partial derivatives with respect to real variables  $x_R$  and  $x_I$ :

$$\frac{\partial f}{\partial x_R} = \frac{\partial \operatorname{Re}\{f\}}{\partial x_R} + j \frac{\partial \operatorname{Im}\{f\}}{\partial x_R}$$

$$\frac{\partial f}{\partial x_I} = \frac{\partial \operatorname{Re}\{f\}}{\partial x_I} + j \frac{\partial \operatorname{Im}\{f\}}{\partial x_I}$$

Total derivative:

$$df = \frac{\partial f}{\partial x_R} dx_R + \frac{\partial f}{\partial x_I} dx_I$$

Target form:

$$df = \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial x^*} dx^*$$

$$\begin{aligned} df &= \frac{\partial f}{\partial x_R} \frac{1}{2} [dx + dx^*] + \frac{\partial f}{\partial x_I} \frac{j}{2} [dx^* - dx] \\ &= dx \underbrace{\frac{1}{2} \left( \frac{\partial f}{\partial x_R} - j \frac{\partial f}{\partial x_I} \right)}_{\frac{\partial f}{\partial x}} + dx^* \underbrace{\frac{1}{2} \left( \frac{\partial f}{\partial x_R} + j \frac{\partial f}{\partial x_I} \right)}_{\frac{\partial f}{\partial x^*}} \end{aligned}$$



Wilhelm Wirtinger (1865-1945),  
Austrian mathematician  
Source: Wikipedia

$$\begin{aligned} x_R &= \frac{1}{2} [x + x^*] & x_I &= \frac{1}{2j} [x - x^*] \\ & & &= \frac{j}{2} [x^* - x] \\ \Rightarrow \text{differentials:} \\ dx_R &= \frac{1}{2} [dx + dx^*] & dx_I &= \frac{j}{2} [dx^* - dx] \end{aligned}$$

# Wirtinger Calculus: Differentiation with Respect to Conjugate Complex Functions

$$x = x_R + jx_I$$

$$\frac{\partial f(x)}{\partial x} = \frac{1}{2} \left[ \frac{\partial f(x)}{\partial x_R} - j \frac{\partial f(x)}{\partial x_I} \right]$$

The Wirtinger Calculus can be used for differentiation with respect to a complex variable  $x$ .



$$\frac{\partial x^*}{\partial x} = 0$$



Wilhelm Wirtinger (1865-1945),  
Austrian mathematician  
Source: Wikipedia

Proof:

$$\frac{\partial x^*}{\partial x} = \frac{1}{2} \left[ \frac{\partial(x_R - jx_I)}{\partial x_R} - j \frac{\partial(x_R - jx_I)}{\partial x_I} \right] = \frac{1}{2} [1 - j(-j)] = 0$$

# Moore-Penrose Pseudoinverse

Eliakim Hastings Moore (1862-1932), American mathematician.  
Sir Roger Penrose (born 1931), English mathematical physicist

Generalization of the inverse matrix to non-square  $M \times N$  matrices.

The pseudoinverse  $\mathbf{A}^+$  of an  $M \times N$  matrix  $\mathbf{A}$  satisfies:

$$\mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A}$$

$$\mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+$$

$$(\mathbf{A}\mathbf{A}^+)^H = \mathbf{A}\mathbf{A}^+$$

$$(\mathbf{A}^+\mathbf{A})^H = \mathbf{A}^+\mathbf{A}$$

]  $\Rightarrow \mathbf{A}\mathbf{A}^+$  and  $\mathbf{A}^+\mathbf{A}$  are hermitian.

Special cases:

Columns of  $\mathbf{A}$  are linearly independent  
 $\Rightarrow \mathbf{A}^H\mathbf{A}$  is invertible.

$$\mathbf{A}^+ = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H$$

$$\mathbf{A}^+\mathbf{A} = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{A} = \mathbf{I}_N$$

Rows of  $\mathbf{A}$  are linearly independent  
 $\Rightarrow \mathbf{A}\mathbf{A}^H$  is invertible.

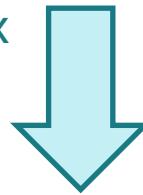
$$\mathbf{A}^+ = \mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1}$$

$$\mathbf{A}\mathbf{A}^+ = \mathbf{A}\mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1} = \mathbf{I}_N$$

# Hermitian Matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1M} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{NM} \end{bmatrix}$$

conjugate complex  
and transpose



$$\mathbf{A}^H = (\mathbf{A}^T)^* = \begin{bmatrix} a_{11}^* & a_{21}^* & \cdots & a_{N1}^* \\ a_{12}^* & a_{22}^* & \cdots & a_{N2}^* \\ a_{13}^* & a_{23}^* & \cdots & a_{N3}^* \\ \vdots & \vdots & \ddots & \vdots \\ a_{1M}^* & a_{2M}^* & \cdots & a_{NM}^* \end{bmatrix}$$

A square matrix  $\mathbf{A}$  is called a *Hermitian matrix*, if  $\mathbf{A}^H = \mathbf{A}$ .

# Non-Recursive Linear MMSE Equalizer (1)

An MMSE equalizer is often applied in noisy channels in order to reduce the detrimental effect of noise enhancement which a zero forcing equalizer would suffer from.

While a zero forcing equalizer aims at eliminating the intersymbol interference without taking additive noise into consideration, the MMSE equalizer minimizes the average energy of the equalization error including additive noise.

In general, an MMSE filter minimizes the mean energy of the filter output with respect to a reference signal. The linear MMSE equalization criterion can be formulated as

$$E \left\{ \left| y_n - d_{n-n_0} \right|^2 \right\} \rightarrow \min_e,$$

where  $y_n$  is the equalizer output at time  $n$ , and  $d_{n-n_0}$  is the desired filter output, i.e. the data symbol  $d_{n-n_0}$ . Similar as for the zero forcing equalizer, we allow for an equalizer processing delay  $n_0$ . The equalizer taps  $e$  are chosen such that the average squared error is minimized. The MMSE criterion has proven to be useful in many signal processing problems. Minimizing the squared error instead of e.g. the error itself has the physical interpretation of minimizing the energy of the error signal. Intuitively, it makes sense to give a higher weight to large errors than to small errors. This is also achieved by taking the square of the error in the MMSE criterion.

The solution of an MMSE problem is in general given by the Wiener-Hopf equation

$$\mathbf{e} = \mathbf{R}_{RR}^{-1} \mathbf{r}_{RD},$$

where  $\mathbf{R}_{RR}$  is the autocorrelation matrix of the filter input signal and  $\mathbf{r}_{RD}$  is the crosscorrelation vector of the filter input signal and the reference signal.

# Non-Recursive Linear MMSE Equalizer (2)

Hence, the MMSE solution requires knowledge of the statistical properties of the equalizer input signal. In case of transmission of uncorrelated data with zero mean, i.e.  $E\{d_n^* d_m\} = E\{d_n^*\} E\{d_m\}$  and  $E\{d_n\} = 0$ , through an ISI channel with additive white Gaussian noise, the MMSE solution for the equalizer taps is given by

$$\mathbf{e}_{\text{MMSE}} = \left( \mathbf{H}^H \mathbf{H} + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1} \mathbf{H}^H \mathbf{i},$$

where  $P_D$  is the average received power of the data signal and  $P_N$  is the noise power. For high SNR, i.e.

$$\frac{P_N}{P_D} \rightarrow 0,$$

the MMSE solution converges to the zero forcing solution

$$\mathbf{e}_{\text{ZF}} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}.$$

In case of low and medium SNR, the channel is not exactly inverted by the MMSE equalizer but the noise enhancement, which results from the matrix inversion  $(\mathbf{H}^H \mathbf{H})^{-1}$  is limited.

# Non-Recursive Linear MMSE Equalizer (3)

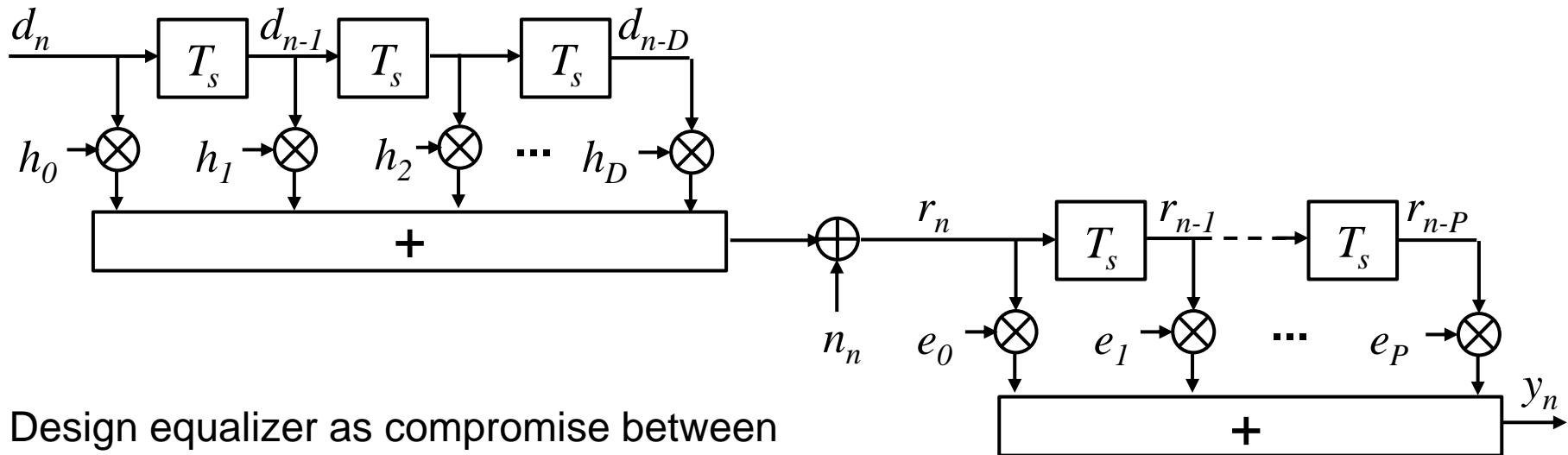
Note that the MMSE solution requires knowledge of the channel matrix  $\mathbf{H}$  and of the SNR at the receiver, while the zero forcing solution requires knowledge of the channel matrix  $\mathbf{H}$  only. Channel knowledge at the receiver is called *channel state information at the receiver (CSIR)*. The channel matrix  $\mathbf{H}$  can be estimated at the receiver based on known training symbols which are periodically transmitted.

Accurate estimation of the SNR is sometimes more difficult as only an observation window of limited length can be used for estimation. A reduced complexity solution is to simply compute the MMSE equalizer taps under the assumption of an expected SNR without explicit SNR estimation. If the true SNR differs significantly from the assumed SNR, the MMSE equalizer performance degrades and can become worse than the performance of a zero forcing equalizer.

In time-varying channels such as mobile communications channels, the channel matrix  $\mathbf{H}$  changes over time. Hence, the equalizer taps have to be adapted to the current channel state, i.e. the equalizer needs to be implemented as an *adaptive filter*. Equalizer adaptation requires frequent execution of computationally expensive operations such as a matrix inversion.

However, computationally efficient recursive algorithms exist for approximation of the MMSE solution without explicit matrix inversion. The most famous examples for such adaptive algorithms are the *Least Mean Squares (LMS) algorithm* and the *Recursive Least Squares (RLS) algorithm*.

# Non-Recursive Linear Minimum Mean Square Error (MMSE) Equalizer



Design equalizer as compromise between perfect equalization and limited noise enhancement such that

mean square error (MSE)

$$\overbrace{E \left\{ \left| y_n - d_{n-n_0} \right|^2 \right\}}^{\lambda} \rightarrow \min_e$$

equalizer output

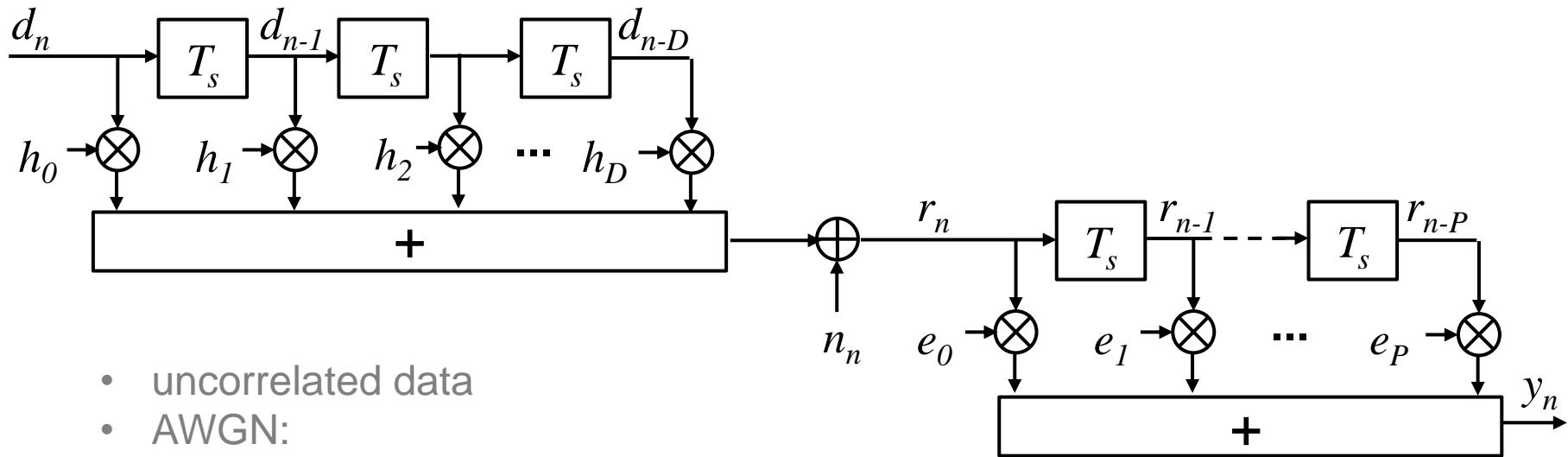
desired equalizer output (perfect equalization)

MMSE equalizer for

- uncorrelated data with zero mean and
- AWGN:

$$\mathbf{e}_{\text{MMSE}} = \left( \mathbf{H}^H \mathbf{H} + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1} \mathbf{H}^H \mathbf{i}$$

# Non-Recursive Linear Equalizer: MMSE and Zero Forcing (ZF)



- uncorrelated data
- AWGN:

MMSE:

$$\mathbf{e}_{\text{MMSE}} = \left( \mathbf{H}^H \mathbf{H} + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1} \mathbf{H}^H \mathbf{i}$$

$\xrightarrow{\text{SNR} \rightarrow \infty}$

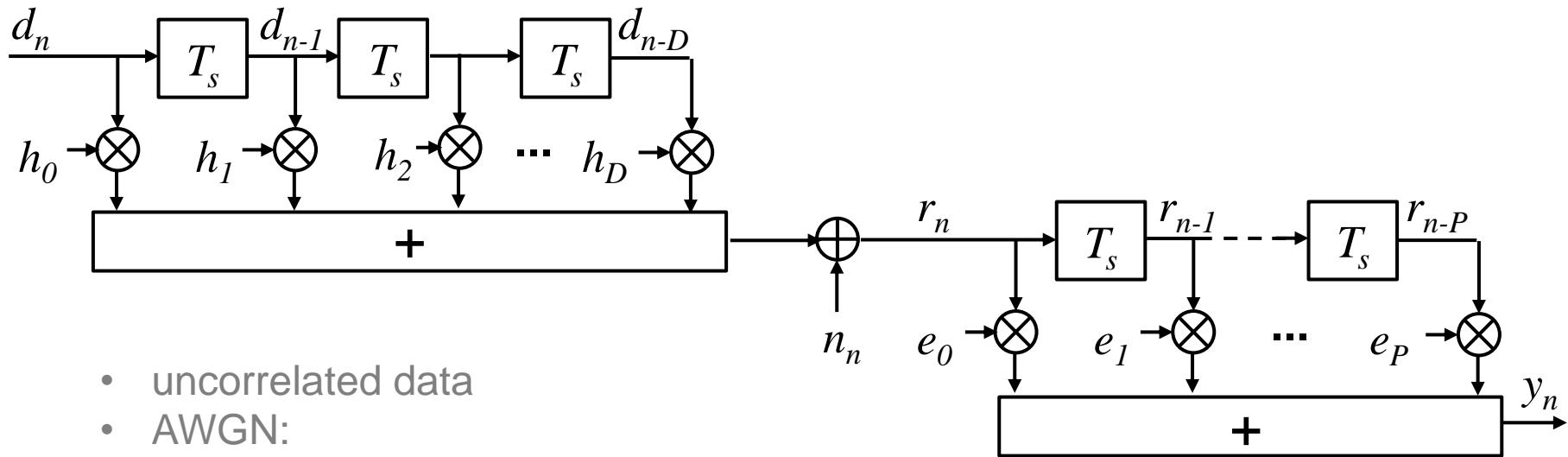
$$\frac{P_N}{P_D} \rightarrow 0$$

Zero Forcing (ZF):

$$\mathbf{e}_{\text{ZF}} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}$$

For large SNR, the MMSE solution converges to the zero forcing (ZF) solution.

# Required Channel Knowledge for MMSE and Zero Forcing (ZF) Equalizer



- uncorrelated data
- AWGN:

MMSE:

$$\mathbf{e}_{\text{MMSE}} = \left( \mathbf{H}^H \mathbf{H} + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1} \mathbf{H}^H \mathbf{i}$$

Zero Forcing (ZF):

$$\mathbf{e}_{\text{ZF}} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{i}$$

Required channel estimation at receiver:

- $\mathbf{H}$
- SNR

- $\mathbf{H}$

# Non-Recursive Linear MMSE Equalizer – Derivation (1)

$$E\left\{\left|y_n - d_{n-n_0}\right|^2\right\} \rightarrow \min_{\mathbf{e}}$$

$$\begin{aligned} E\left\{\left|y_n - d_{n-n_0}\right|^2\right\} &= E\left\{\left(y_n^* - d_{n-n_0}^*\right)\left(y_n - d_{n-n_0}\right)\right\} \\ &= E\left\{\left(\mathbf{e}^H \mathbf{r}^* - d_{n-n_0}^*\right)\left(\mathbf{r}^T \mathbf{e} - d_{n-n_0}\right)\right\} \\ &\quad \uparrow \\ y_n &= \mathbf{r}^T \mathbf{e} \end{aligned}$$

$$= E\left\{\mathbf{e}^H \mathbf{r}^* \mathbf{r}^T \mathbf{e} - \mathbf{e}^H \mathbf{r}^* d_{n-n_0} - d_{n-n_0}^* \mathbf{r}^T \mathbf{e} + |d_{n-n_0}|^2\right\}$$

$\mathbf{e}$  is deterministic

$$\downarrow = \underbrace{\mathbf{e}^H E\{\mathbf{r}^* \mathbf{r}^T\} \mathbf{e}}_{\mathbf{R}_{RR}} - \underbrace{\mathbf{e}^H E\{\mathbf{r}^* d_{n-n_0}\}}_{\mathbf{r}_{RD}} - \underbrace{E\{d_{n-n_0}^* \mathbf{r}^T\} \mathbf{e}}_{\mathbf{r}_{RD}^H} + \underbrace{E\{|d_{n-n_0}|^2\}}_{P_D}$$

autocorrelation  
matrix of equalizer  
input signal

crosscorrelation vector  
of equalizer input signal  
and data at time  $n-n_0$

average power  
of data signal

# Non-Recursive Linear MMSE Equalizer – Derivation (2)

$$\text{MSE} = E \left\{ \left| y_n - d_{n-n_0} \right|^2 \right\} = \mathbf{e}^H \mathbf{R}_{RR} \mathbf{e} - \mathbf{e}^H \mathbf{r}_{RD} - \mathbf{r}_{RD}^H \mathbf{e} + P_D$$

$$\frac{d\text{MSE}}{d\mathbf{e}^H} = \mathbf{R}_{RR} \mathbf{e} - \mathbf{r}_{RD} = 0$$

↑  
Wirtinger  
calculus

$$\mathbf{e} = \mathbf{R}_{RR}^{-1} \mathbf{r}_{RD}$$

Wiener-Hopf equation

# Non-Recursive Linear MMSE Equalizer – Derivation (3)

**For AWGN channel:**  $\mathbf{r} = \mathbf{H}^T \mathbf{d} + \mathbf{n}$

## Autocorrelation matrix of equalizer input signal:

$$\begin{aligned}\mathbf{R}_{RR} &= E\left\{\mathbf{r}^* \mathbf{r}^T\right\}=E\left\{\left(\mathbf{H}^H \mathbf{d}^*+\mathbf{n}^*\right)\left(\mathbf{d}^T \mathbf{H}+\mathbf{n}^T\right)\right\} \\&= E\left\{\mathbf{H}^H \mathbf{d}^* \mathbf{d}^T \mathbf{H}+\mathbf{H}^H \mathbf{d}^* \mathbf{n}^T+\mathbf{n}^* \mathbf{d}^T \mathbf{H}+\mathbf{n}^* \mathbf{n}^T\right\}\end{aligned}$$

**H** is known at receiver due to channel estimation

$$\downarrow = \mathbf{H}^H \underbrace{E\{\mathbf{d}^* \mathbf{d}^T\}}_{\mathbf{R}_{DD}} \mathbf{H}^H + \mathbf{H}^H \underbrace{E\{\mathbf{d}^* \mathbf{n}^T\}}_{E\{\mathbf{d}^*\} E\{\mathbf{n}^T\}} + \underbrace{E\{\mathbf{n}^* \mathbf{d}^T\}}_{E\{\mathbf{n}^*\} E\{\mathbf{d}^T\}} \mathbf{H} + \underbrace{E\{\mathbf{n}^* \mathbf{n}^T\}}_{\mathbf{R}_{NN}}$$

(data and noise are uncorrelated)

$$= \mathbf{H}^H \underbrace{\mathbf{R}_{DD}}_{P_D \mathbf{I}_{P+D+1}} \mathbf{H}^H + \mathbf{H}^H E\{\mathbf{d}^*\} E\{\mathbf{n}^T\} \xrightarrow{\mathbf{0}} + E\{\mathbf{n}^*\} E\{\mathbf{d}^T\} \mathbf{H} + \underbrace{\mathbf{R}_{NN}}_{P_N \mathbf{I}_{P+N}}$$

for uncorrelated data  
with zero mean

## for AWGN

## for AWGN

# Non-Recursive Linear MMSE Equalizer – Derivation (4)

Crosscorrelation vector of equalizer input signal and data at time  $n-n_0$ :

$$\mathbf{r}_{RD} = E\left\{\mathbf{r}^* d_{n-n_0}\right\} = E\left\{d_{n-n_0} \left(\mathbf{H}^H \mathbf{d}^* + \mathbf{n}^*\right)\right\}$$

$\uparrow$   
 $\mathbf{r} = \mathbf{H}^T \mathbf{d} + \mathbf{n}$       for AWGN

**H** is known at receiver due to channel estimation

$$\downarrow = \mathbf{H}^H E\left\{d_{n-n_0} \mathbf{d}^*\right\} + \underbrace{E\left\{d_{n-n_0} \mathbf{n}^*\right\}}_{E\left\{d_{n-n_0}\right\} E\left\{\mathbf{n}^*\right\}} \quad (\text{data and noise are uncorrelated})$$

**0 for AWGN**

$$= \mathbf{H}^H E \left\{ d_{n-n_0} \begin{bmatrix} d_n^* \\ \vdots \\ d_{n-n_0}^* \\ \vdots \\ d_{n-D-P}^* \end{bmatrix} \right\} = \mathbf{H}^H \begin{bmatrix} E\{d_{n-n_0}\} E\{d_n^*\} \\ \vdots \\ E\{|d_{n-n_0}|^2\} \\ \vdots \\ E\{d_{n-n_0}\} E\{d_{n-D-P}^*\} \end{bmatrix} = \mathbf{H}^H \begin{bmatrix} \mathbf{0} \\ \vdots \\ P_D \\ \vdots \\ \mathbf{0} \end{bmatrix} = P_D \mathbf{H}^H \begin{bmatrix} \mathbf{0} \\ \vdots \\ 1 \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

↑ uncorrelated data      ↑ data with zero mean

# Non-Recursive Linear MMSE Equalizer – Derivation (5)

Autocorrelation matrix of equalizer input signal:

$$\mathbf{R}_{RR} = P_D \mathbf{H}^H \mathbf{H}^H + P_N \mathbf{I}_{P+1} = P_D \left( \mathbf{H}^H \mathbf{H}^H + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right) \quad \leftarrow$$

Assumptions:

- AWGN
- uncorrelated data with zero mean

$$\Rightarrow \mathbf{R}_{RR}^{-1} = \frac{1}{P_D} \left( \mathbf{H}^H \mathbf{H}^H + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1}$$

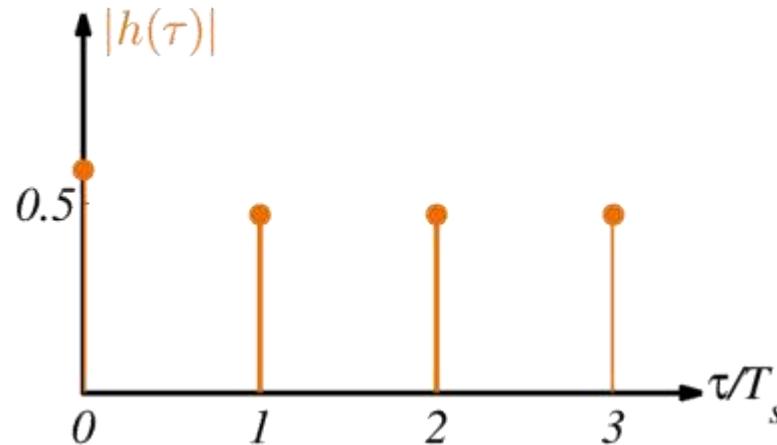
$$\mathbf{r}_{RD} = P_D \mathbf{H}^H \mathbf{i}$$

Equalizer taps:

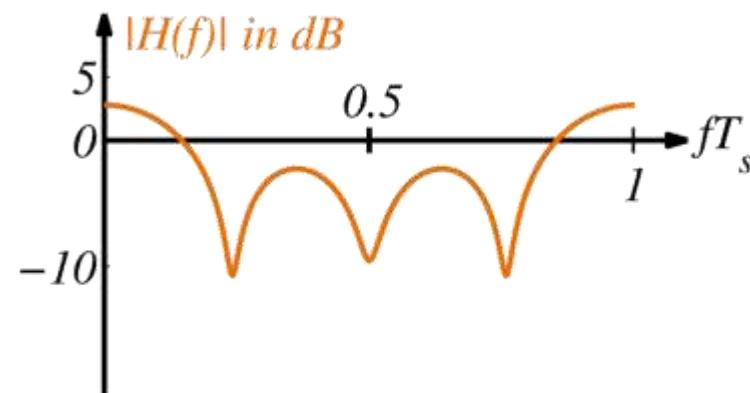
$$\mathbf{e}_{\text{MMSE}} = \mathbf{R}_{RR}^{-1} \mathbf{r}_{RD} = \left( \mathbf{H}^H \mathbf{H} + \frac{P_N}{P_D} \mathbf{I}_{P+1} \right)^{-1} \mathbf{H}^H \mathbf{i}$$

# Zero-Forcing vs. MMSE Equalizer (1)

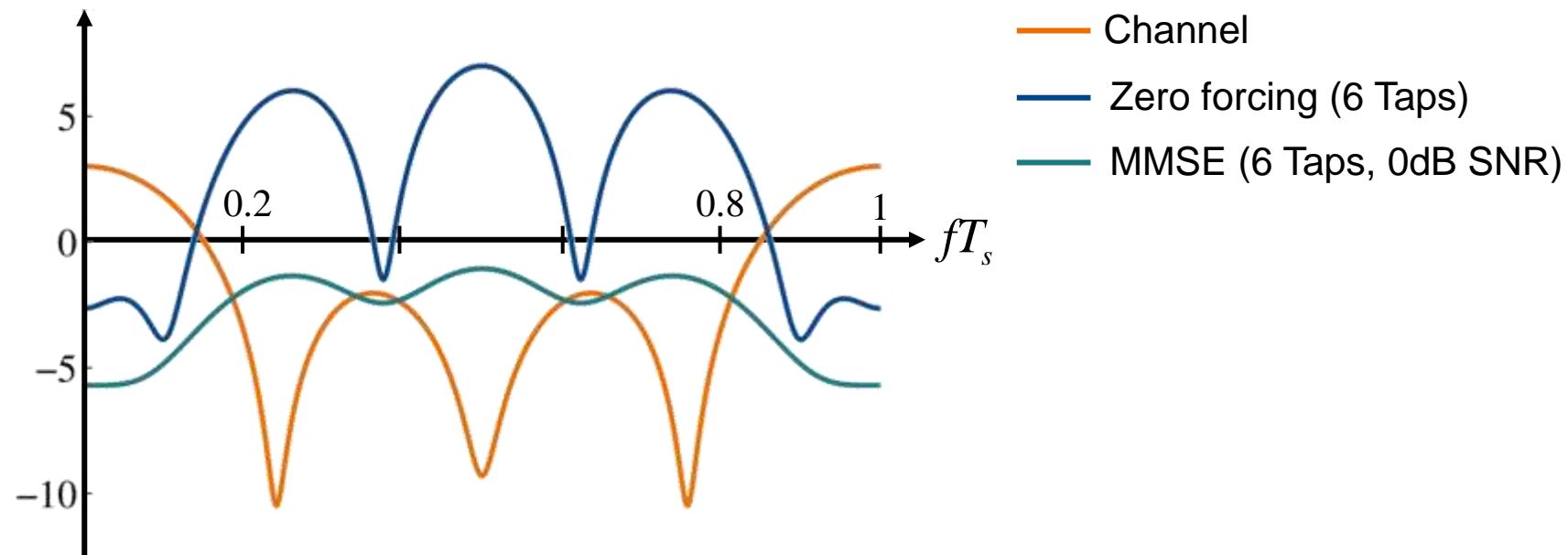
Channel impulse response



Channel transfer function

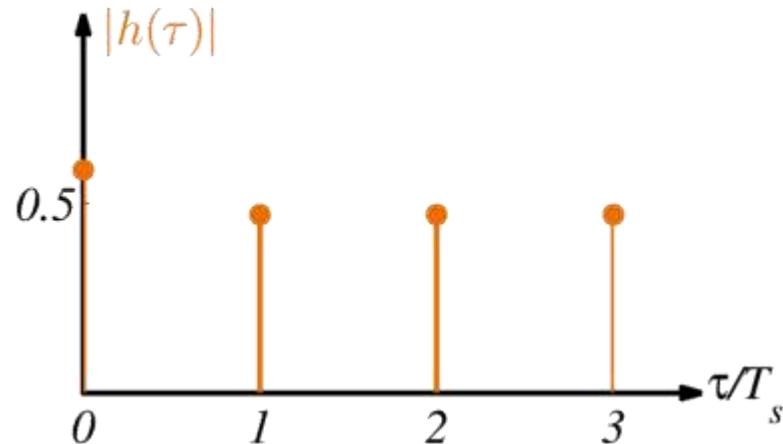


Equalizer transfer function

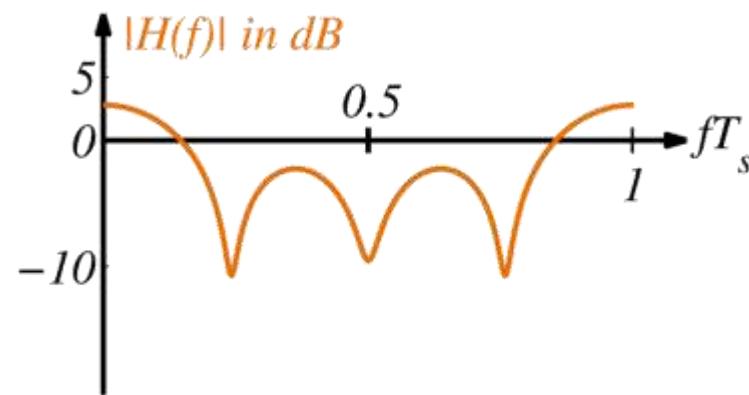


# Zero-Forcing vs. MMSE Equalizer (2)

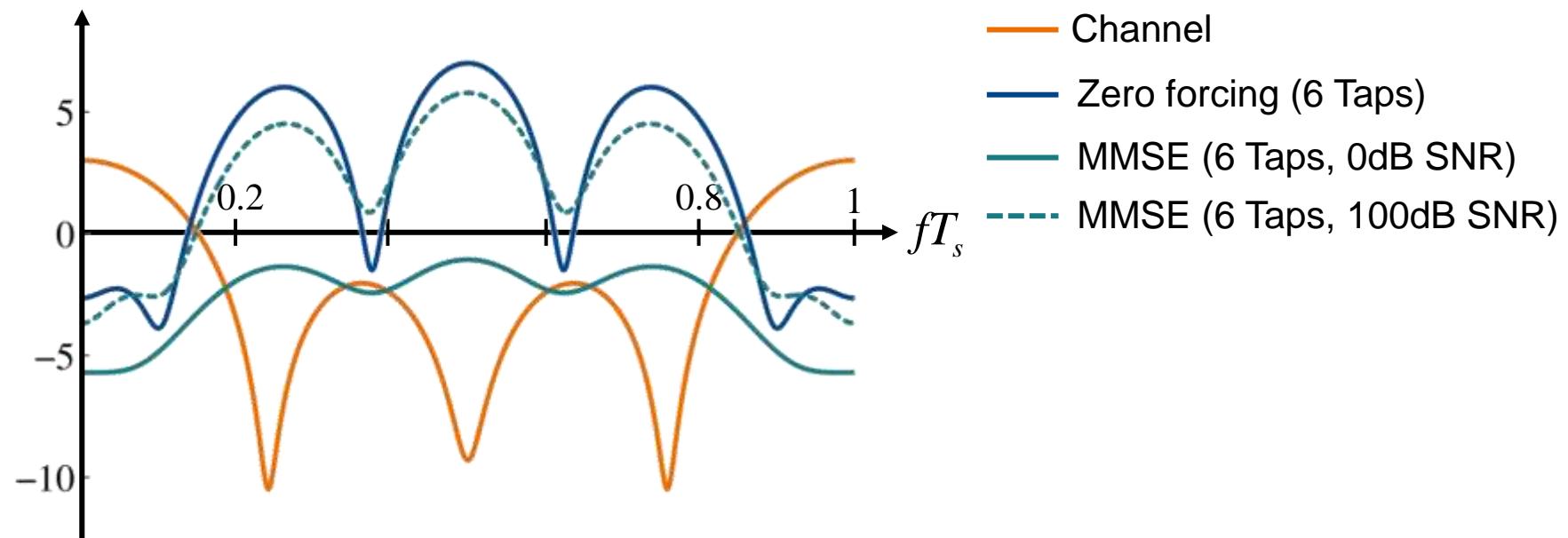
Channel impulse response



Channel transfer function

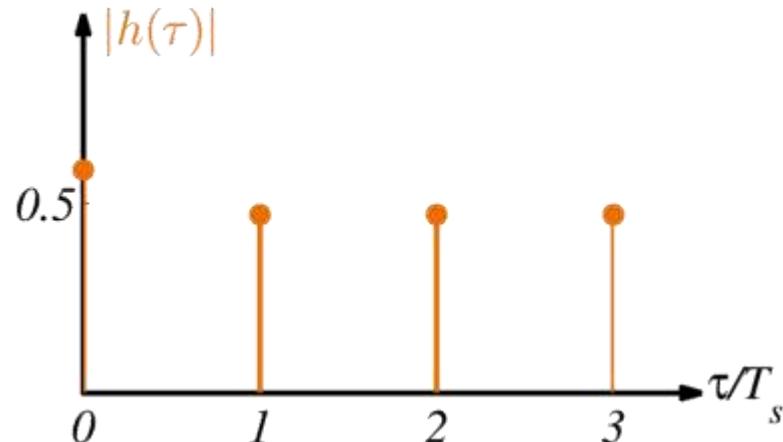


Equalizer transfer function

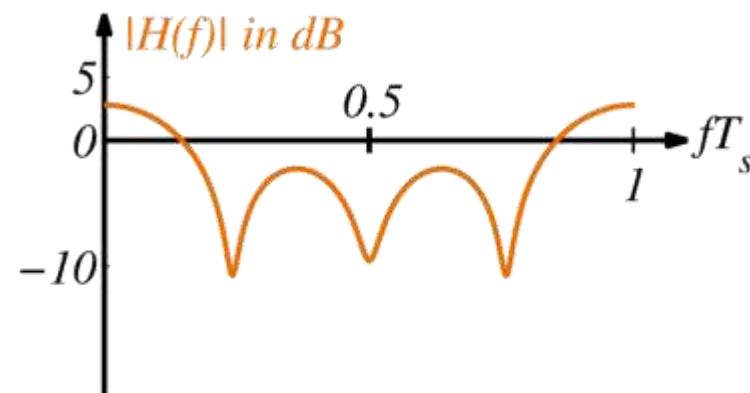


# Zero-Forcing vs. MMSE Equalizer (3)

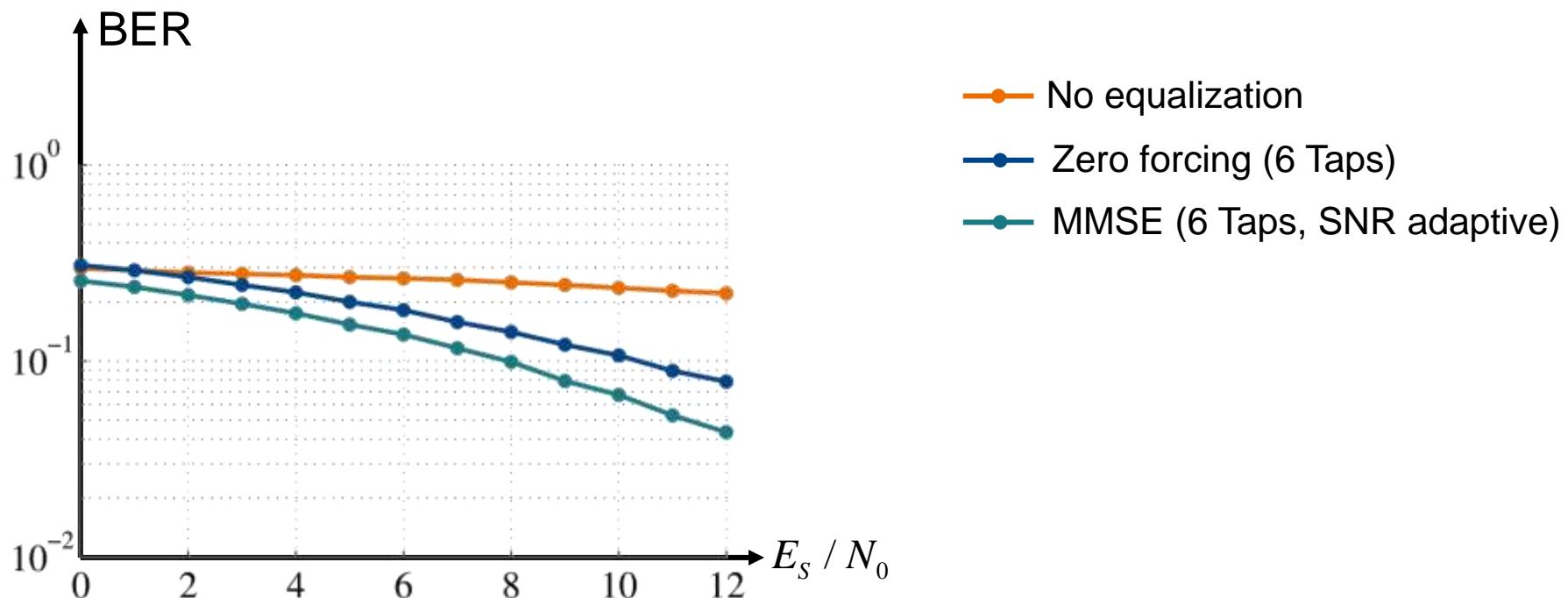
Channel impulse response



Channel transfer function

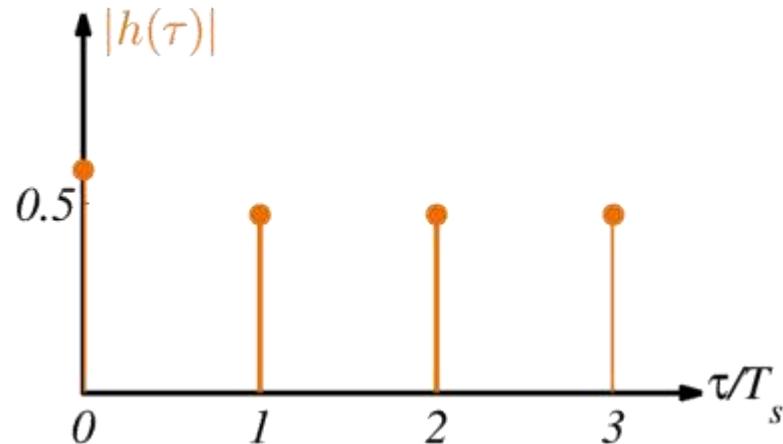


BER

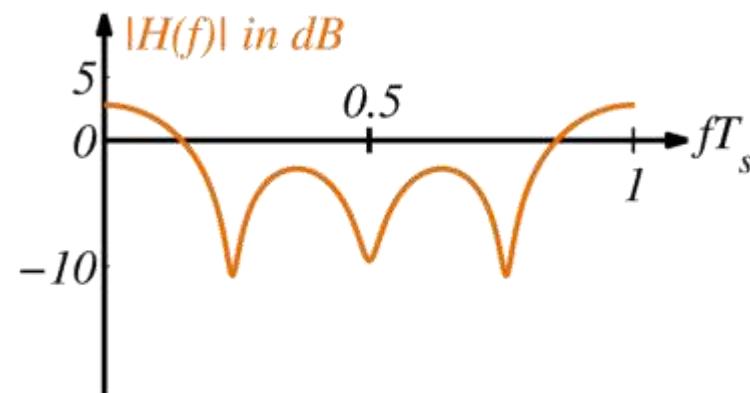


# Zero-Forcing vs. MMSE Equalizer (4)

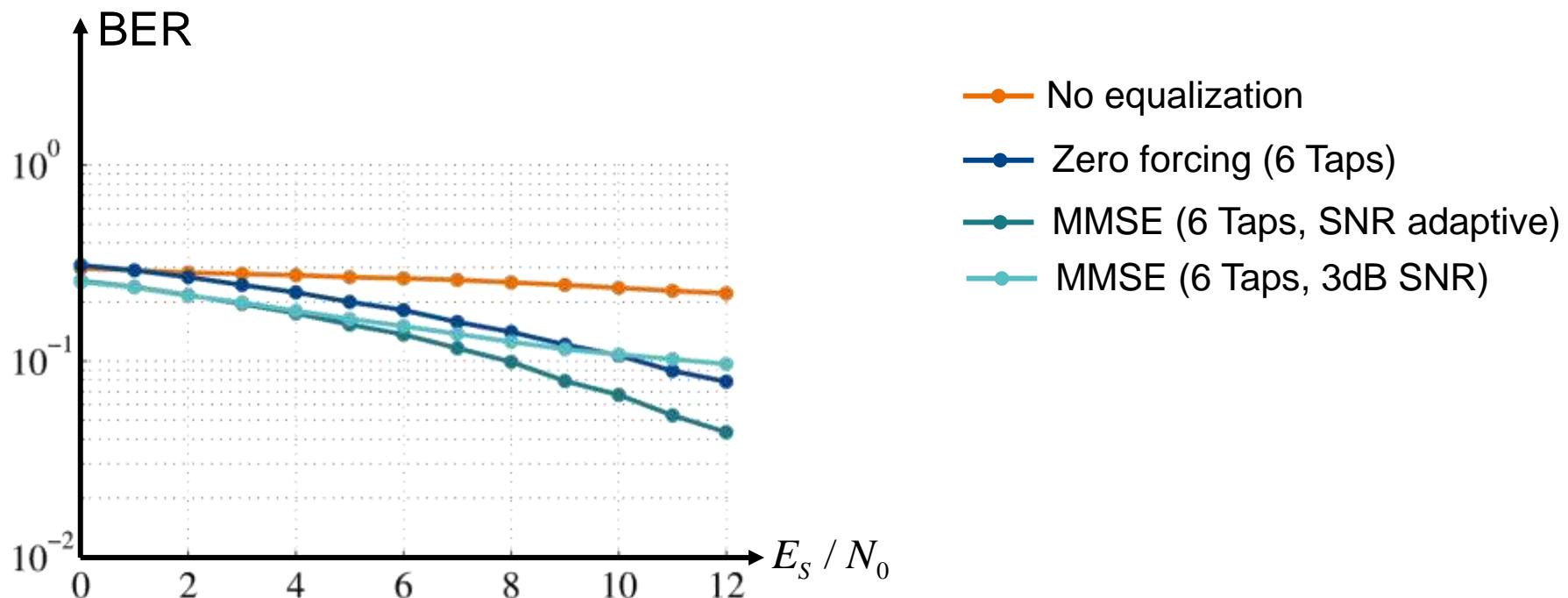
Channel impulse response



Channel transfer function

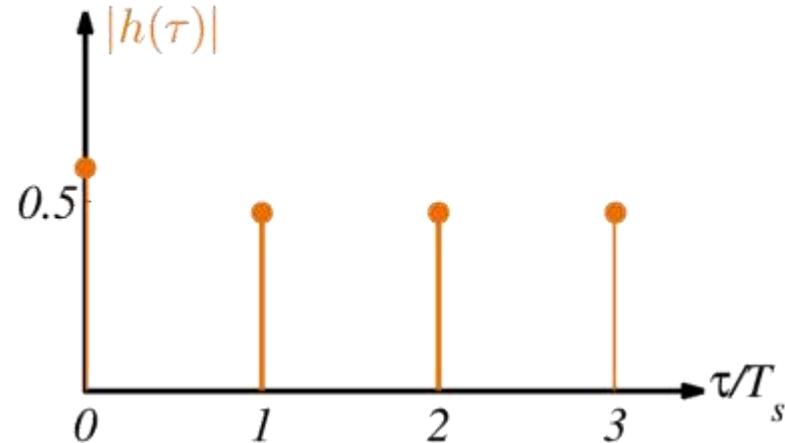


BER

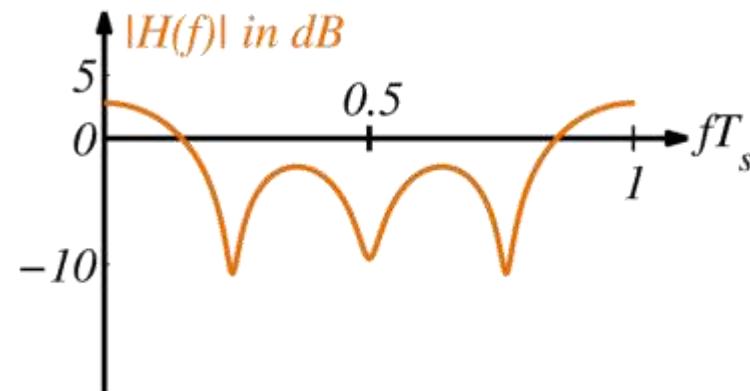


# Zero-Forcing vs. MMSE Equalizer (5)

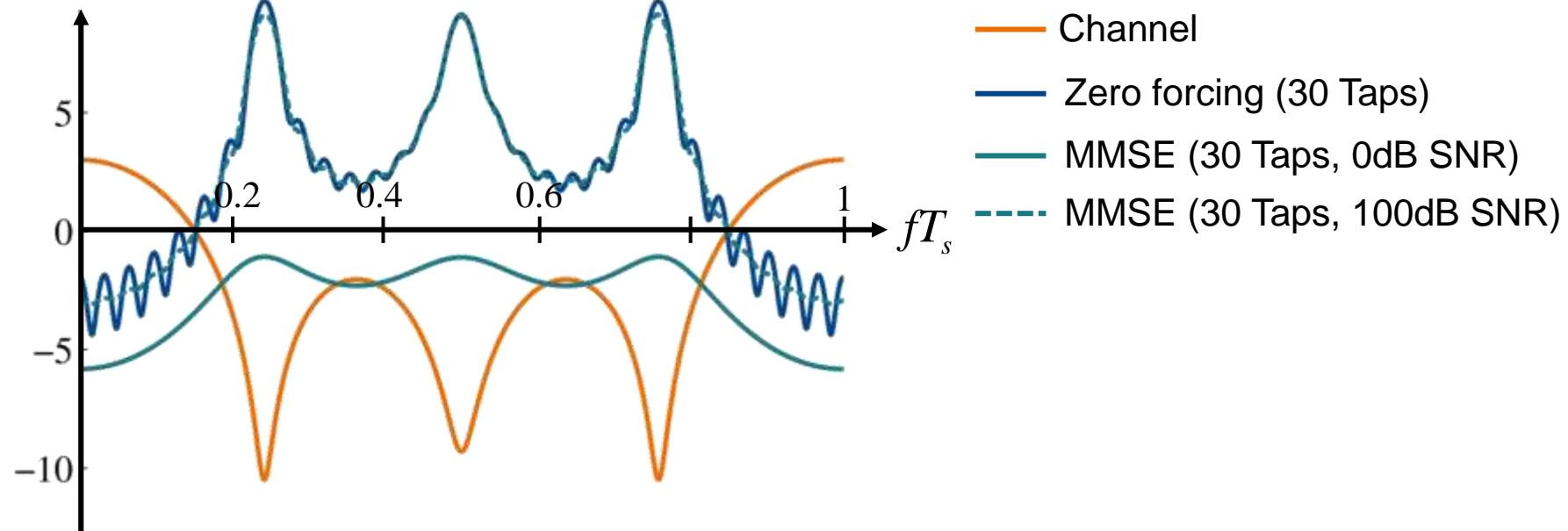
Channel impulse response



Channel transfer function

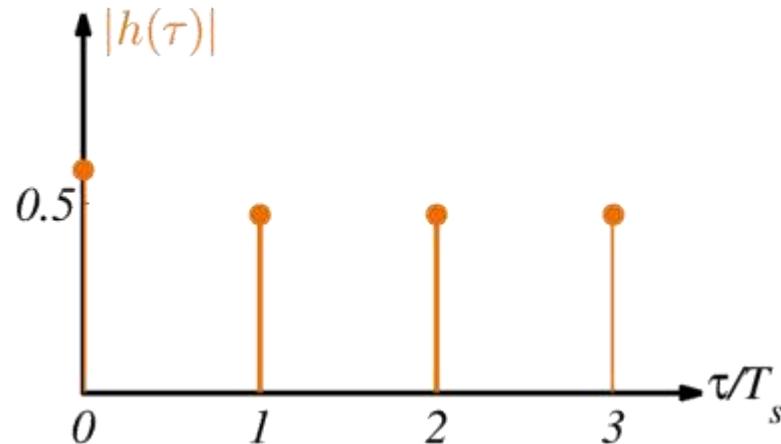


Equalizer transfer function

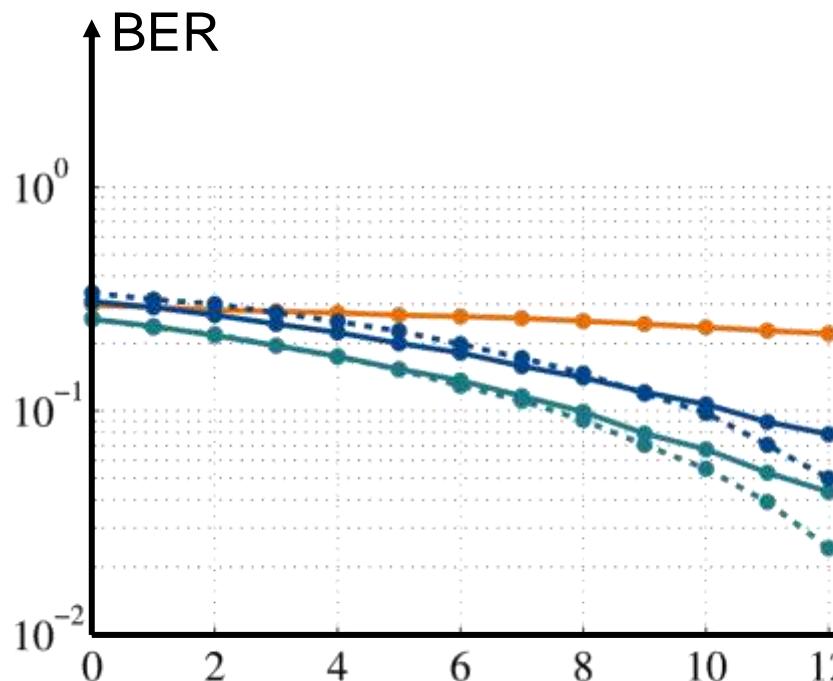
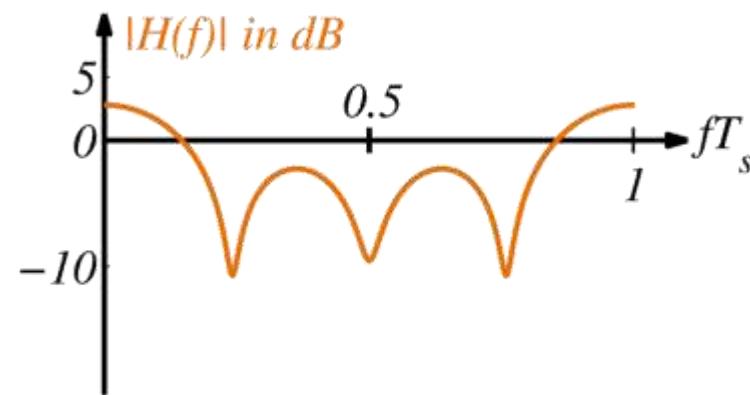


# Zero-Forcing vs. MMSE Equalizer (6)

Channel impulse response



Channel transfer function



- No equalization
- Zero forcing (6 Taps)
- MMSE (6 Taps, adaptive SNR)
- Zero forcing (30 Taps)
- MMSE (30 Taps, adaptive SNR)

# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (1)

Noise enhancement is the major problem of linear equalizers. Even though the effect is reduced when using a linear MMSE equalizer rather than a zero-forcing (ZF) equalizer, it is still there. A way to combat the detrimental effect of noise enhancement is the application of a non-linear equalizer. A decision feedback equalizer (DFE) is a straightforward non-linear solution. A DFE takes a hard decision on the transmit symbol  $d_n$  when the sample

$$r_n = h_0 d_n + \sum_{\nu=1}^D h_\nu d_{n-\nu} + n_n$$

is received. As estimates on the interfering symbols  $d_{n-1}, d_{n-2}, \dots, d_{n-D}$  have already been decided at time  $n$ , their effect can be cancelled from the received sample  $r_n$  before making the decision on  $d_n$ . This would yield a decision unit input sample

$$\hat{r}_n = h_0 d_n + \sum_{\nu=1}^D h_\nu d_{n-\nu} + n_n - \sum_{\nu=1}^D h_\nu \hat{d}_{n-\nu}$$

which in case of no decision errors, i.e.  $\hat{d}_{n-\nu} = d_{n-\nu}$ , would result in

$$\hat{r}_n = h_0 d_n + n_n.$$

In the equation above, all interference has been cancelled without noise enhancement. However, the message symbol  $d_n$  is weighted by the first channel tap  $h_0$ .

# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (2)

Therefore, we normalize the received sample such that the normalized sample

$$\tilde{r}_n = \frac{1}{h_0} r_n - \sum_{v=1}^D \frac{h_v}{h_0} \hat{d}_{n-v}$$

enters the DFE, which in case of no decision errors yields

$$\tilde{r}_n = d_n + \frac{1}{h_0} n_n.$$

The decision on  $d_n$  is made at time  $n$ , i.e. when  $d_n$  was received via the first tap gain  $h_0$  but before the multipath components  $h_0, h_1, \dots, h_D$ , which also carry information on  $d_n$  have been received. In fact, the energy contained in the multipath components  $h_0, h_1, \dots, h_D$  is lost due to the interference cancellation. This results also in a kind of SNR loss as can be seen from the multiplication of the additive noise by  $1/h_0$ . (As a channel cannot add additional energy, we will have  $|h_0| \leq 1$ ). The SNR loss is the more pronounced, the smaller  $|h_0|$ . In order to limit this SNR loss, a DFE is typically combined with a linear FIR equalization part which removes weak precursors in the channel impulse response before the DFE equalization part. Hence, the DFE will face an effective channel with sufficiently strong first tap gain  $|h_0|$ .

The major problem of a DFE is the danger of erroneous decisions, i.e.  $\hat{d}_{n-v} \neq d_{n-v}$ , which will cause error propagation in the interference cancellation. As a rule of thumb, subtracting something wrong is typically worse than subtracting no interference at all. Again, the probability of error propagation is the higher, the smaller  $|h_0|$ .

# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (3)

In the DFE solution described above, the feedback filter order is equal to the channel memory  $D$  or in case of an additional feedforward (FIR) filter, the feedback filter order is given by the memory of the effective channel seen at the output of the FIR equalization part.

Instead of perfectly cancelling the remaining interference by the DFE, an MMSE approach can be chosen, where the feedforward filter  $\mathbf{f}$  and the feedback filter  $\mathbf{b}$  are designed jointly.

This approach may also be chosen if for complexity reasons the feedback filter order must be chosen smaller than the effective channel memory.

# Tomlinson-Harashima Precoding (1)

Error propagation due to decision errors is the major problem of a DFE. If the channel impulse response  $\mathbf{h}$  is known to the transmitter, the decision feedback equalization can be moved to the transmitter. I.e., the transmit signal is pre-equalized before transmission such that intersymbol-interference-free samples are observed at the receiver.

Accurate channel knowledge is difficult to obtain in practice. In case of duplex communication via a reciprocal channel, the channel can be estimated from the incoming signal. E.g. in a wireless time division duplex (TDD) system, uplink and downlink are operated in the same frequency band. Consequently, the channel is reciprocal and estimation of the channel impulse response from the uplink channel can be used for pre-equalization of the downlink signal. Note, that in practice, the RF components used in uplink and downlink transmission are not reciprocal making accurate channel knowledge more difficult to obtain at the transmitter.

Another option to obtain channel knowledge at the transmitter is via feedback from the receiver. The transmitter sends pilot symbols to the receiver. The receiver estimates the channel impulse response based on the received pilot symbols and feeds the channel estimation result back to the transmitter via a signalling channel. Since the available data rate on the feedback channel is typically very limited, only a quantized version of the channel impulse response can be signalled to the transmitter. In slowly varying channels, the assumption of channel knowledge at the transmitter is less critical. We now abstract from those practical problems and assume that perfect knowledge of the channel impulse response is available at the transmitter.

As the data  $\mathbf{d}$  to be transmitted is known at the transmitter, there is no problem of error propagation when the intersymbol interference is cancelled at the transmitter.

# Tomlinson-Harashima Precoding (2)

Instead of transmitting the data symbol  $d_n$  at time  $n$ , the pre-equalized symbol

$$\hat{d}_n = \frac{1}{h_0} d_n - \sum_{v=1}^D \frac{h_v}{h_0} \hat{d}_{n-v} \quad (1)$$

could be transmitted. However, subtracting the intersymbol interference may result in a transmit symbol with large magnitude  $|\hat{d}_n|$ , which would consume a lot of transmit power. Most of the transmit power would be used for cancellation of the intersymbol interference rather than for the desired symbol itself. Therefore, a modulo operation is applied which limits the magnitude of the transmit symbols  $\tilde{d}_n$  to the range

$$|\tilde{d}_n| < \tilde{d}_{\max}.$$

The modulo operation is equivalent to adding multiples of  $2\tilde{d}_{\max}$  to the symbols  $\hat{d}_n$ . Hence, the transmit symbols are given by

$$\tilde{d}_n = d_n - \sum_{v=1}^D \frac{h_v}{h_0} \hat{d}_{n-v} + k \cdot 2\tilde{d}_{\max}, \quad k \in \mathbb{Z}_0, \quad (2)$$

where  $k$  is chosen such that  $|\tilde{d}_n| < \tilde{d}_{\max}$ . Note, that in contrast to equation (1),  $d_n$  is not scaled by  $h_0$  in equation (2). This is in order not to destroy the weighting relative to the modulo term  $k \cdot 2\tilde{d}_{\max}$ .

The modulo operation also avoids possible instabilities caused by the feedback filter. This method of pre-equalization at the transmitter and a modulo operation for transmit power limitation is called *Tomlinson-Harashima precoding*. In case of complex symbols  $d_n$ , the modulo operation has to be performed in both quadrature components, i.e. in both real and imaginary part.

# Tomlinson-Harashima Precoding (3)

At the output of the ISI channel, the receiver observes

$$\begin{aligned}
 r_n &= h_0 \tilde{d}_n + \sum_{\nu=1}^D h_\nu \tilde{d}_{n-\nu} + n_n \\
 &= h_0 \left( d_n - \sum_{\nu=1}^D \frac{h_\nu}{h_0} \tilde{d}_{n-\nu} + 2k\tilde{d}_{\max} \right) + \sum_{\nu=1}^D h_\nu \tilde{d}_{n-\nu} + n_n \\
 &= h_0 \left( d_n + 2k\tilde{d}_{\max} \right) + n_n,
 \end{aligned}$$

which after normalization by  $h_0$  yields

$$\tilde{r}_n = d_n + 2k\tilde{d}_{\max} + \frac{n_n}{h_0}.$$

Hence, the received samples are free of intersymbol interference and no equalization is required at the receiver. The only operations which the receiver has to perform are the multiplication by  $1/h_0$  and reversing the modulo operation by subtracting  $2k\tilde{d}_{\max}$ .

As the noise is enhanced by the factor  $1/h_0$ , a channel with strong first tap gain  $h_0$  is desirable. In case  $h_0$  is weak, Tomlinson-Harashima precoding can be complemented by an FIR filter which eliminates weak precursors of the channel impulse response in a similar way as it is done for DFE at the receiver. However, since operation is done at the transmitter, the FIR filter has to be applied following Tomlinson-Harashima precoding. In this way, the FIR filter and the directly connected channel form a superchannel with an effective impulse response. Tomlinson-Harashima precoding takes care of the remaining intersymbol-interference of this effective channel.

# Tomlinson-Harashima Precoding (3)

The modulo operation at the transmitter has limited the transmit power. However, the transmit symbols are not anymore elements of a well defined signal space constellation but can have any value within the range defined by the modulo operation, i.e.  $-\tilde{d}_{\max} < \tilde{d}_n < \tilde{d}_{\max}$  (for real valued data symbols). In fact, the transmit symbols are approximately uniformly distributed within that range. Such a signal has a high dynamic range which is often measured by the peak-to-average power ratio (PAPR) of the transmit signal.

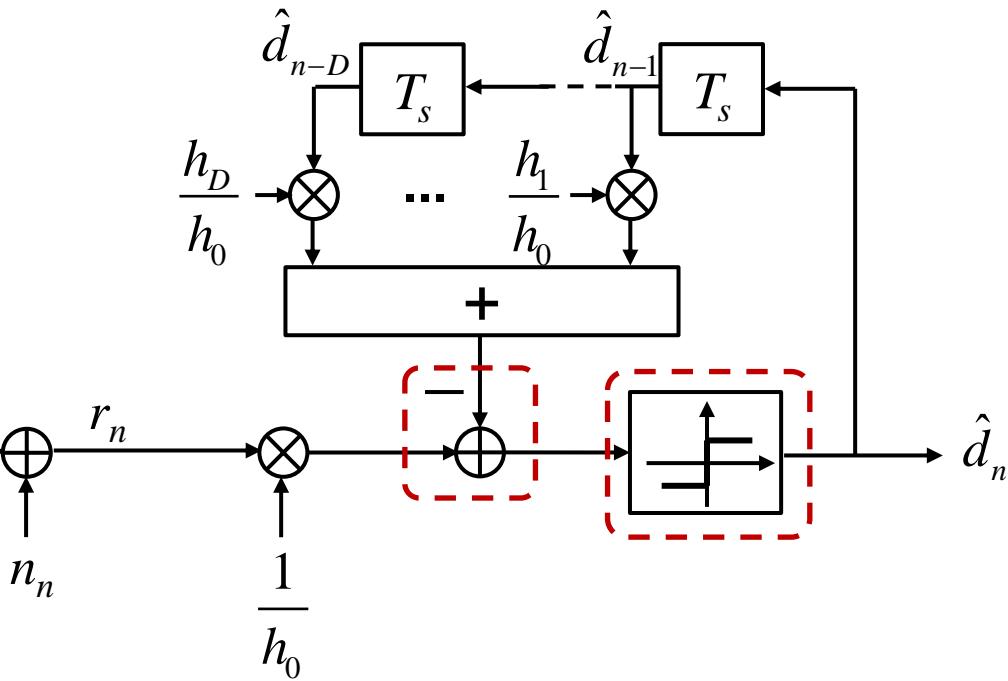
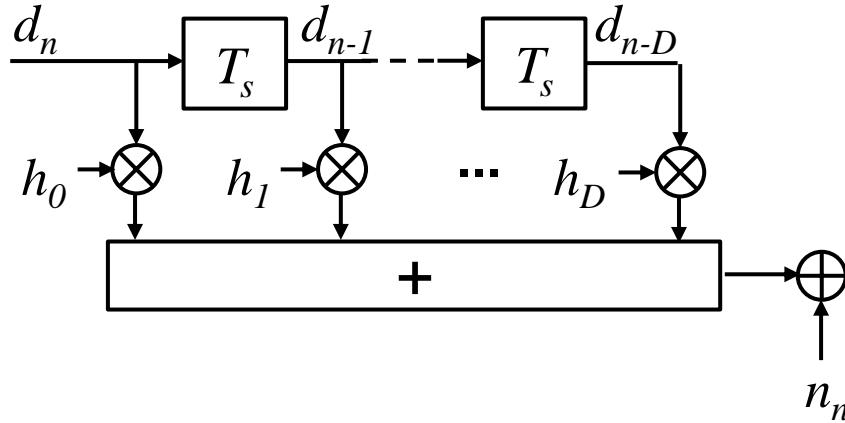
A high PAPR is undesirable as the high power amplifiers need to be linear in a wide dynamic range. This is difficult to achieve in practice and often requires to operate the power amplifier in an inefficient operation point, i.e. with large input power backoff.

Another disadvantage of Tomlinson-Harashima precoding is that even though the transmit signal power is limited due to the modulo operation, the received signal before the modulo operation at the receiver has a large dynamic range as the intersymbol interference is added by the channel. This puts strong requirements on the receiver hardware. In particular high resolution analog to digital converters and amplifiers with large linear range have to be applied.

An analysis of the error probability with Tomlinson-Harashima precoding unveils that Tomlinson-Harashima precoding shows advantages over a DFE at the receiver mainly for higher order modulation schemes, whereas despite the problem of error propagation, a DFE tends to be preferable in case of binary transmission.

Tomlinson-Harashima precoding is e.g. applied in high speed data modems.

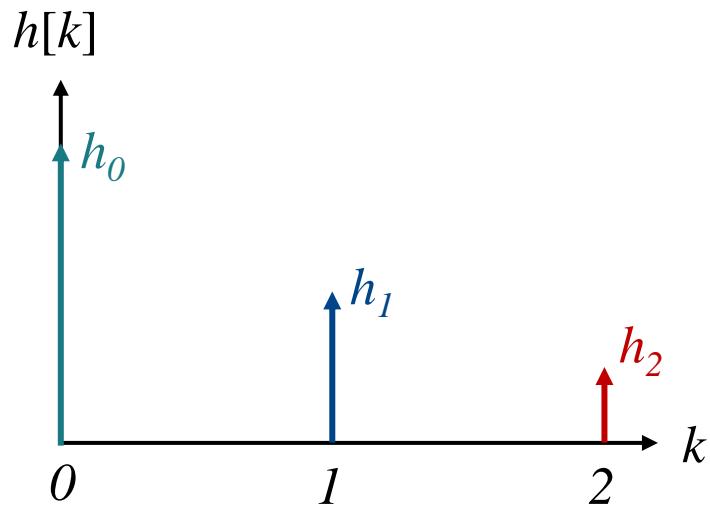
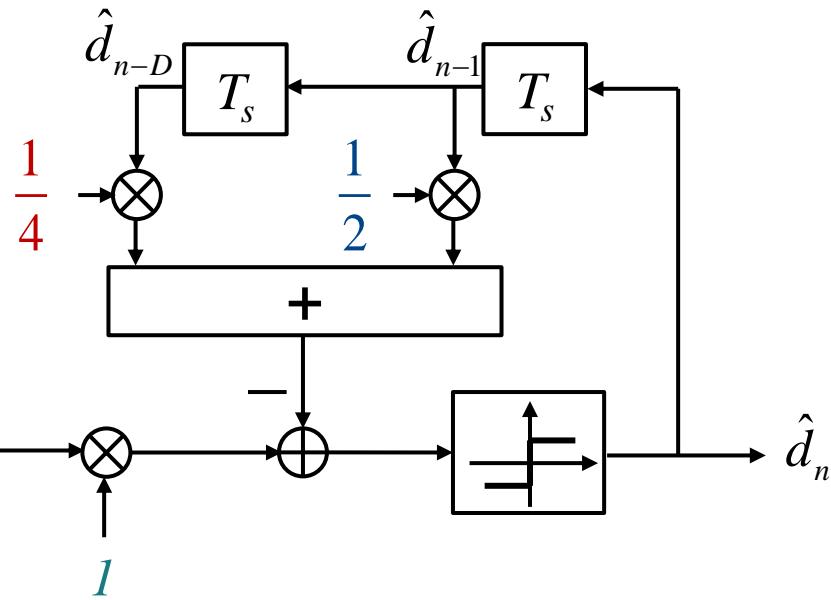
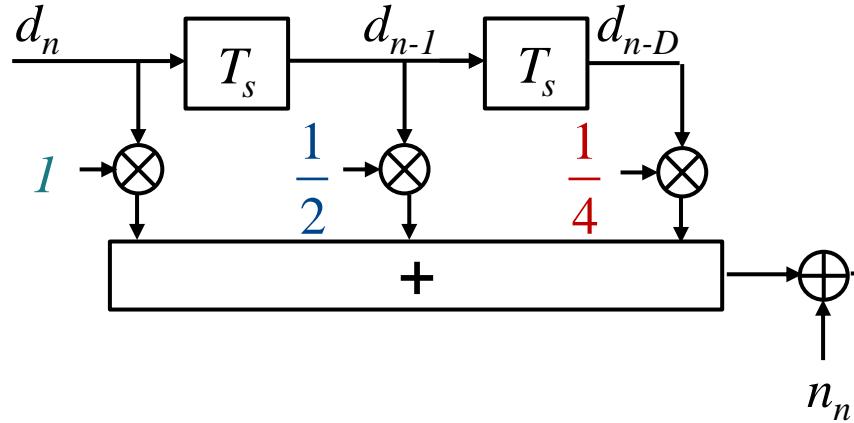
# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (1)



- + no noise enhancement
- decision error propagation

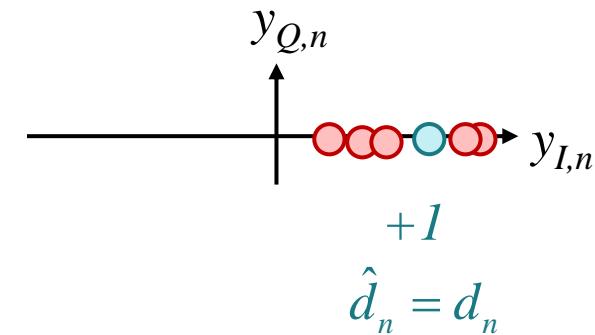
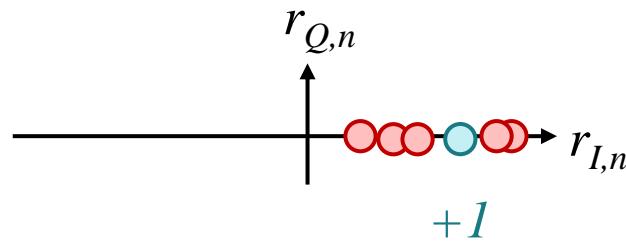
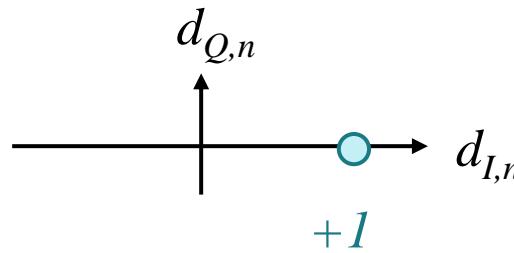
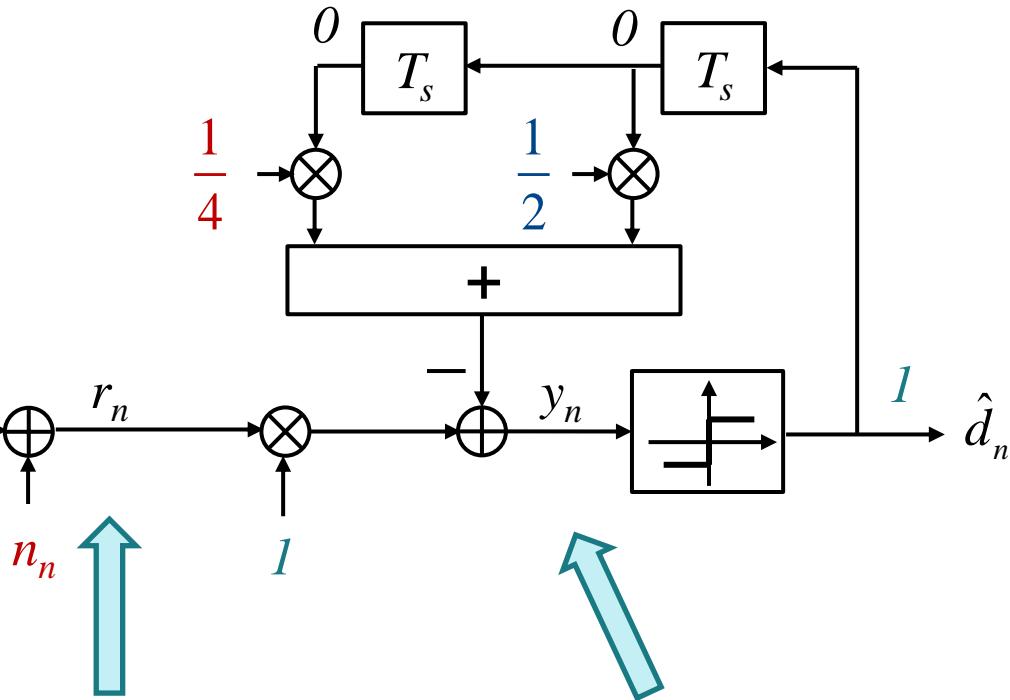
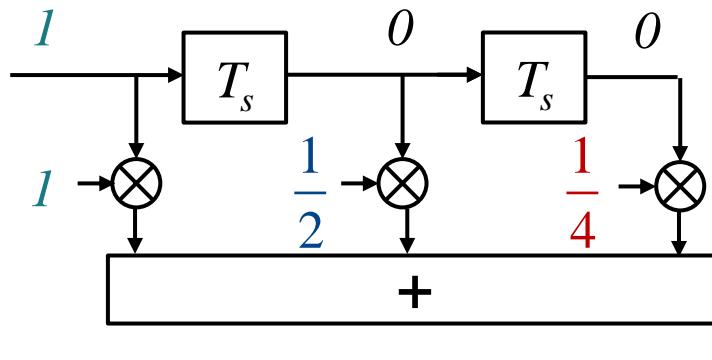
$$\hat{d}_n = Q \left\{ \frac{1}{h_0} r_n - \sum_{\nu=1}^D \frac{h_\nu}{h_0} \hat{d}_{n-\nu} \right\}$$

# Non-Linear Equalization: Decision Feedback Equalizer - Example (1)

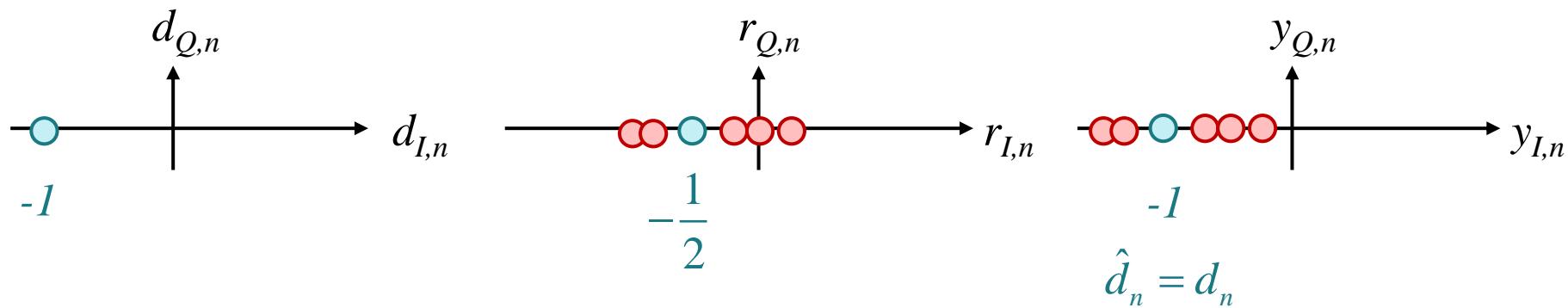
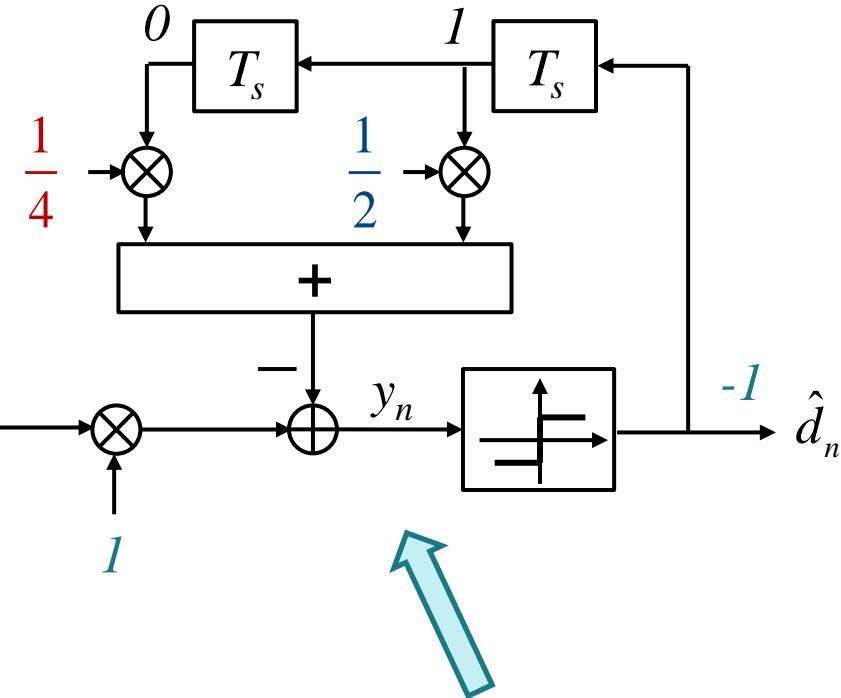
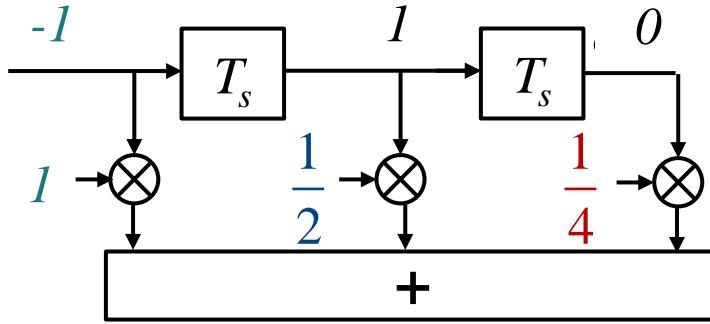


First tap  $h_0$  is strong !

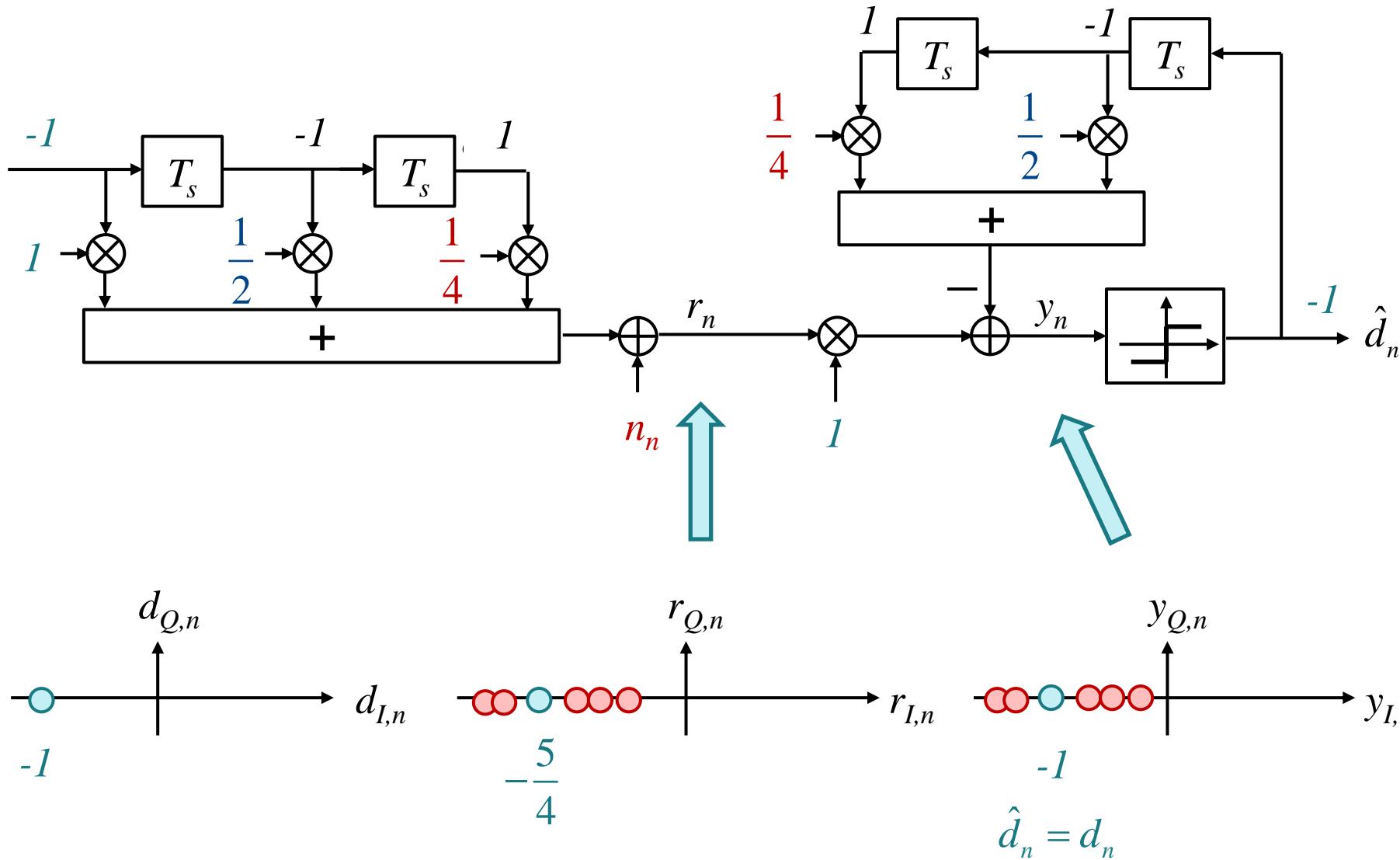
# Non-Linear Equalization: Decision Feedback Equalizer - Example (2)



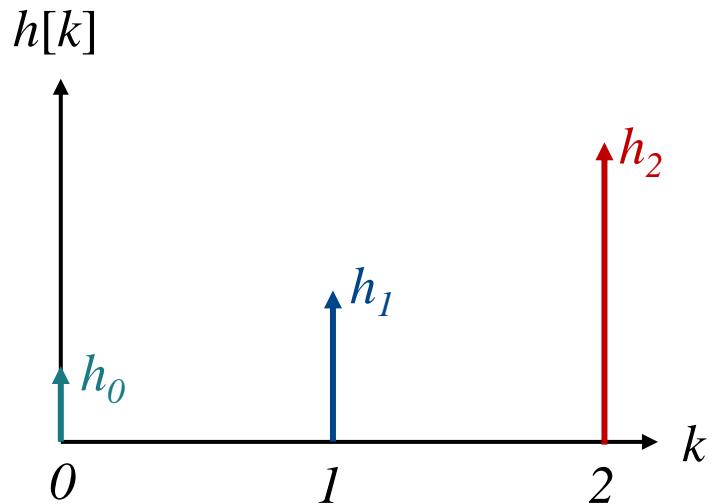
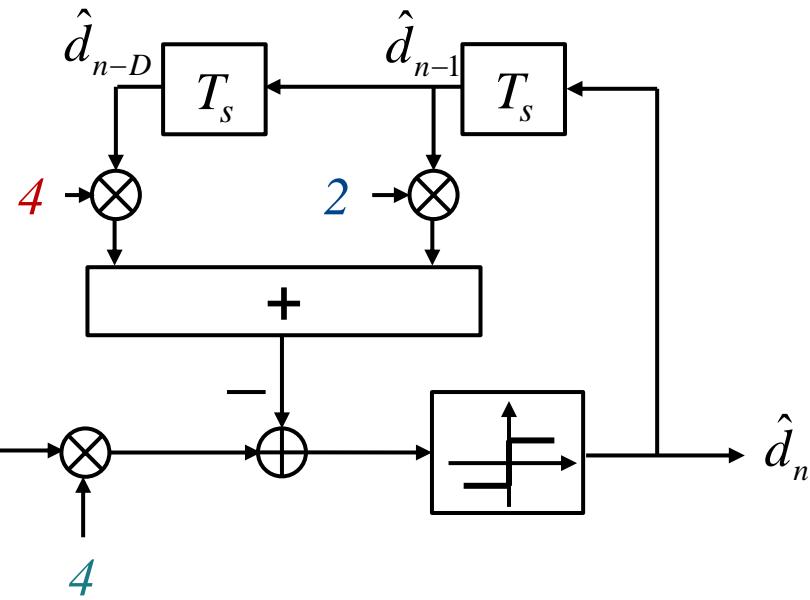
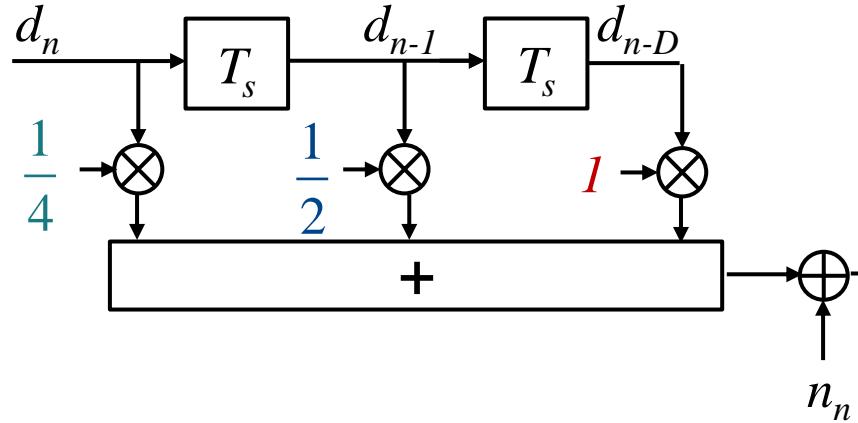
# Non-Linear Equalization: Decision Feedback Equalizer - Example (3)



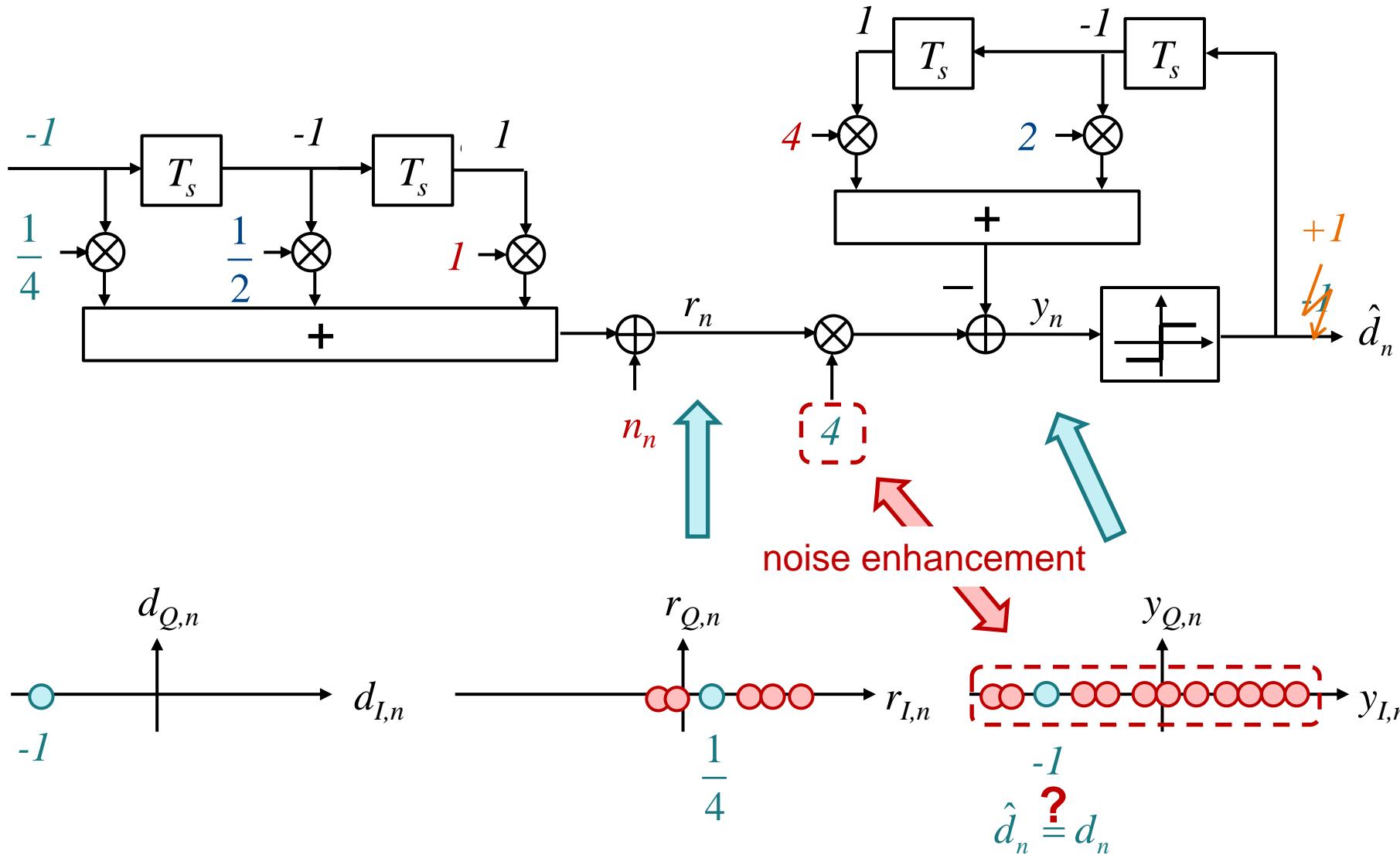
# Non-Linear Equalization: Decision Feedback Equalizer - Example (4)



# Non-Linear Equalization: Decision Feedback Equalizer - Example (5)

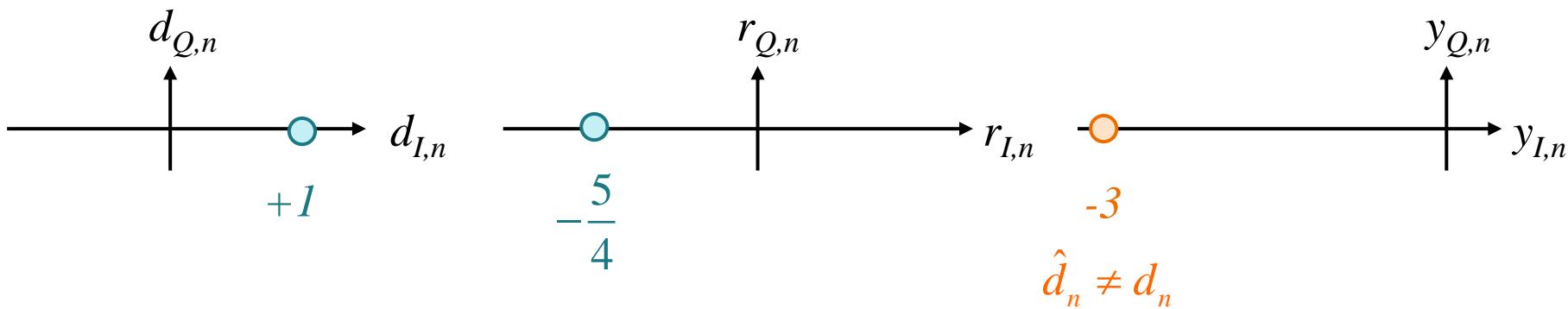
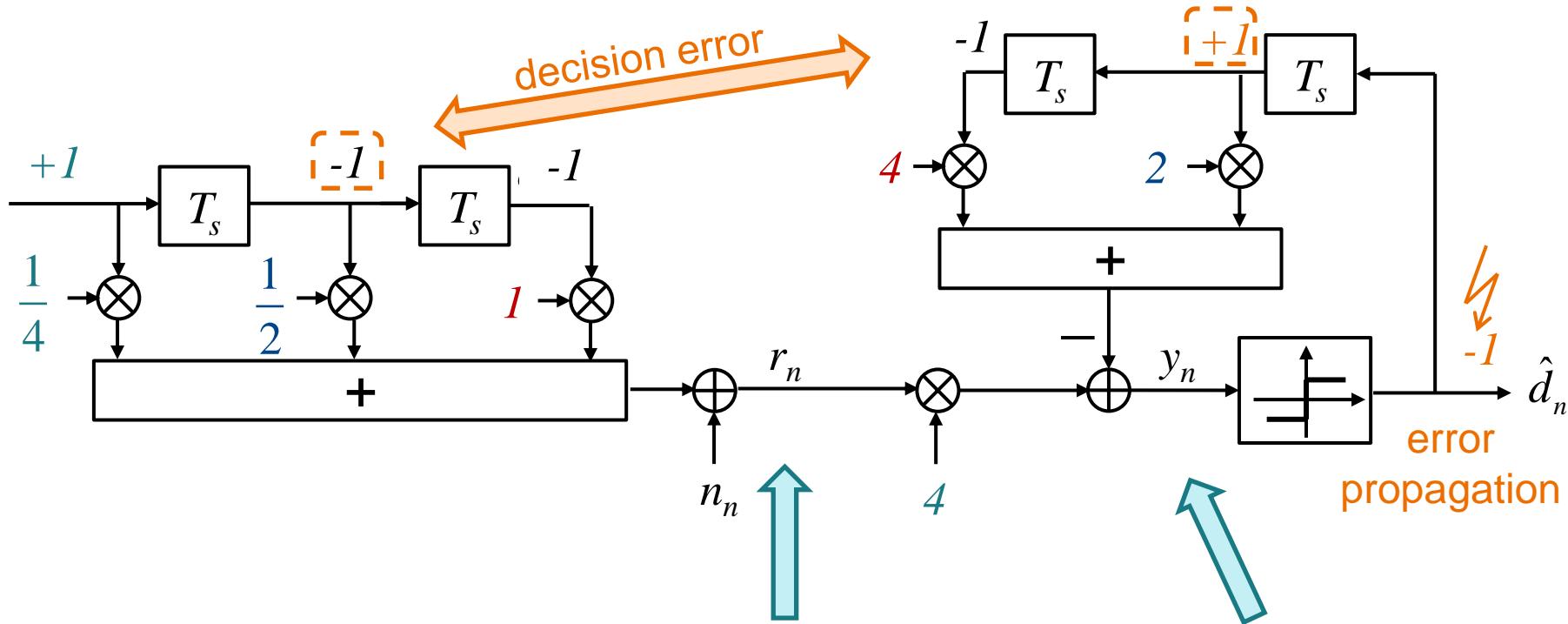


# Non-Linear Equalization: Decision Feedback Equalizer - Example (6)

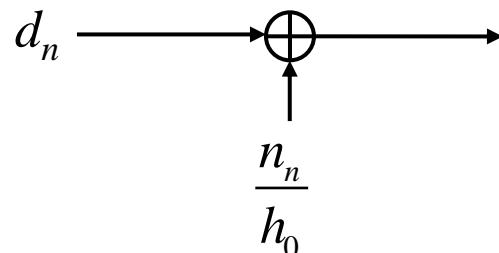
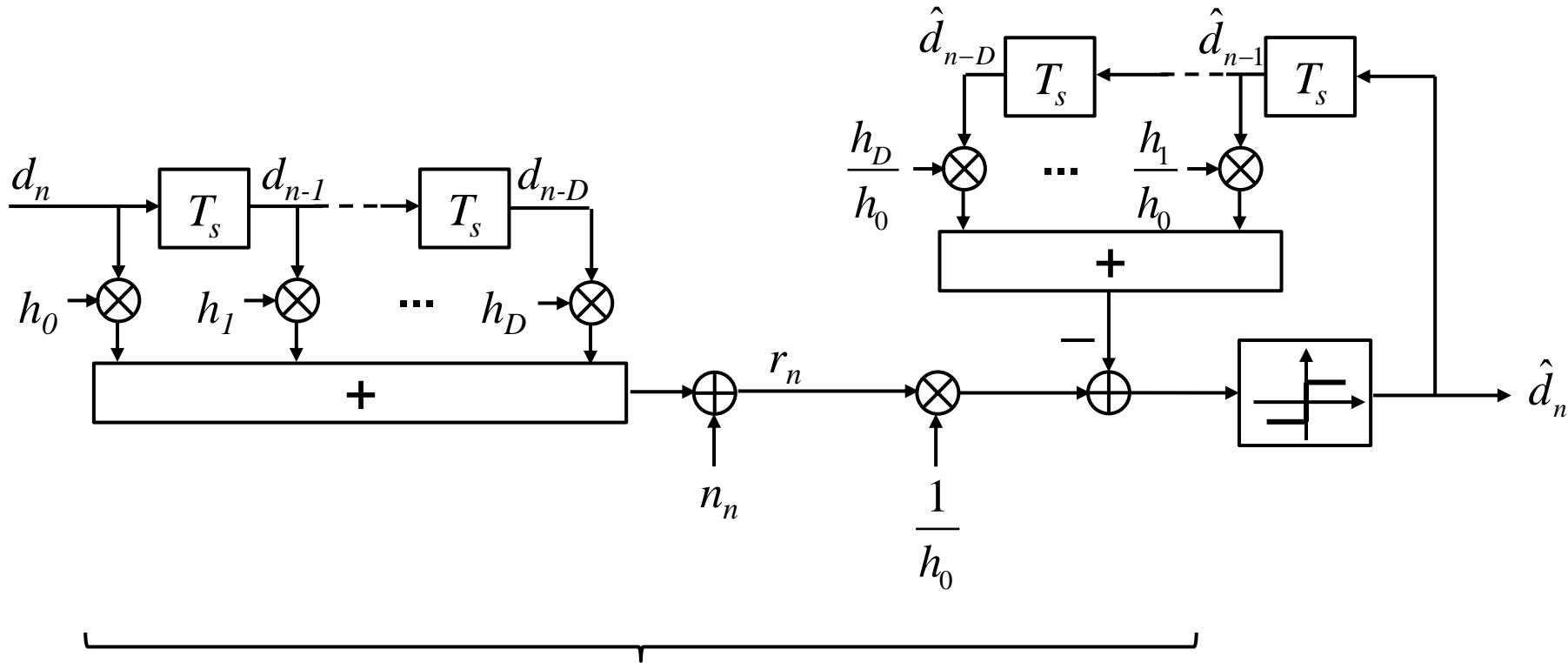


High risk of error propagation, if channel impulse response has weak precursors!

# Non-Linear Equalization: Decision Feedback Equalizer - Example (7)



# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (2)

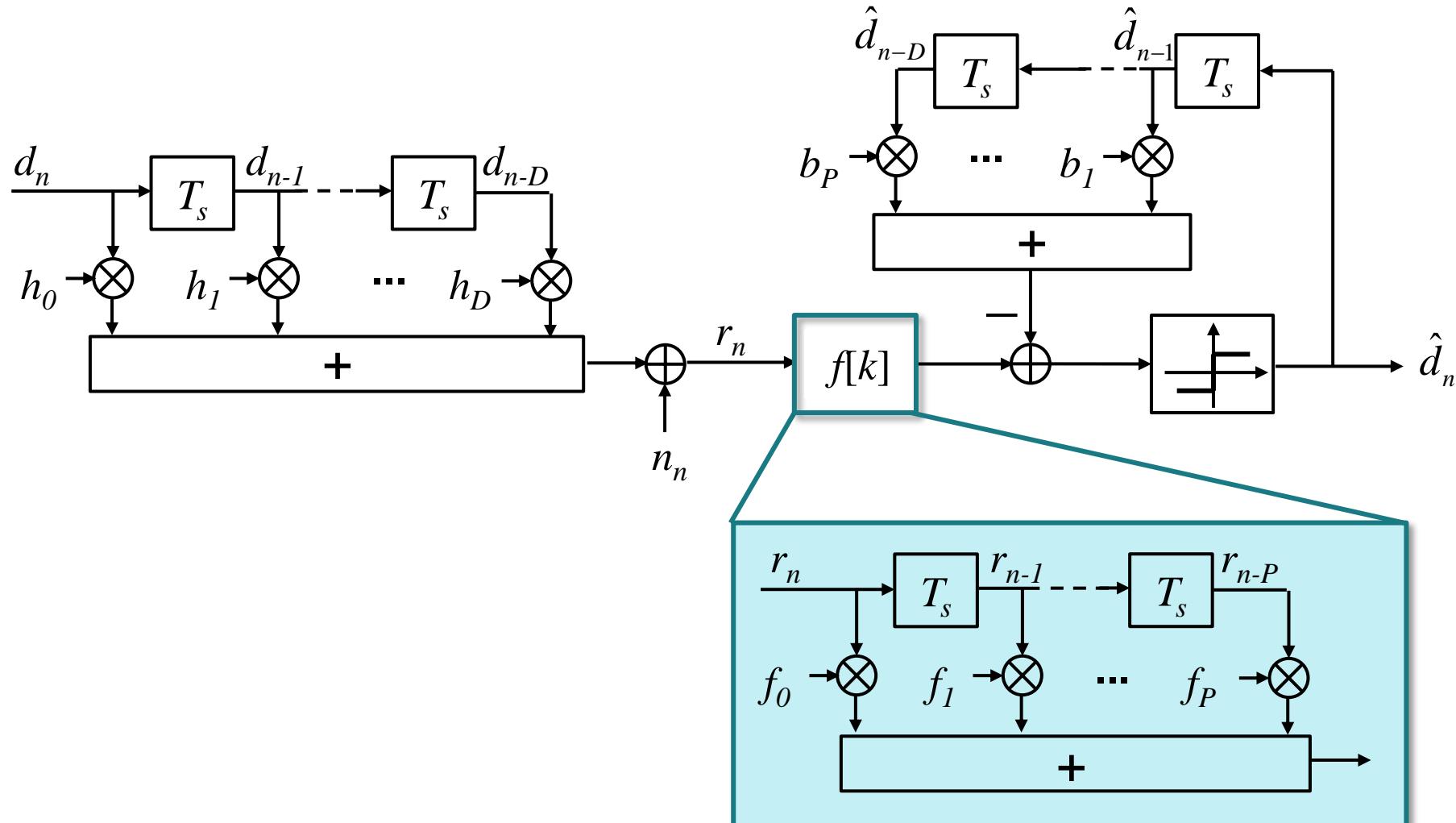


(in case of no decision errors)

**Problem:**  
Low effective SNR, if  $h_0$  is weak:

$$\text{SNR}_{\text{eff}} = \frac{E\{|d_n|^2\}|h_0|^2}{P_N}$$

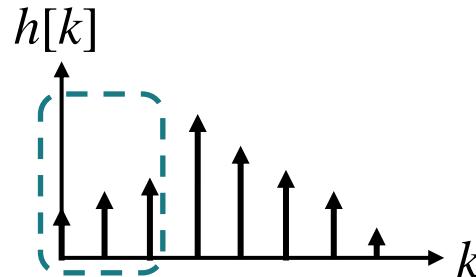
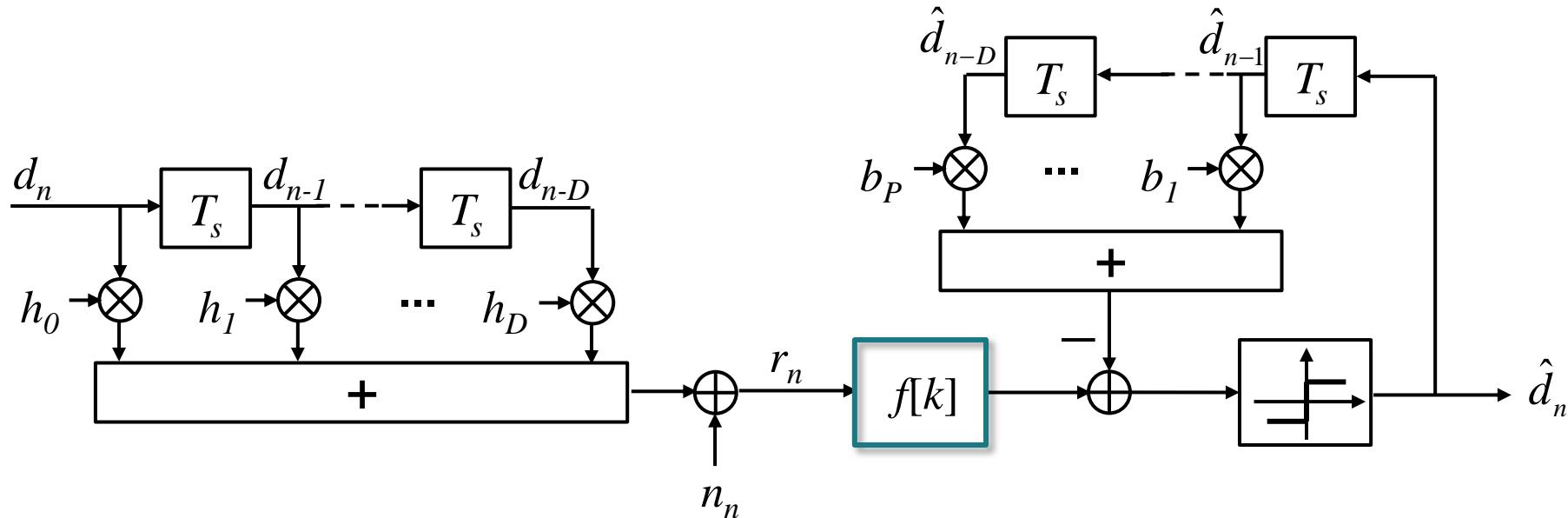
# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (3)



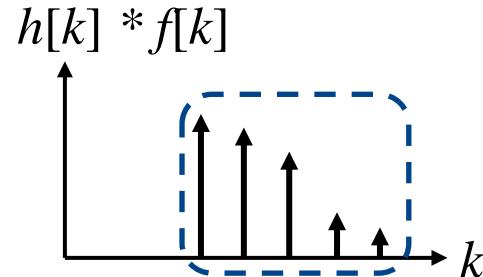
Non-recursive (FIR) linear equalizer in order to

- eliminate precursors in channel impulse response or
- make the channel approximately minimum phase

# Non-Linear Equalization: Decision Feedback Equalizer (DFE) (4)

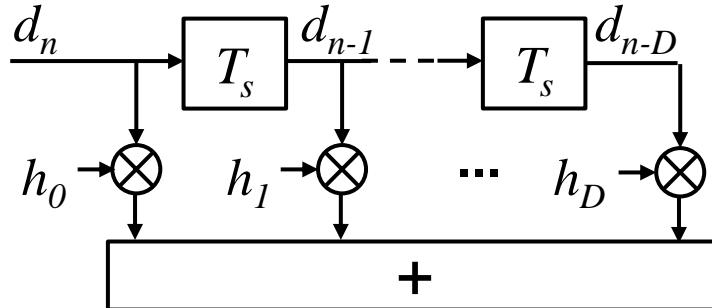


eliminated by  
FIR filter  $f[k]$



cancelled by  
decision  
feedback part

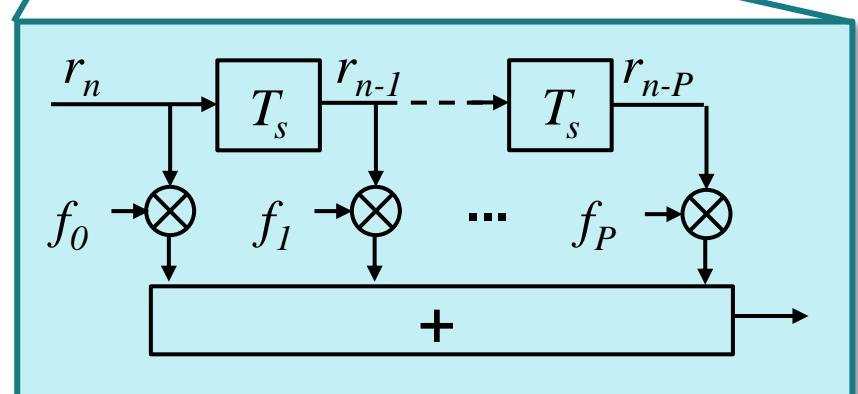
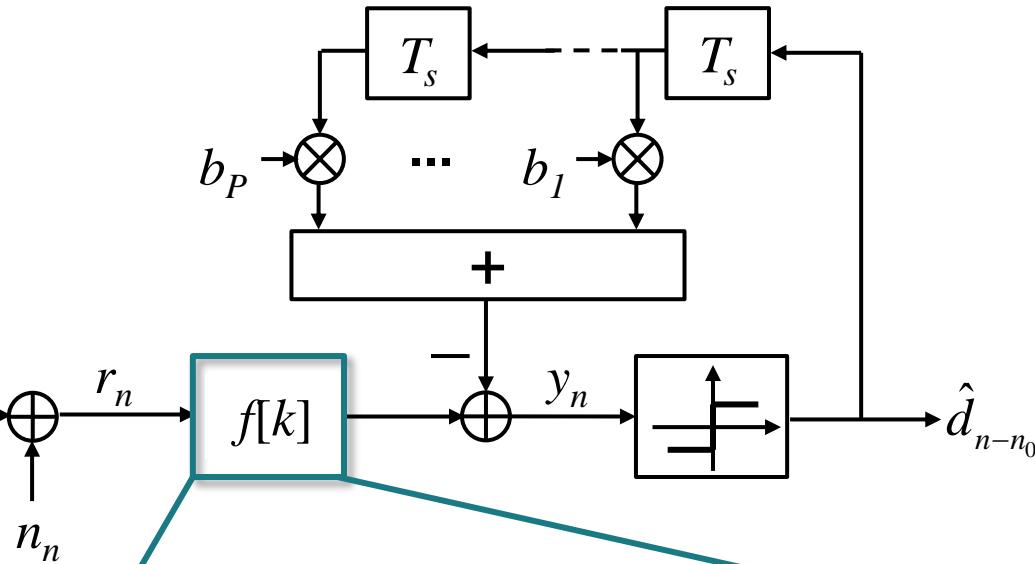
# Non-Linear Equalization: Decision Feedback MMSE Equalizer



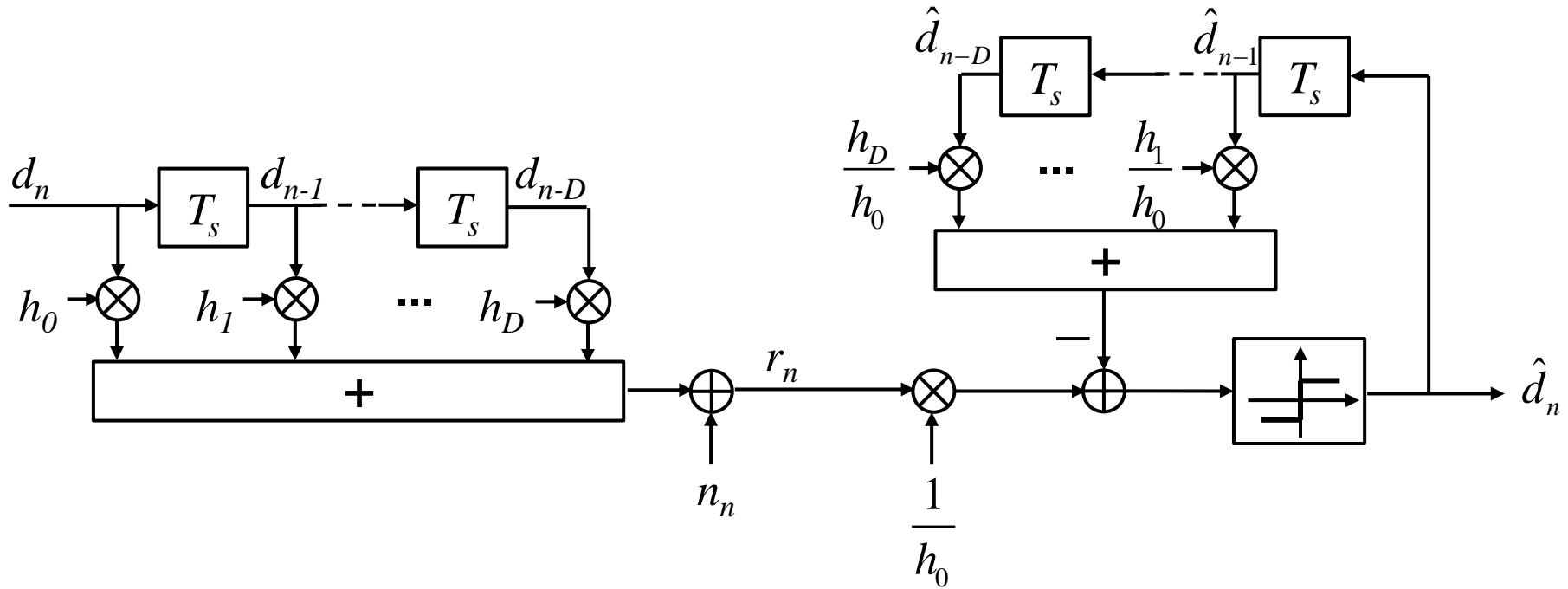
$$E\left\{\left|y_n - d_{n-n_0}\right|^2\right\} \rightarrow \min_{\mathbf{f}, \mathbf{b}}$$

$$\mathbf{f}_{\text{MMSE}} = \left[ \mathbf{R}_{RR} - \frac{P_N}{P_D} \mathbf{R}_{RD} \mathbf{R}_{RD}^H \right]^{-1} \mathbf{r}_{RD}$$

$$\mathbf{b}_{\text{MMSE}} = \frac{P_N}{P_D} \mathbf{R}_{RD}^H \mathbf{f}_{\text{MMSE}}$$



# Tomlinson-Harashima Precoding (1)

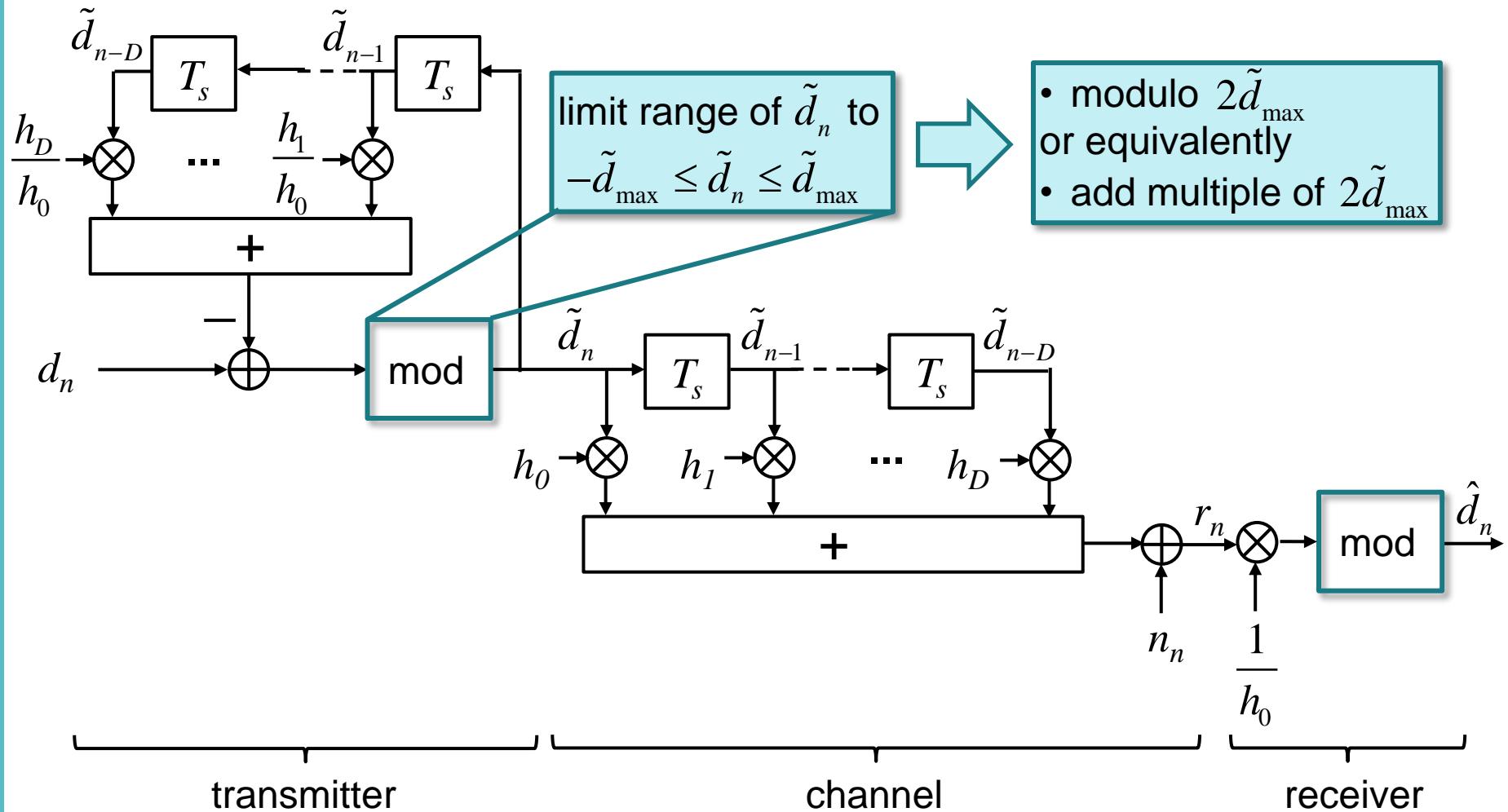


Major problem of decision feedback equalizer (DFE):  
Decision error propagation.

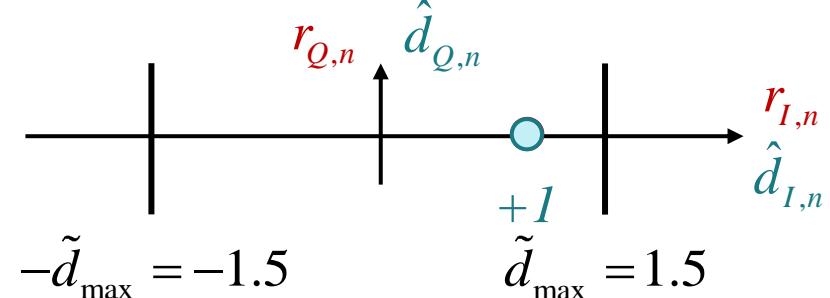
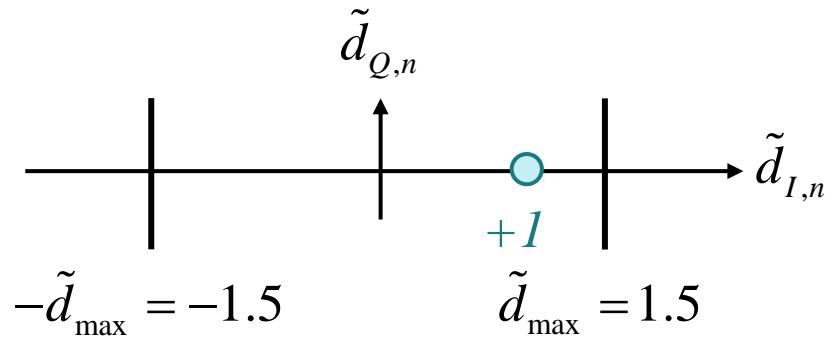
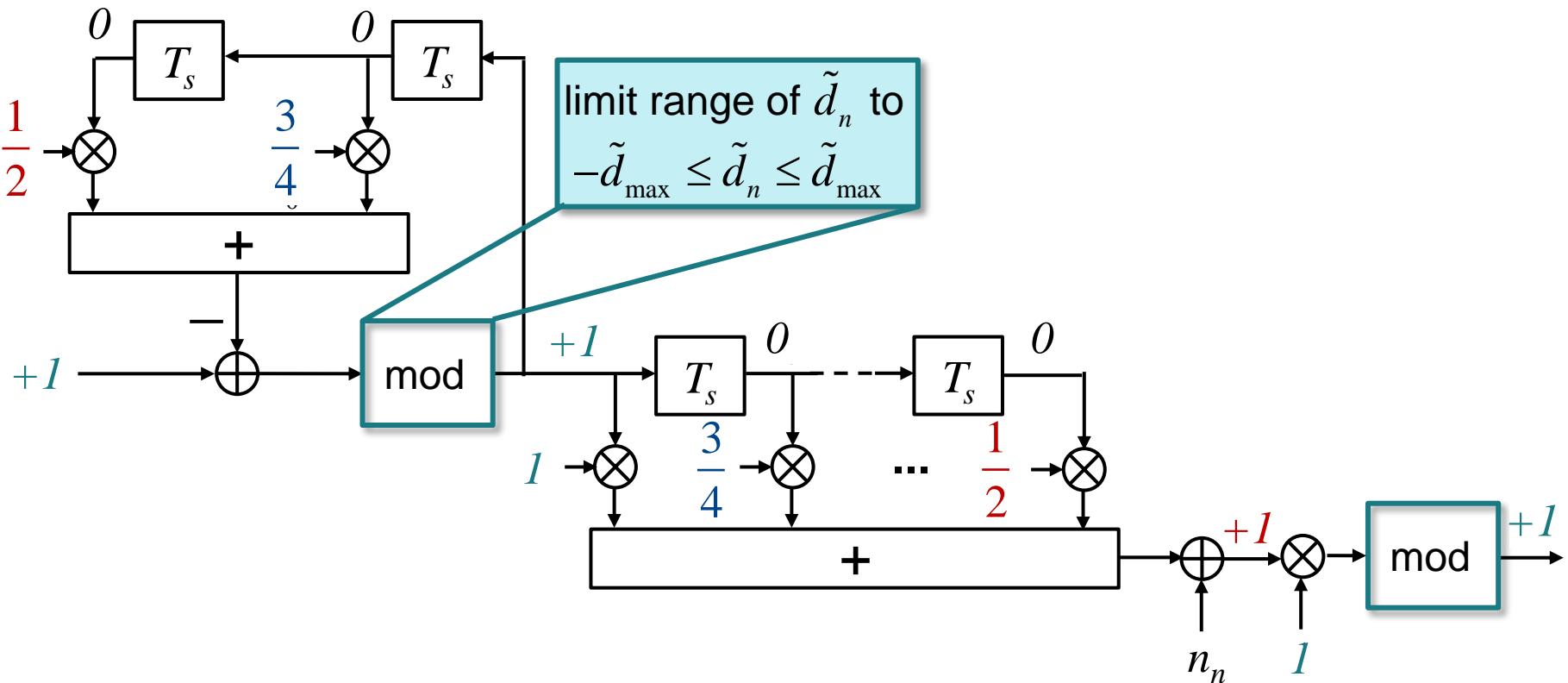
- Move decision feedback part to transmitter (Tomlinson-Harashima precoding).
- No errors since data is known at transmitter.

But: Channel impulse response  $h[k]$  must be known at transmitter!

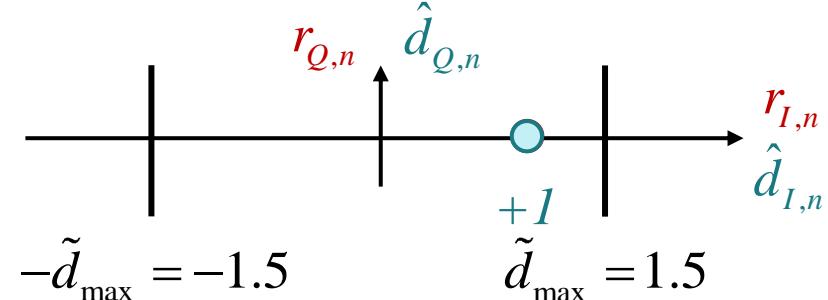
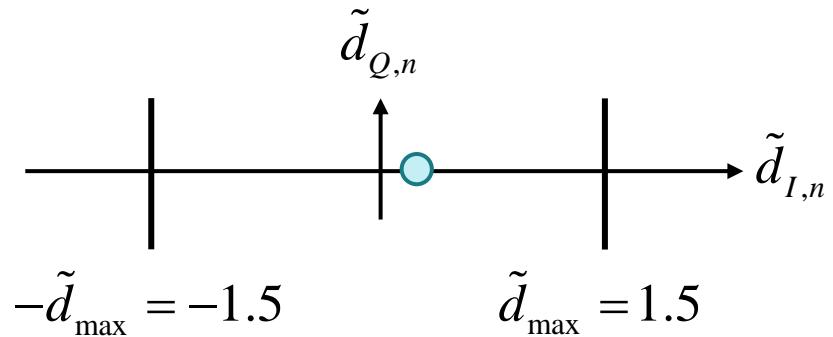
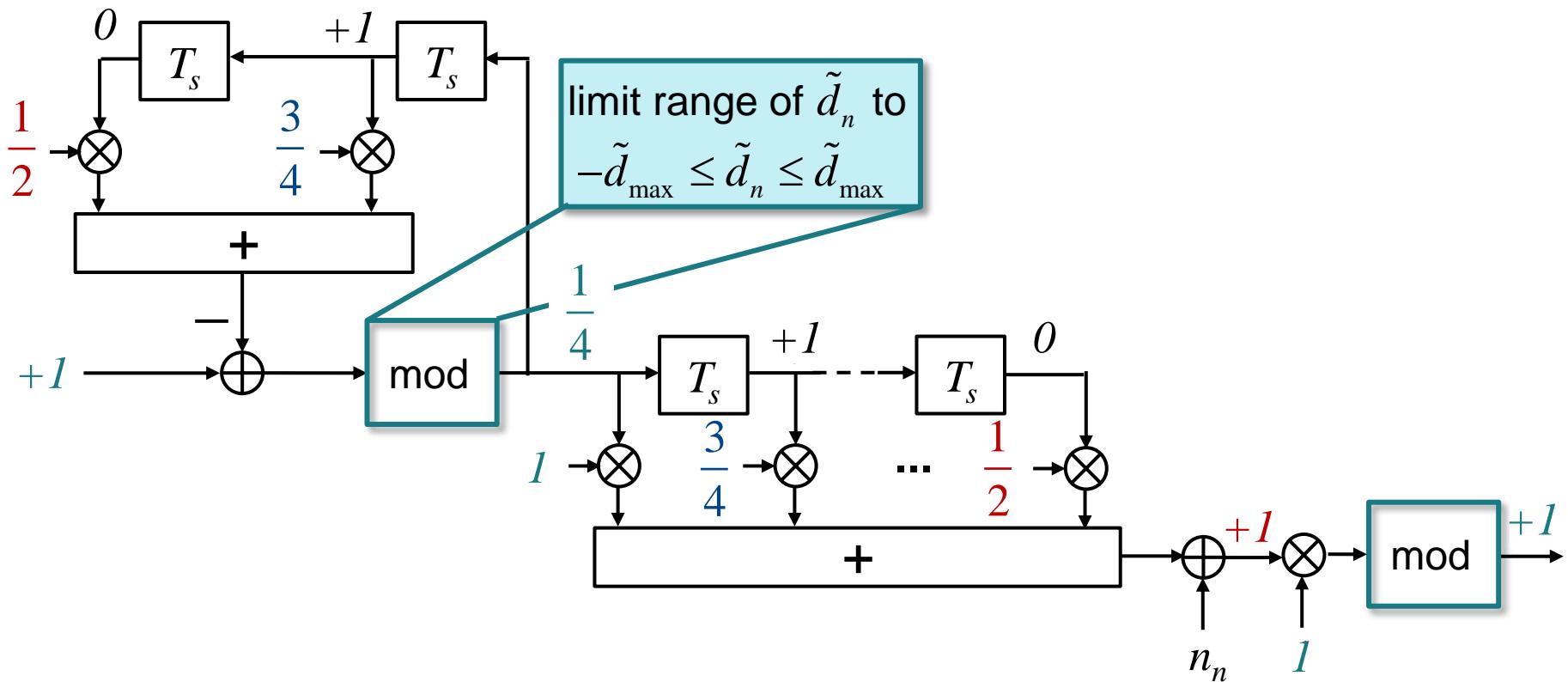
# Tomlinson-Harashima Precoding (2)



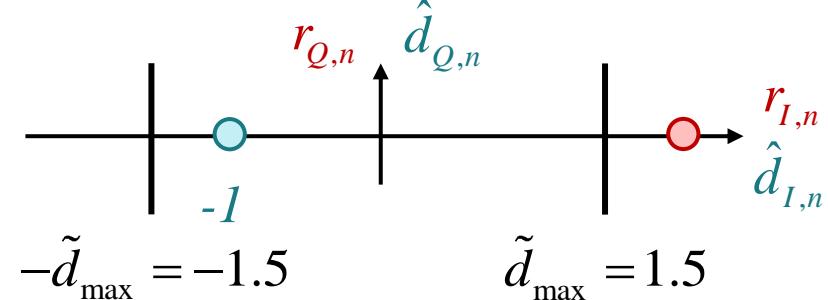
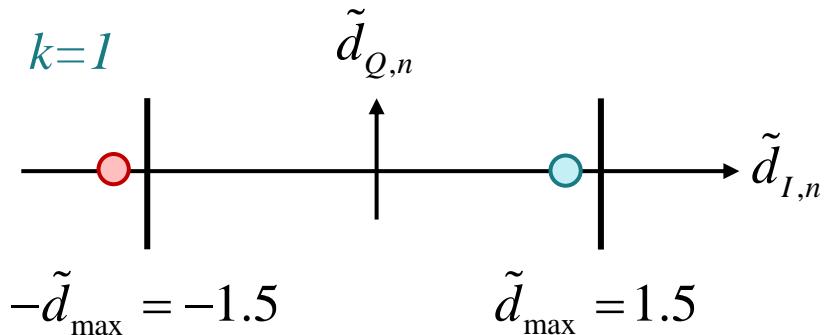
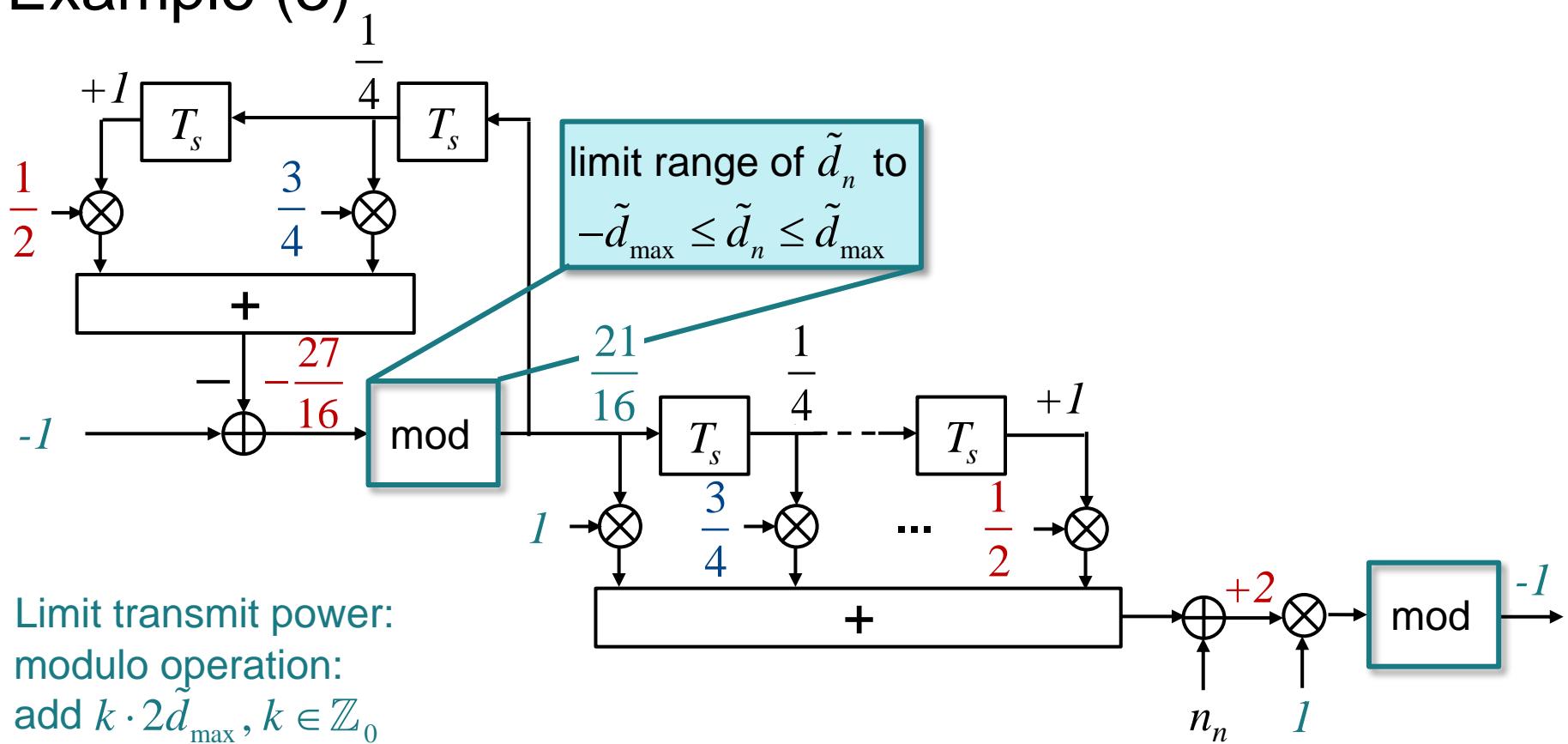
# Tomlinson-Harashima Precoding – Example (1)



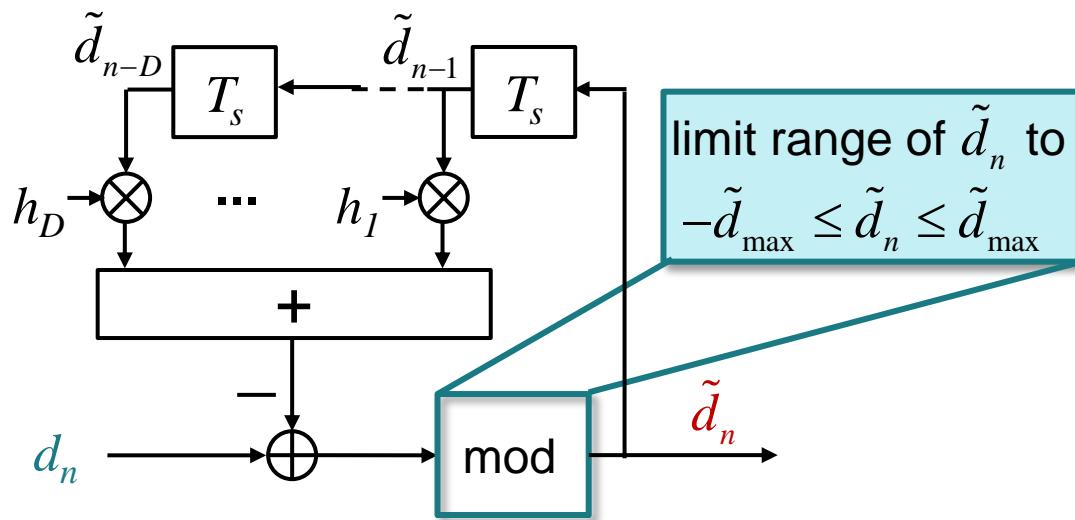
# Tomlinson-Harashima Precoding – Example (2)



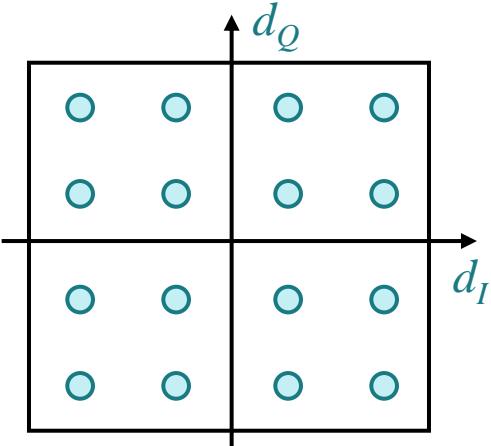
# Tomlinson-Harashima Precoding – Example (3)



# Tomlinson-Harashima Precoding: Transmit Signal Constellation



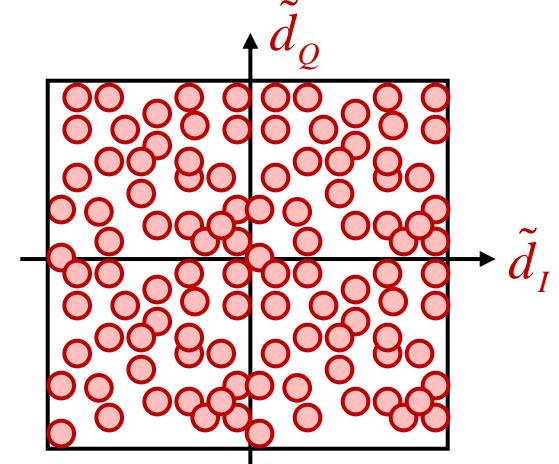
16-QAM



Tomlinson-Harashima  
precoding



Approximately  
uniform distribution



# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (1)

The optimum receiver in terms of minimizing the sequence error rate is a maximum a-posteriori probability (MAP) detector. A frequency-selective channel has memory, i.e. the received sample  $r_n$  at time  $n$  depends not only on the transmit symbol  $d_n$  at time  $n$ , but also on previous transmit symbols  $d_{n-1}, \dots, d_{n-D}$ . As a consequence, a MAP detector cannot make independent symbol-wise decisions as in a memoryless AWGN channel but has to estimate the transmitted data sequence

$$\mathbf{d} = [d_0, d_1, \dots, d_{N-1}]^T$$

based on the received channel output sequence

$$\mathbf{r} = [r_0, r_1, \dots, r_{N-1}]^T.$$

In case of an  $M$ -ary modulation scheme, there are  $M^N$  possible transmit sequences of length  $N$ . Since the MAP detector needs to compute the a-posteriori probability  $P(\mathbf{d}/\mathbf{r})$  for all of those candidate sequences  $\mathbf{d}$  in order to find the sequence  $\hat{\mathbf{d}}$  with highest a-posteriori probability, the computational complexity of MAP equalization in general increases exponentially with the sequence length  $N$ , which is prohibitive in most applications. Fortunately, there exists an algorithm, the *Viterbi algorithm*, which allows to perform MAP equalization with only linear complexity increase in the sequence length  $N$ .

# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (2)

A channel model is defined by the transition probabilities  $P(\mathbf{r}|\mathbf{d})$  rather than by the a posteriori probabilities  $P(\mathbf{d}|\mathbf{r})$ . Therefore, in order to determine the a posteriori probabilities, we reformulate the MAP criterion using Baye's rule and obtain

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{d}|\mathbf{r}) = \arg \max_{\mathbf{d}} \frac{P(\mathbf{r}|\mathbf{d})P(\mathbf{d})}{P(\mathbf{r})}.$$

The term  $P(\mathbf{r})$  is independent on the transmitted sequence  $\mathbf{d}$  and, hence, will have no impact on the result of the maximization with respect to  $\mathbf{d}$ . Therefore, it can be dropped and the MAP decision criterion simplifies to

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{r}|\mathbf{d})P(\mathbf{d}).$$

The term  $P(\mathbf{d})$  is the a priori probability of the transmit sequence  $\mathbf{d}$ . It may be known from knowledge on the source statistics, but in many cases, the a priori probability will be unknown at the receiver. Therefore, all  $M^N$  possible transmit sequences are assumed to be equally likely, i.e. the a priori probability becomes

$$P(\mathbf{d}) = \frac{1}{M^N},$$

which is independent of  $\mathbf{d}$  and, hence, can be dropped in the maximization.

# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (3)

For equally likely transmit sequences, the MAP decision criterion simplifies to

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{r}|\mathbf{d}).$$

The equation above is called the *maximum likelihood (ML)* decision criterion. Note, that MAP and ML equalization are equivalent in case of equally likely transmit sequences. However, in case the a priori probabilities of the transmit sequences differ, MAP equalization will yield a lower sequence error rate than ML equalization.

# MAP and ML Sequence Estimation in Frequency-Selective AWGN Channels (1)

We consider MAP and ML sequence estimation in a frequency-selective channel with memory  $D$  and additive white Gaussian noise  $n_n$ , i.e. the received channel output sample at time  $n$  is given by

$$r_n = \sum_{k=0}^D h_k d_{n-k} + n_n.$$

Even though the ISI channel has memory, the AWGN part is a memoryless channel with input symbols  $\sum h_k d_{n-k}$  and output symbols  $r_n$ . As the noise samples of white Gaussian noise are statistically independent, the channel transition probability can be expressed by

$$P(\mathbf{r}|\mathbf{d}) = \prod_{n=0}^{N-1} f_{R|D}(r_n | d_n),$$

where

$$f_{R|D}(r_n | d_n) = \frac{1}{2\pi\sigma_N^2} e^{-\frac{\left|r_n - \sum_{k=0}^D h_k d_{n-k}\right|^2}{2\sigma_N^2}}$$

is the Gaussian probability density function of the channel output  $r_n$  given the transmit symbols  $d_{n-D}, \dots, d_n$ .

# MAP and ML Sequence Estimation in Frequency-Selective AWGN Channels (2)

Similarly, we assume that the transmit symbols  $d_n$  are statistically independent, i.e.

$$P(\mathbf{d}) = \prod_{n=0}^{N-1} P(d_n).$$

With the above assumptions, the MAP decision rule becomes

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \{P(\mathbf{r}|\mathbf{d})P(\mathbf{d})\} = \arg \max_{\mathbf{d}} \left\{ \prod_{n=0}^{N-1} f_{R|D}(r_n | d_n) P(d_n) \right\}.$$

The exponential function in the Gaussian probability density function  $f_{R/D}(r_n/d_n)$  is inconvenient and computationally expensive. Therefore, we maximize the logarithm of the expression above which does not change the maximization result  $\hat{\mathbf{d}}$ .

The MAP decision rule then reads

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} [\log f_{R|D}(r_n | d_n) + \log P(d_n)] \right\}.$$

# MAP and ML Sequence Estimation in Frequency-Selective AWGN Channels (3)

Using the logarithm

$$\log f_{R|D}(r_n | d_n) = \log \frac{1}{2\pi\sigma_N^2} - \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2}$$

of the Gaussian probability density function, we obtain

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} \left[ \log \frac{1}{2\pi\sigma_N^2} - \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} + \log P(d_n) \right] \right\}.$$

The additive term  $\log(1/(2\pi\sigma_N^2))$  is a constant which is independent of  $\mathbf{d}$  and, hence, can be dropped in the maximization. Moreover, maximizing the expression above is equivalent to minimizing the negative expression. Hence, we can get rid of the minus sign and obtain

$$\hat{\mathbf{d}} = \arg \min_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] \right\}.$$

# MAP and ML Sequence Estimation in Frequency-Selective AWGN Channels (4)

For an efficient algorithmic computation of the expression above, it is useful to split the sum into three parts: From the viewpoint at time  $m$ , we may express the MAP decision rule as a first term containing samples from the past, a second term containing the present samples at time  $m$ , and a third term containing all future samples:

$$\hat{\mathbf{d}} = \arg \min_{\mathbf{d}} \left\{ \sum_{n=0}^{m-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] + \frac{\left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2}{2\sigma_N^2} \right. \\ \left. - \log P(d_m) + \sum_{n=m+1}^{N-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] \right\} \quad (1).$$

Using this representation, we can compute the a posteriori probability of each candidate sequence stepwise. This idea is pursued in the Viterbi algorithm, where the second term in the expression above is successively added as the metric increment

$$\Delta \mu_m^{s_i \rightarrow s_j} = \frac{\left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2}{2\sigma_N^2} - \log P(d_m).$$

# MAP and ML Equalization: Viterbi Algorithm (1)

The Viterbi algorithm works on a trellis representation of the possible transmit sequences. The trellis is a graphical representation over time of all possible transmit sequences consisting of states and transitions between the states. A state  $s_i$  is defined by the symbols in the memory elements of the discrete-time ISI channel model, i.e. by the transmit symbols  $d_{n-1}, d_{n-2}, \dots, d_{n-D}$ . Hence, for an  $M$ -ary modulation scheme and a channel memory  $D$ , the trellis has  $M^D$  states. At time  $m$ , a new symbol  $d_m$  is shifted into the channel tapped delay line causing a transition in the trellis from state  $s_i$  to state  $s_j$ . The trellis has  $M$  transitions for each state  $s_i$  to different successor states  $s_j$ .

Each possible transmit sequence  $\mathbf{d}$  defines a path through the trellis. It is the task of the equalizer to find the MAP or ML path through the trellis.

Viterbi's essential finding was the fact that a path which is suboptimum at a certain point  $m$  in time will never become optimum and, therefore, can be dropped at time  $m$ . Based on a metric  $\mu_m^{s_j}$ , the Viterbi algorithm finds the most likely out of the  $M$  paths entering a state  $s_j$  at time  $m$ . As all paths entering a state will have a common future through the trellis and, hence the same metric increments, only the most likely path needs to be further pursued at each state. All  $M-1$  other paths can be dropped. Since only one path per state is considered further, the complexity of the Viterbi algorithm grows only linearly with the sequence length  $N$  rather than exponentially as it would be the case when the a posteriori probabilities of all  $M^N$  possible transmit sequences were evaluated.

# MAP and ML Equalization: Viterbi Algorithm (2)

The metric used by the Viterbi algorithm is basically the expression (1) above. At time  $n=0$ , the metrics  $\mu_0^{s_j}$  of all states  $s_j$  are initialized. Typically,  $D$  known symbols  $d_{D}, \dots, d_2, d_1$  are transmitted in the beginning in order to initialize the channel in a well defined state. Without loss of generality, we assume that the channel is initialized in the state  $s_0$ . As due to the initialization, the state at time  $n=0$  is known, we initialize the metric of state  $s_0$  by  $\mu_0^{s_0} = 0$  and all other states by  $\mu_0^{s_j} = \infty, j \neq 0$ .

A transition  $s_i \rightarrow s_j$  from state  $s_i$  to state  $s_j$  at time  $m$  is associated with a certain receive sample

$$\sum_{m=0}^D h_k d_{m-k}$$

before adding the noise, where the symbols  $d_{m-1}, d_{m-2}, \dots, d_{m-D}$  are determined by the originating state  $s_i$  and the symbol  $d_m$  determines the transition to the successor state  $s_j$ . Corresponding to equation (1) above, the transition  $s_i \rightarrow s_j$  is associated with a metric increment

$$\Delta \mu_m^{s_i \rightarrow s_j} = \frac{\left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2}{2\sigma_N^2} - \log P(d_m)$$

i.e. the metric increment is given by the squared Euclidean distance between the received sample  $r_n$  and the noiseless receive sample  $\sum h_k d_{m-k}$  which labels the transition plus the a priori information of the symbol  $d_m$  which causes the transition  $s_i \rightarrow s_j$ .

# MAP and ML Equalization: Viterbi Algorithm (3)

In order to determine the metric of a state  $s_j$  at time  $m+1$ , the metric of each incoming path is determined according to

$$\mu_m^{s_i} + \Delta\mu_m^{s_i \rightarrow s_j},$$

where  $\mu_m^{s_i}$  is the metric of the predecessor state  $s_i$  for the transition  $s_i \rightarrow s_j$  at time  $m$ . As according to equation (1), the most likely path is the path with minimum metric, the metric of state  $s_j$  at time  $m+1$  is obtained from the minimum metrics among all incoming paths, i.e.

$$\mu_{m+1}^{s_j} = \min_i \left\{ \mu_m^{s_i} + \Delta\mu_m^{s_i \rightarrow s_j} \right\}.$$

Only the path corresponding to the minimum metric is kept while all other incoming paths are eliminated from the trellis.

The same procedure is continued for all states up to the end of the trellis. At the end of the trellis, we identify the state  $s_j$  with minimum metric  $\mu_N^{s_j}$ . The surviving path in the respective state  $s_j$  at time  $N$  is the MAP path.

As the MAP path is labelled with a particular data sequence  $\mathbf{d}$ , the MAP estimate  $\hat{\mathbf{d}}$  on the data sequence is obtained from tracing back the MAP path.

In order to decide for the MAP path, the Viterbi algorithm needs to be executed for the complete data sequence. This requires to store all surviving paths and, even more important, it causes a significant decision delay which is a problem in some applications which require low latency. Therefore, in practice the Viterbi algorithm is often implemented with a limited decision delay.

# MAP and ML Equalization: Viterbi Algorithm (4)

I.e., the decision on the data symbol  $d_n$  is taken already by tracing back the path with highest metric at time  $n+L$ , where  $L$  is called the decision delay. Even though it cannot be guaranteed that  $\hat{d}_n$  is on the MAP path, for sufficiently long decision delay  $L$ , all paths other than the MAP path will be already eliminated with high probability for the deision on  $d_n$ . As a rule of thumb, the decision delay should be about five times the channel memory, i.e.  $L \approx 5D$ .

The Viterbi algorithm can also be implemented as a maximum likelihood (ML) sequence estimator rather than a MAP sequence estimator. For ML sequence estimation, the Viterbi algorithm works in exactly the same way as for MAP sequence estimation with the only difference in the metric increment: The a priori probabilities  $P(d_n)$  in the MAP metric increment

$$\Delta\mu_m^{s_i \rightarrow s_j} = \frac{\left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2}{2\sigma_N^2} - \log P(d_m)$$

are dropped. As a consequence, the noise variance  $2\sigma_N^2$  is now only a constant scaling factor of all metric increments throughout the trellis, which does not impact the minimization result. Therefore, the factor  $2\sigma_N^2$  can be dropped and the ML metric increment becomes

$$\Delta\mu_m^{s_i \rightarrow s_j} = \left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2.$$

# MAP and ML Equalization:

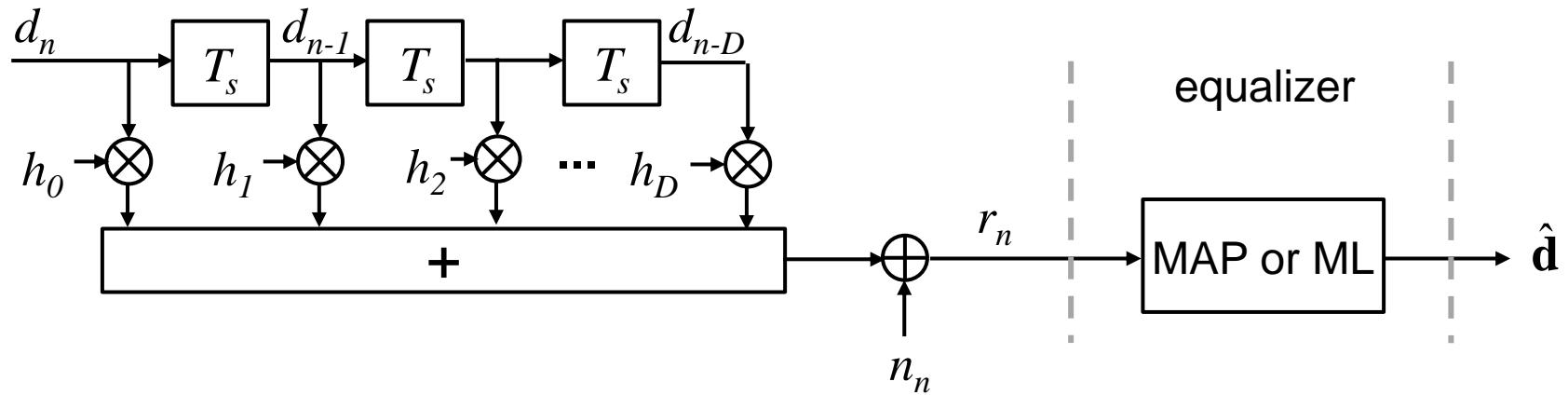
## Viterbi Algorithm (5)

Dropping the factor  $2\sigma_N^{-2}$  in the metric increment has an important practical aspect: In MAP equalization, the factor  $2\sigma_N^{-2}$  is needed as the weighting factor among the channel information and the a priori information. Hence, the noise variance needs to be estimated at the receiver. However, precise estimation of the noise variance based on a limited observation window is a challenging task. On the other hand, an inaccurate estimate of the noise variance will degrade the performance of MAP equalization due to inaccurate weighting of channel information and a priori information in the metric computation. Depending on the estimation mismatch, this degradation may even result in worse performance compared to the actually simpler ML equalizer.

In summary, unlike MAP equalization, ML equalization does neither require estimation of the noise variance nor a priori information about the transmit symbol statistics. Therefore, it is not only computationally simpler but is also more robust.

Note, that the Viterbi algorithm is a sequence estimator. It detects the MAP or ML sequence  $\hat{\mathbf{d}}$  and, therefore minimizes the sequence error probability. However, this is not equivalent to minimizing the symbol error probability. There exists a trellis based equalization algorithm called BCJR algorithm (named after the inventors **Bahl, Cocke, Jelinek, Raviv**), which delivers symbolwise MAP or ML decisions for the individual symbols  $d_n$  rather than for the total transmit sequence  $\mathbf{d}$ . The BCJR algorithm is a bit more complex than the Viterbi algorithm. Simply put, it consists of two Viterbi-type metric computations. One metric is computed starting from the beginning of the trellis towards the end. The other metric computation runs from the end of the trellis towards the beginning. Both metrics are combined in order to symbolwise determine the equalizer outputs  $\hat{d}_n$ . While the Viterbi algorithm is a sequence estimator which minimizes the sequence error probability, the BCJR algorithm is a symbol estimator which minimizes the symbol error probability.

# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (1)



Maximum a-posteriori (MAP) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{d} | \mathbf{r})$$

Maximum likelihood (ML) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{r} | \mathbf{d})$$

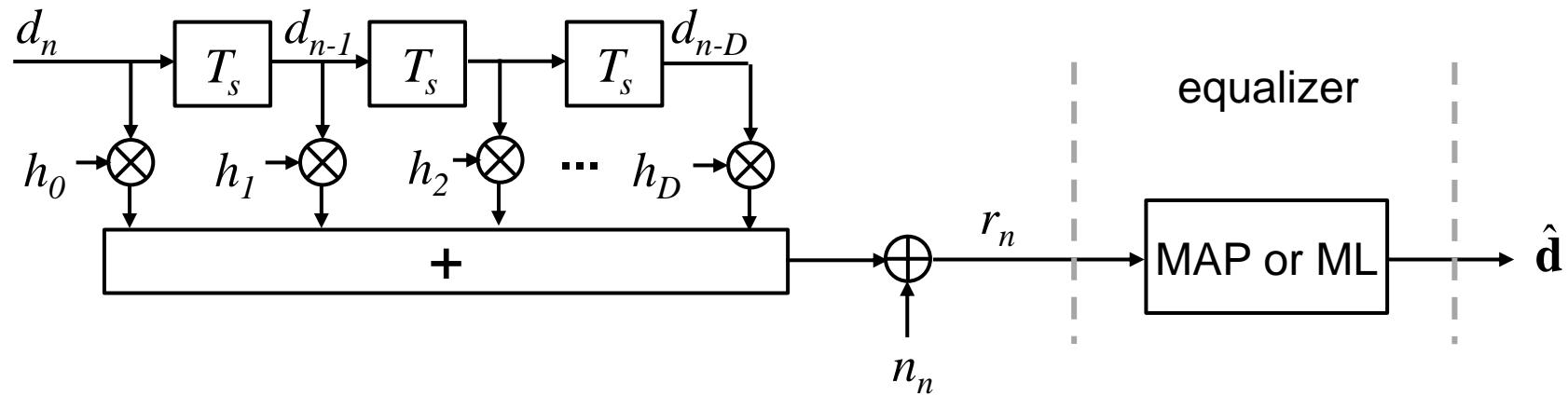
$$\mathbf{d} = [d_0, d_1, \dots, d_{N-1}]$$

$$\mathbf{r} = [r_0, r_1, \dots, r_{N-1}]$$

$$\hat{\mathbf{d}} = [\hat{d}_0, \hat{d}_1, \dots, \hat{d}_{N-1}]$$

→ sequences!

# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (2)



Maximum a-posteriori (MAP) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{d} | \mathbf{r}) = \arg \max_{\mathbf{d}} P(\mathbf{r} | \mathbf{d}) \underbrace{P(\mathbf{d})}_{\text{MAP requires knowledge of statistics of transmit data.}}$$

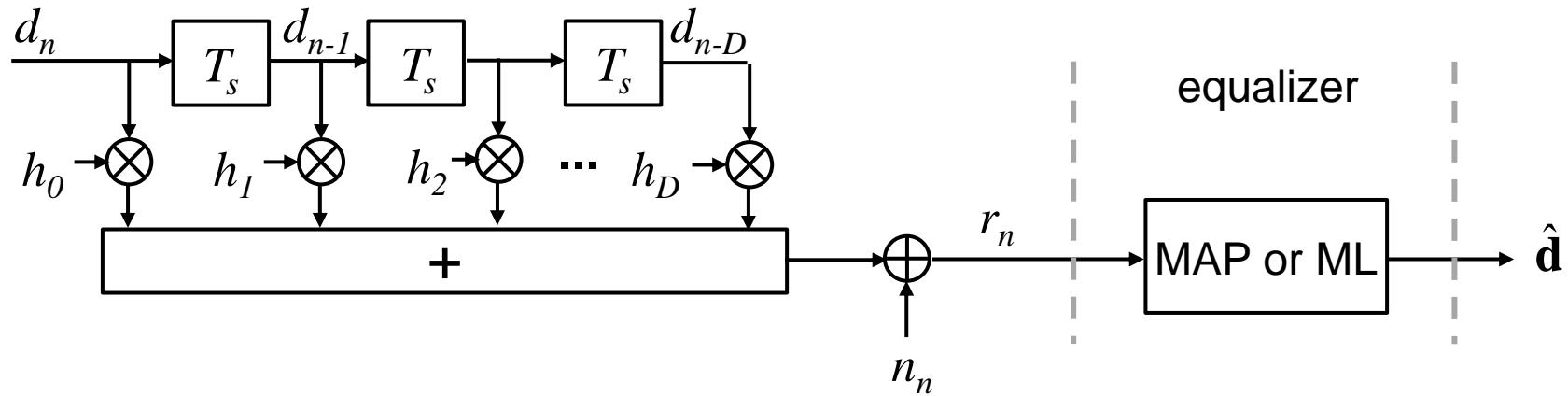
MAP requires knowledge of statistics of transmit data.

Maximum likelihood (ML) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{r} | \mathbf{d})$$

For equally likely transmit sequences, i.e.  
 $P(\mathbf{d})=\text{const}$ , MAP and ML detection are equivalent.

# Maximum A-Posteriori (MAP) and Maximum Likelihood (ML) Sequence Estimation (3)



Maximum a-posteriori (MAP) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} P(\mathbf{d} | \mathbf{r}) = \arg \max_{\mathbf{d}} \frac{P(\mathbf{r} | \mathbf{d}) P(\mathbf{d})}{P(\mathbf{r})}$$

↑  
Bayes rule

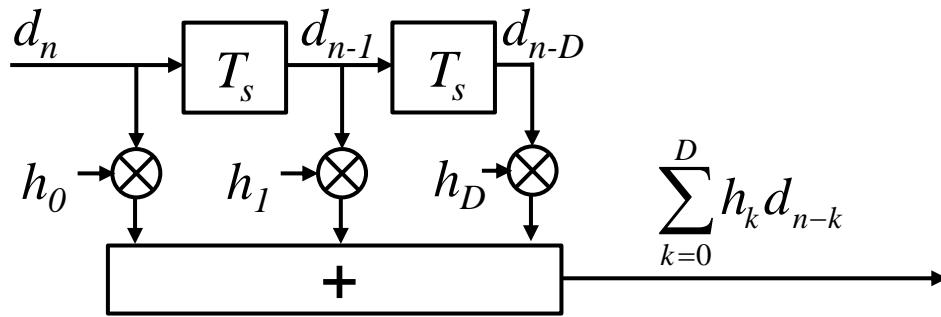
independent of  $\mathbf{d}$   
 → no impact on maximization

$$= \arg \max_{\mathbf{d}} P(\mathbf{r} | \mathbf{d})$$

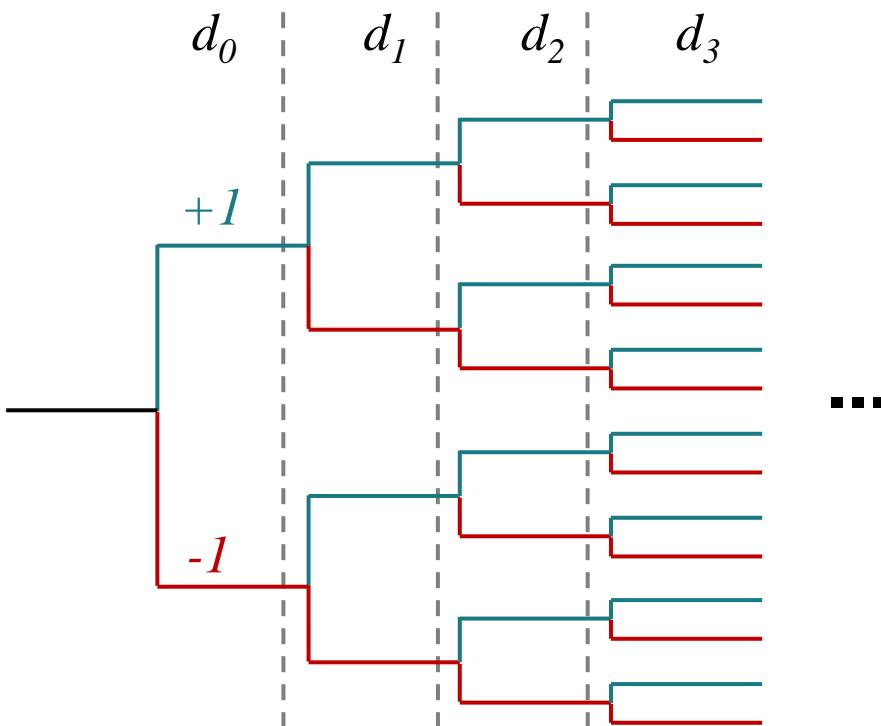
↑  
all possible transmit  
sequences  $\mathbf{d}$  equally likely

If all possible transmit sequences  $\mathbf{d}$  are equally likely, MAP and ML detection are equivalent.

# Possible Transmit Sequences: Tree Representation

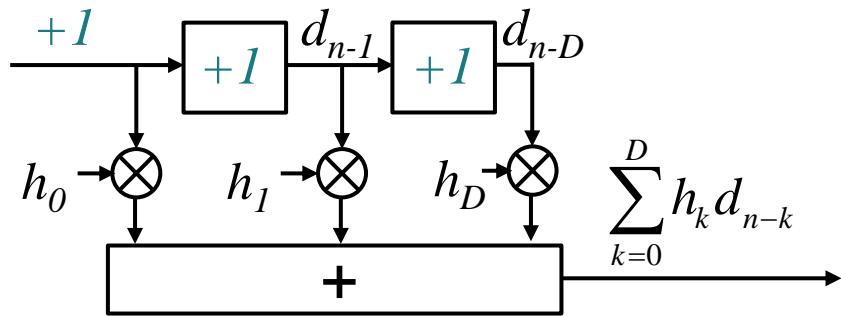


Binary data  $d_n \in \pm 1$ :



- $M$ -ary transmit symbols  $d_n$
  - Sequence of  $N$  transmit symbols
- $M^N$  possible transmit sequences
- Computational complexity of general MAP or ML detection increases exponentially with sequence length  $N$  !

# Trellis Diagram (1)

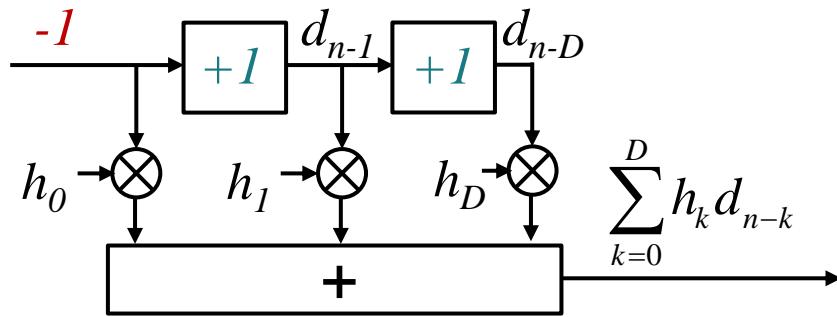


state

$d_{n-1}d_{n-2}$



# Trellis Diagram (2)



state

$d_{n-1}d_{n-2}$

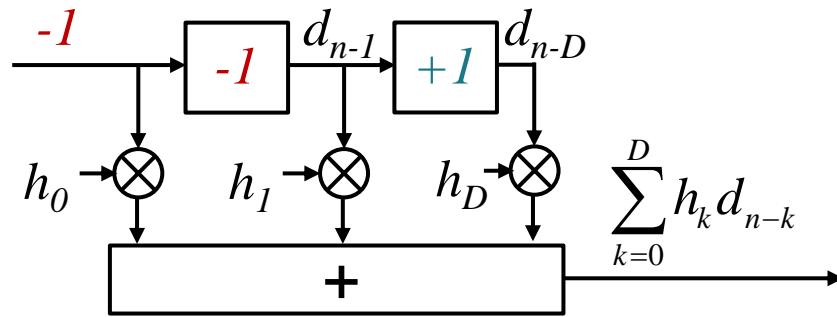


$+1-1$

$-1+1$

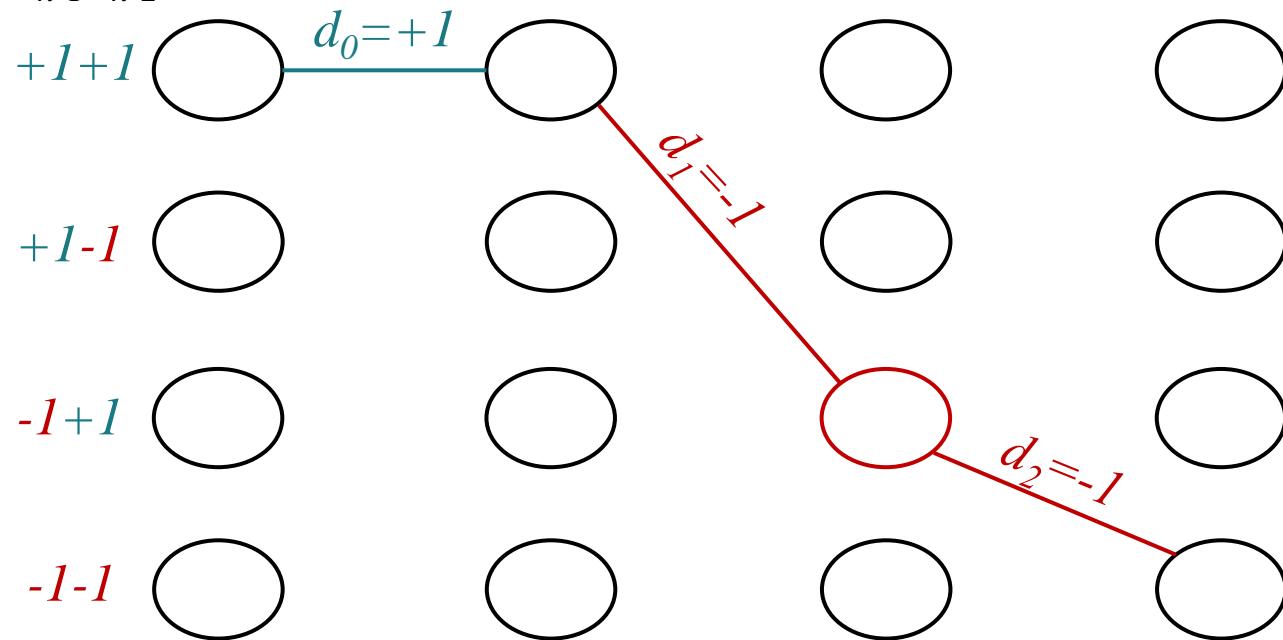
$-1-1$

# Trellis Diagram (3)

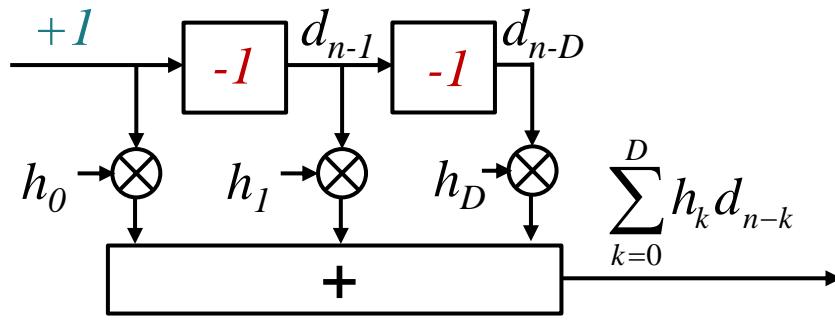


state

$d_{n-1}d_{n-2}$

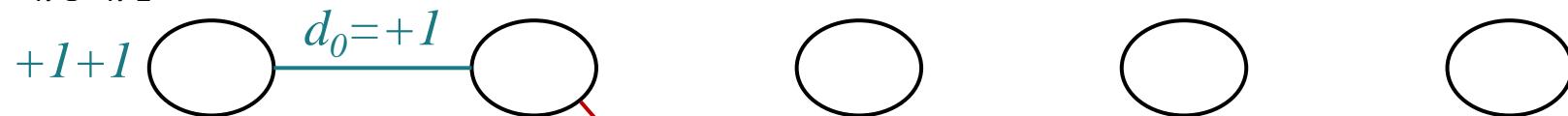


# Trellis Diagram (4)



state

$d_{n-1}d_{n-2}$

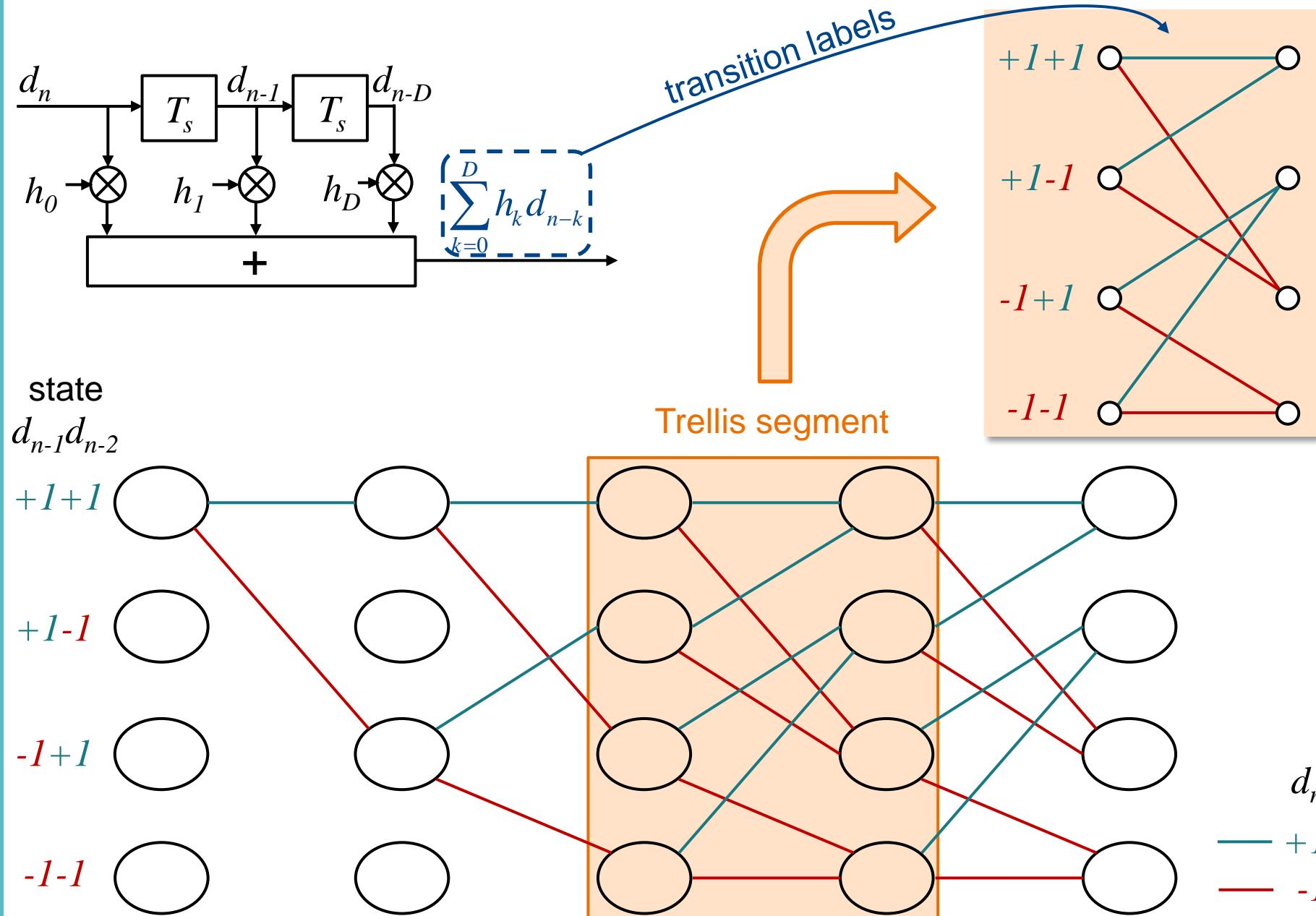


$+1-1$

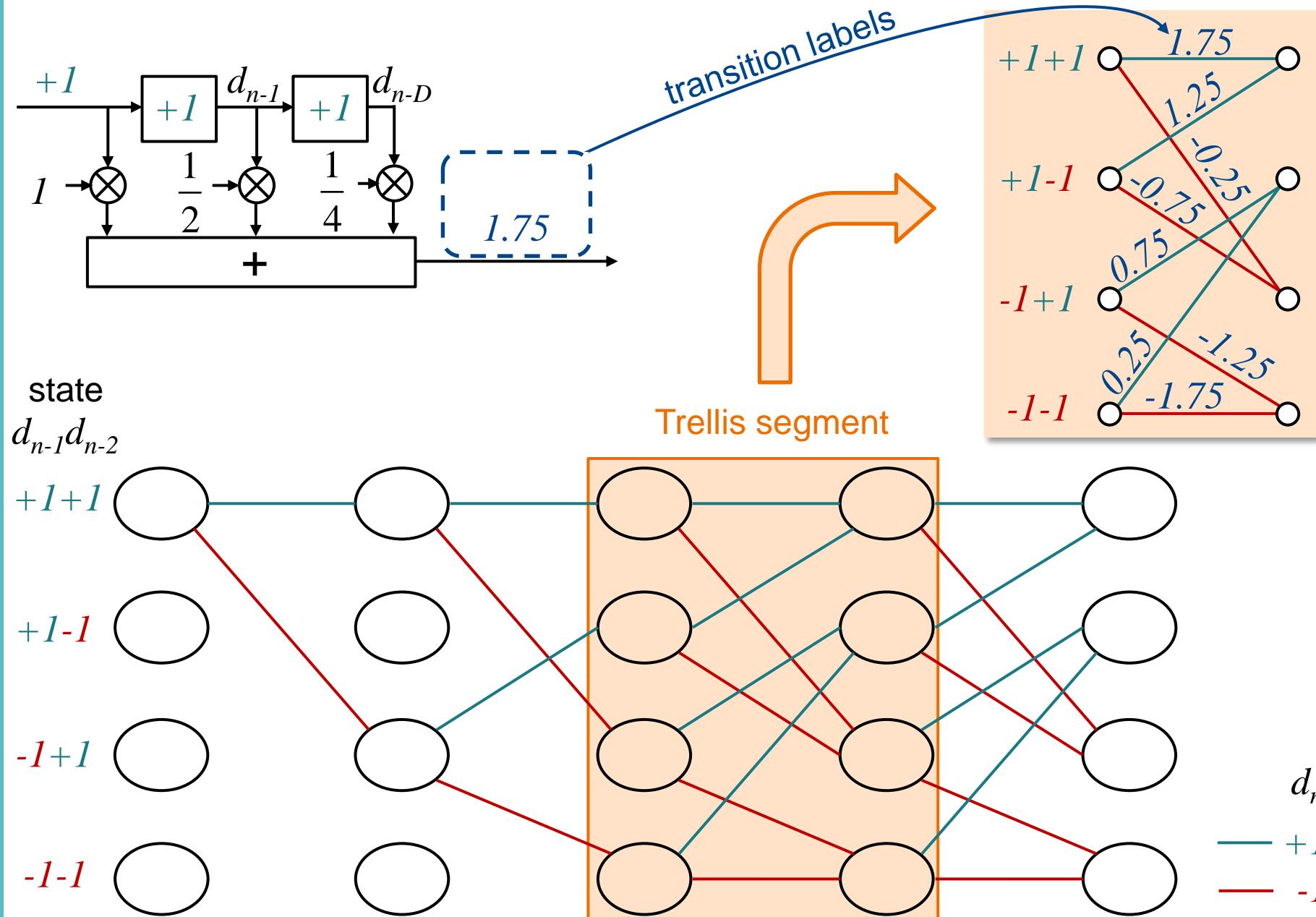
$-1+1$

$-1-1$

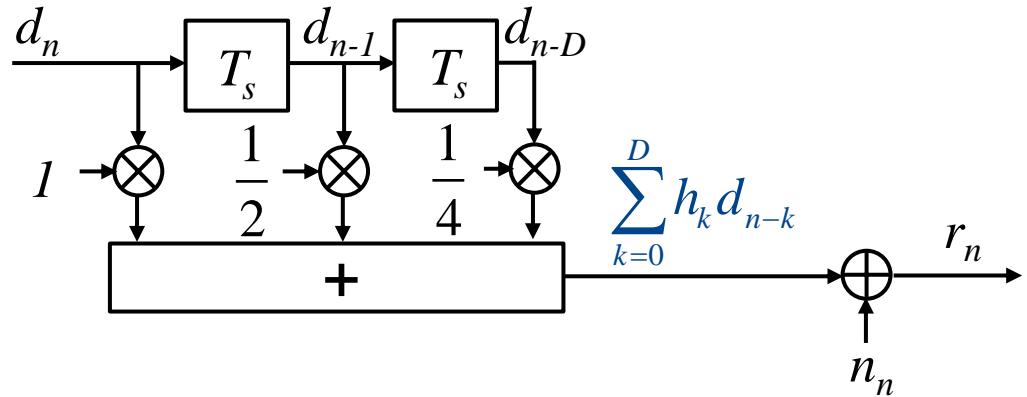
# Trellis Diagram (5)



# Trellis Diagram (6)

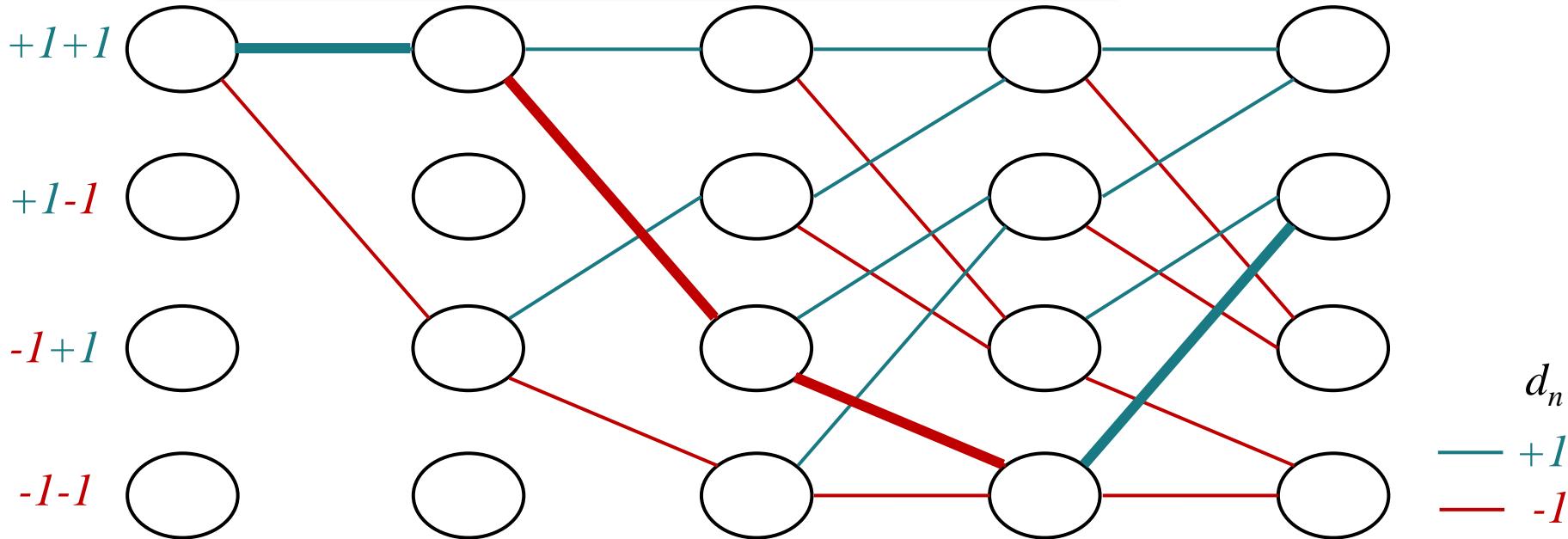
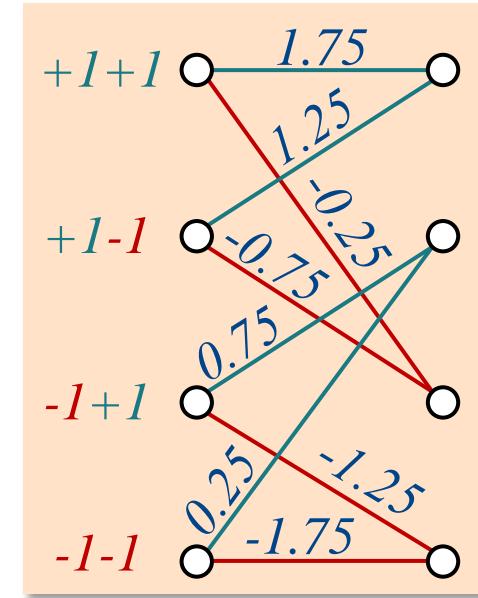


# Trellis Diagram (7)



state  
 $d_{n-1}d_{n-2}$

The detector needs to find the most likely path through the trellis given the received sequence.



# ML Equalizer: Viterbi Algorithm (1)

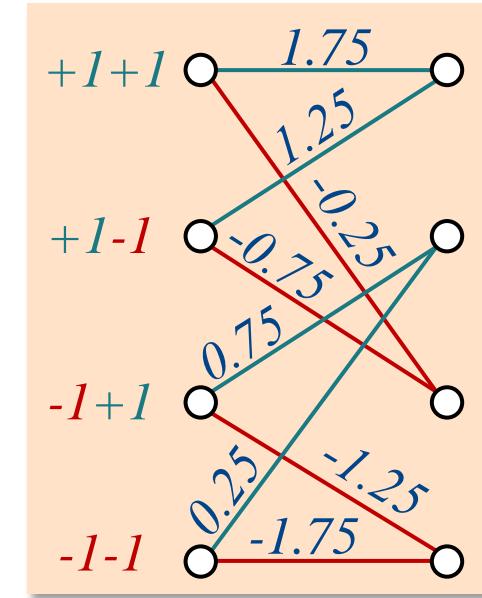
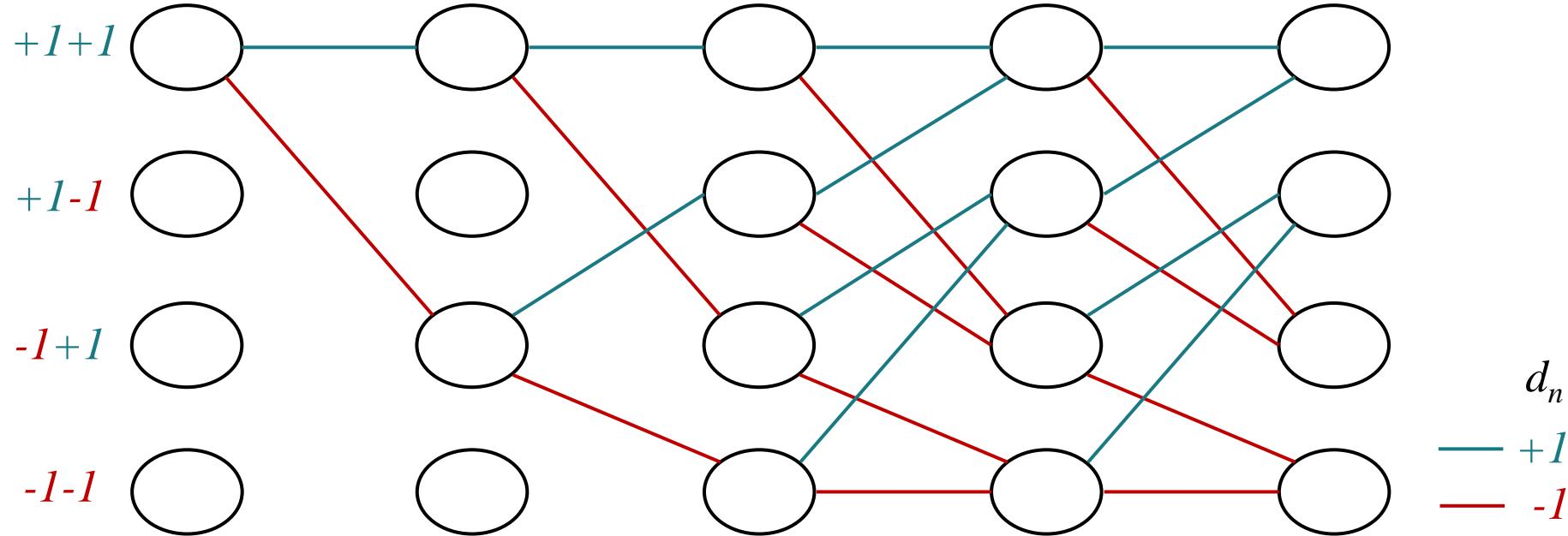
Metric of state  $s_j$  at time  $m$ :

$$\mu_{m+1}^{s_j} = \min_i \left\{ \mu_m^{s_i} + \Delta\mu_m^{s_i \rightarrow s_j} \right\}$$

$$\Delta\mu_m^{s_i \rightarrow s_j} = \left| \mathbf{r}_m - \sum_{k=0}^D h_k d_{m-k} \right|^2$$

Metric increment for transition from state  $s_i$  to state  $s_j$  at time  $m$ .

state  
 $d_{n-1}d_{n-2}$



# ML Equalizer: Viterbi Algorithm (2)

Metric of state  $s_j$  at time  $m$ :

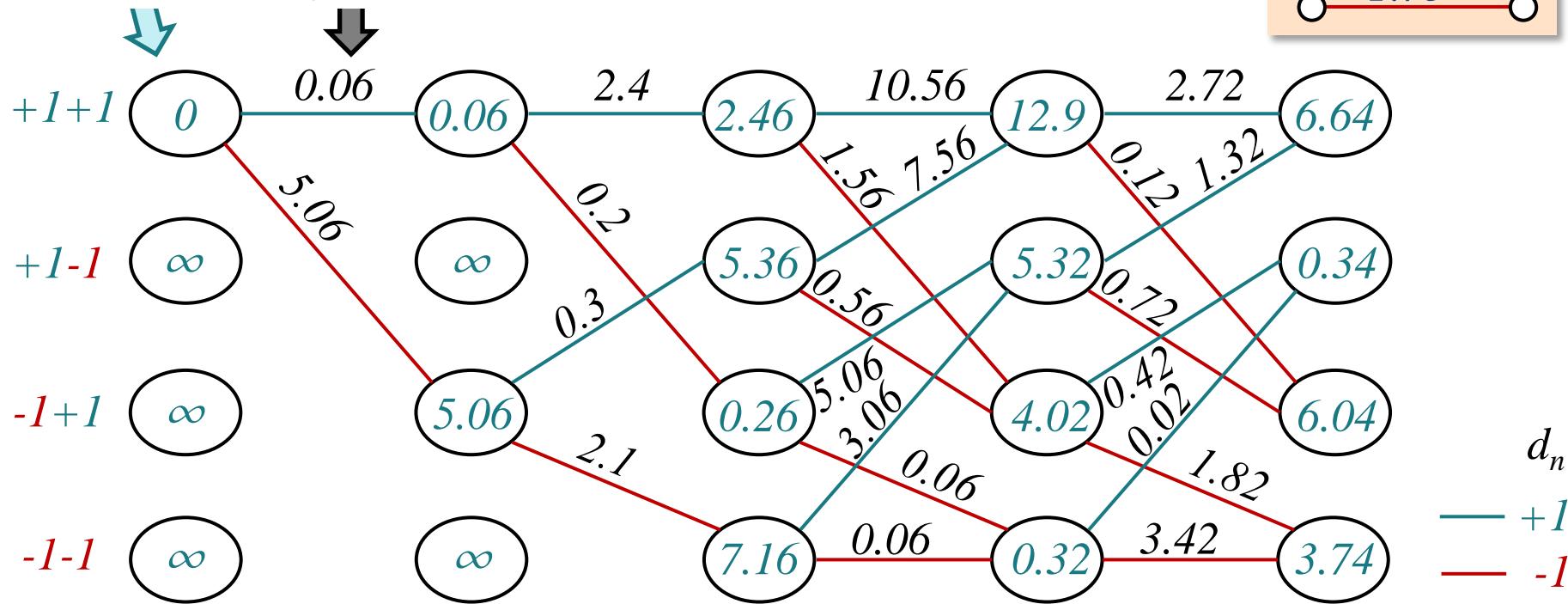
$$\mu_{m+1}^{s_j} = \min_i \left\{ \mu_m^{s_i} + \Delta\mu_m^{s_i \rightarrow s_j} \right\}$$

initialize metrics

$$\Delta\mu_m^{s_i \rightarrow s_j} = \left| \mathbf{r}_m - \sum_{k=0}^D h_k d_{m-k} \right|^2$$

determine metric increments

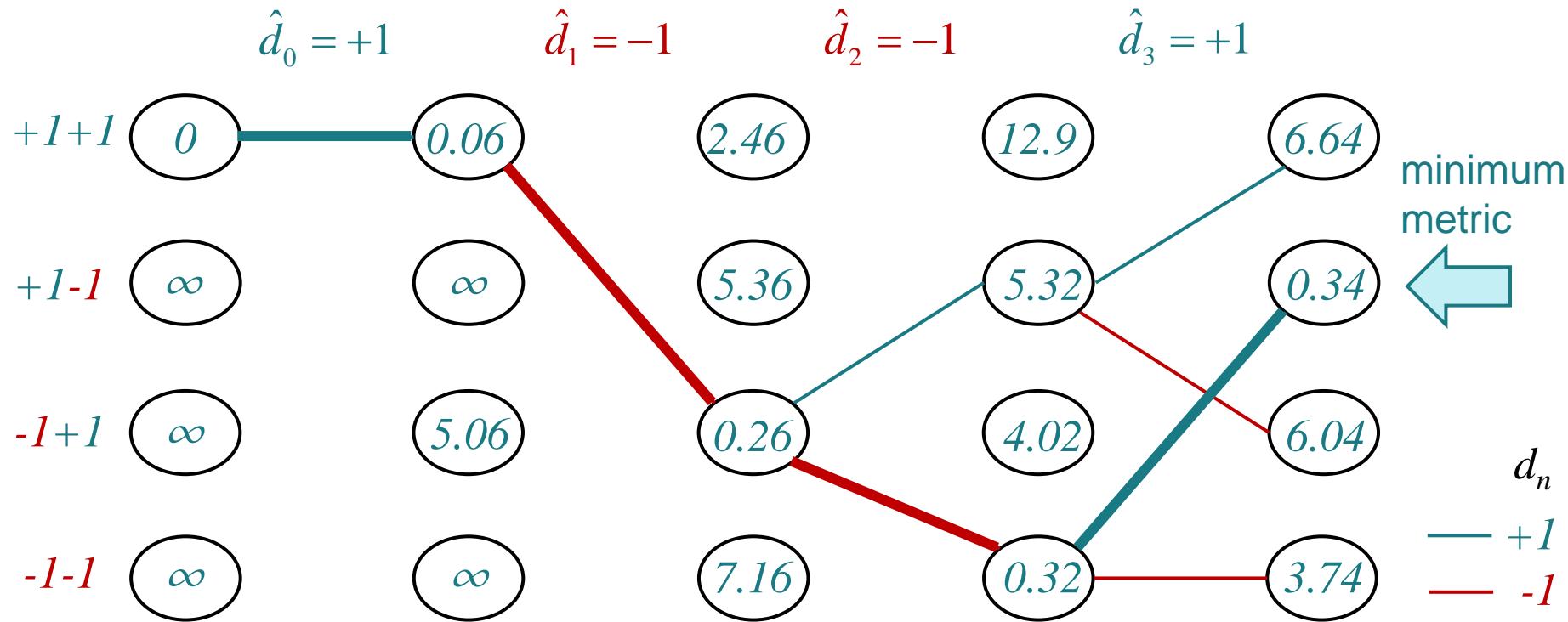
received sequence:  $r_0=2$      $r_1=0.2$      $r_2=-1.5$      $r_3=0.1$



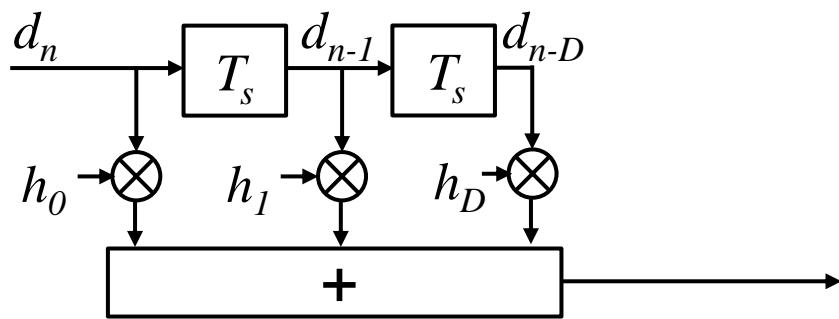
# ML Equalizer: Viterbi Algorithm (3)

- Find most likely path:
  - Find path with minimum metric.
- Trace back most likely path in order to detect the data sequence.

Detected sequence:



# Computational Complexity of the Viterbi Equalizer



$M$ -ary modulation

Channel memory:  $D$

Number of data symbols:  $N$

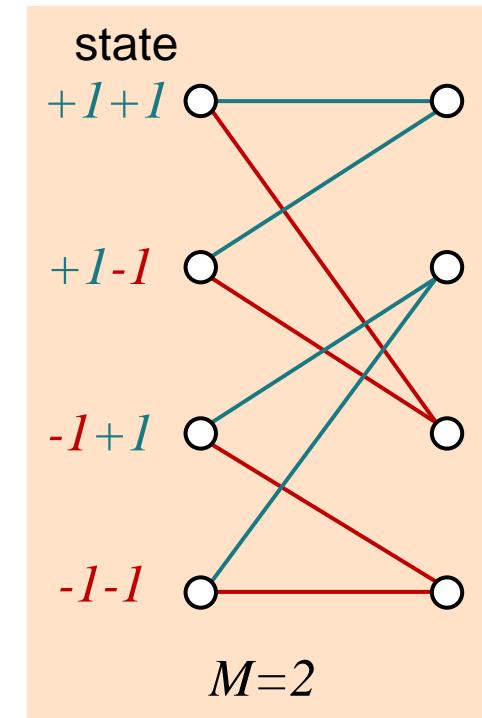


Number of trellis states:  $M^D$

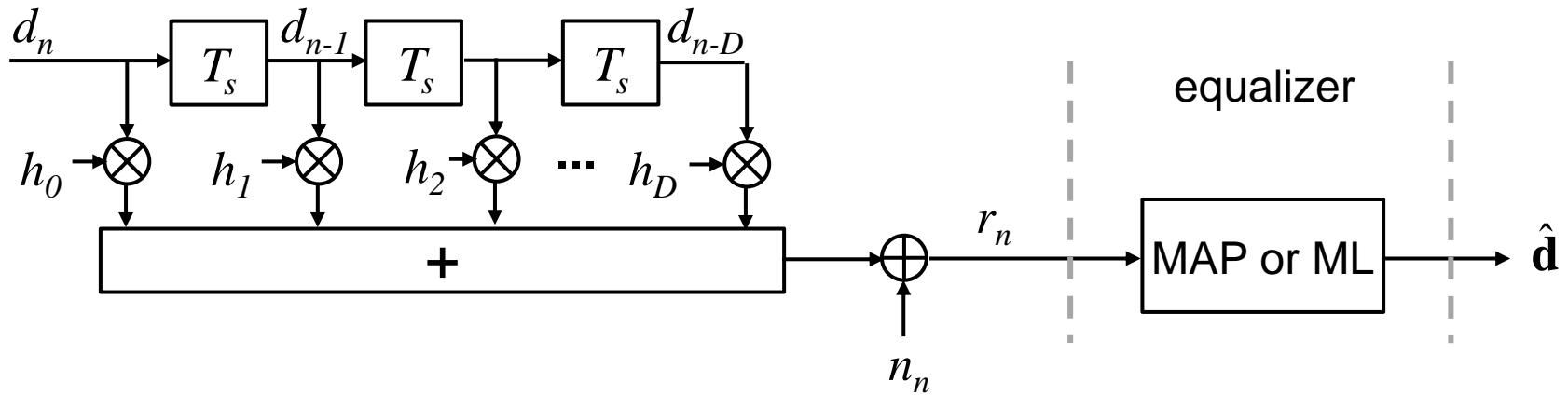
- Complexity grows exponentially with channel memory  $D$ .
- **Prohibitive complexity for long channel impulse response and/or high modulation order.**

Number of trellis segments:  $N$

- Complexity grows only linearly with data sequence length  $N$ .
- **Major advantage of Viterbi algorithm.**



# Viterbi Algorithm: Derivation of Metric (1)



Maximum a-posteriori (MAP) sequence estimation:

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \left\{ P(\mathbf{r}|\mathbf{d}) P(\mathbf{d}) \right\} = \arg \max_{\mathbf{d}} \left\{ \prod_{n=0}^{N-1} f_{R|D}(r_n | d_n) P(d_n) \right\}$$

$$P(\mathbf{r}|\mathbf{d}) = \prod_{n=0}^{N-1} f_{R|D}(r_n | d_n) \text{ for discrete memoryless channel}$$

$$P(\mathbf{d}) = \prod_{n=0}^{N-1} P(d_n) \text{ for statistically independent transmit symbols}$$

# Viterbi Algorithm: Derivation of Metric (2)

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \left\{ \prod_{n=0}^{N-1} f_{R|D}(r_n | d_n) P(d_n) \right\} = \arg \max_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} \left[ \log f_{R|D}(r_n | d_n) + \log P(d_n) \right] \right\}$$

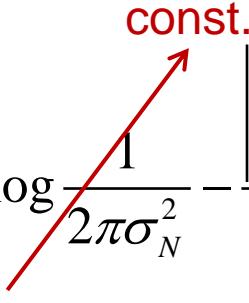
$$r_n = \sum_{k=0}^D h_k d_{n-k} + n_n$$

$$f_{R|D}(r_n | d_n) = \frac{1}{2\pi\sigma_N^2} e^{-\frac{|r_n - \sum_{k=0}^D h_k d_{n-k}|^2}{2\sigma_N^2}}$$

for AWGN

$$\log f_{R|D}(r_n | d_n) = \log \frac{1}{2\pi\sigma_N^2} - \frac{|r_n - \sum_{k=0}^D h_k d_{n-k}|^2}{2\sigma_N^2}$$

$$\hat{\mathbf{d}} = \arg \max_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} \left[ \log \frac{1}{2\pi\sigma_N^2} - \frac{|r_n - \sum_{k=0}^D h_k d_{n-k}|^2}{2\sigma_N^2} + \log P(d_n) \right] \right\}$$


**const.**

# Viterbi Algorithm: Derivation of Metric (3)

$$\hat{\mathbf{d}} = \arg \min_{\mathbf{d}} \left\{ \sum_{n=0}^{N-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] \right\}$$

$$= \arg \min_{\mathbf{d}} \left\{ \sum_{n=0}^{m-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] + \Delta \mu_m^{s_i \rightarrow s_j} + \sum_{n=m+1}^{N-1} \left[ \frac{\left| r_n - \sum_{k=0}^D h_k d_{n-k} \right|^2}{2\sigma_N^2} - \log P(d_n) \right] \right\}$$

$\mu_m^{s_j}$

# Viterbi Algorithm: Derivation of Metric (4)

Metric for state  $s_j$  at time  $m+1$ :

$$\mu_{m+1}^{s_j} = \min_i \left\{ \mu_m^{s_i} + \Delta\mu_m^{s_i \rightarrow s_j} \right\}$$

Metric increment for MAP equalizer:

$$\Delta\mu_m^{s_i \rightarrow s_j} = \frac{\left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2}{2\sigma_N^2} - \log P(d_m)$$

Metric increment for ML equalizer:

$$\Delta\mu_m^{s_i \rightarrow s_j} = \left| r_m - \sum_{k=0}^D h_k d_{m-k} \right|^2$$

# Equalizers

Approach	Type		Algorithms	Properties
signal processing based	linear	sample-spaced	<ul style="list-style-type: none"> <li>zero forcing (ZF)</li> <li>MMSE</li> </ul>	+ computationally efficient - noise enhancement
		fractionally spaced		
non-linear			decision feedback equalizer (DFE)	+ no noise enhancement - error propagation
probability based	<ul style="list-style-type: none"> <li>maximum likelihood (ML) sequence estimation</li> <li>maximum a-posteriori (MAP) sequence estimation</li> </ul>		Viterbi algorithm	+ optimum in terms of sequence error probability - huge computational complexity for long channel impulse response and high modulation order