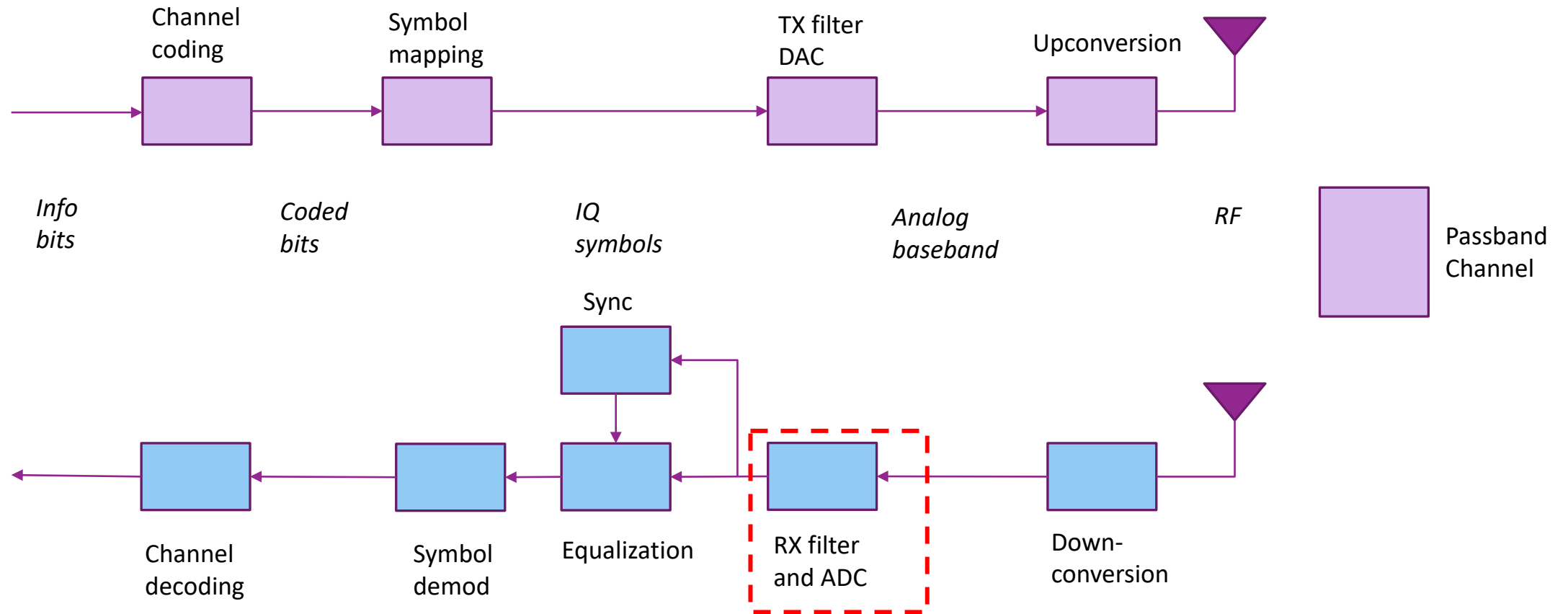


Unit 3: Receive Filtering

EL-GY 6013: DIGITAL COMMUNICATIONS

PROF. SUNDEEP RANGAN

This Unit



Learning Objectives

- ❑ Describe the steps in recovering symbols for a linearly modulated signal
 - Determine the matched filter response
- ❑ Determine MF for known gain and delay in the channel
- ❑ Compute the effective discrete-time channel given
 - Channel response, TX and RX filter
 - Time-domain or frequency-domain method
- ❑ Determine if there is ISI
- ❑ Compute the frequency response using digital RX filtering and downsampling
- ❑ Determine specifications on the digital and analog filters

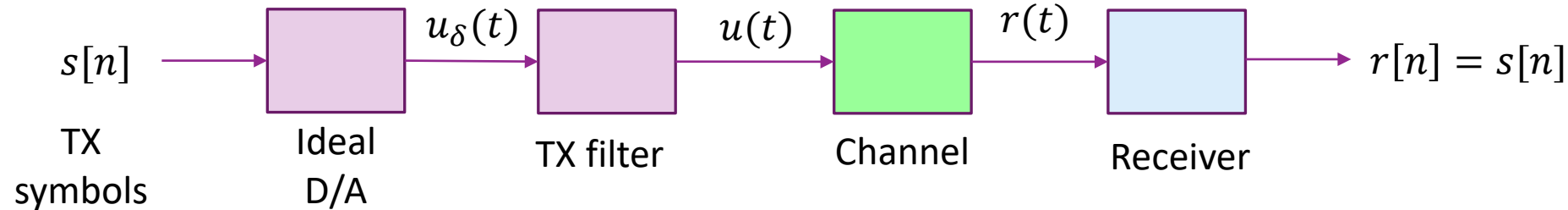
Outline



- Receiver filtering and sampling
 - Perfect reconstruction with orthonormal modulation
 - General channels: Time-domain analysis
 - General channels: Frequency-domain analysis
 - PSD Analysis
 - Practical RX filter design
 - Channel sounding



Receiver Problem



□ Transmit steps so far:

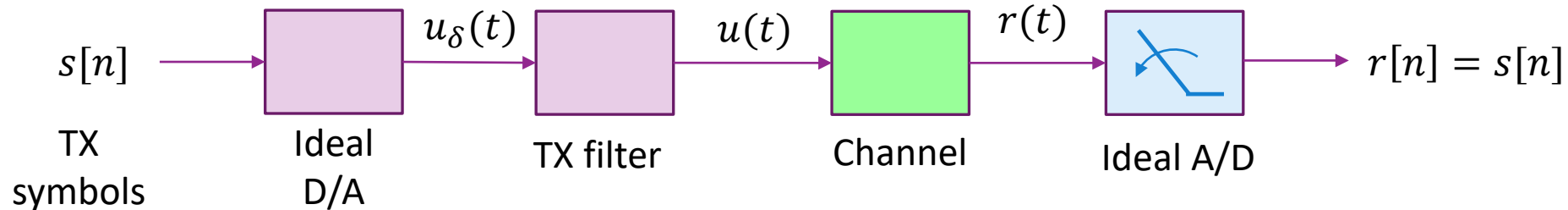
- Symbols $s[n]$
- Linearly modulate: $u(t) = \sum s[n]p_{tx}(t - nT)$
- Baseband equivalent channel $r(t) = h_{chan}(t) * u(t)$

□ Question at the receiver: Can we recover the transmitted symbols?

□ Want a mapping that input $r(t)$ to samples $r[n]$

□ Ideally $r[n] = s[n]$

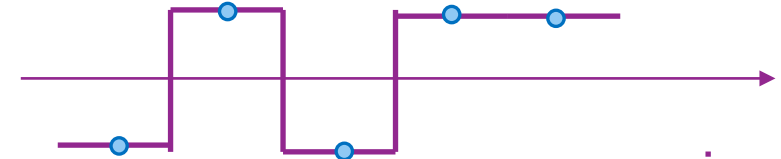
Simple Idea: Sampling



□ Take samples with an ideal A/D: $r[n] = r(nT)$

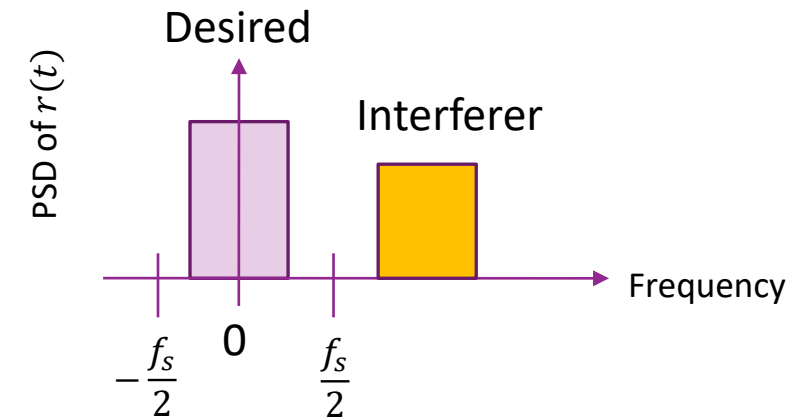
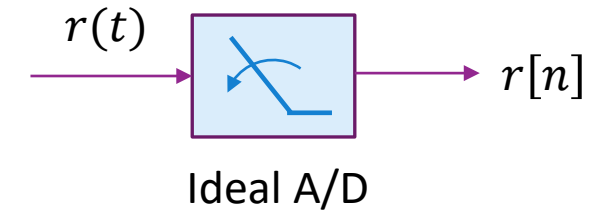
□ This could work: Example

- Suppose $p_{tx}(t) = \text{Rect}\left(\frac{t}{T}\right)$ (Ideal zero-order-hold D/A)
- $h_{chan}(t) = \delta(t)$ (no channel effect)
- Then: $r(t) = u(t) = \sum_n s[n] \text{Rect}\left(\frac{t-nT}{T}\right)$
- So, $r(nT) = s[n]$
- Hence, if we sample at exactly the right time, we can recover $s[n]$



Problems with Ideal Sampling

- ❑ Three problems in implementing an ideal A/D
- ❑ **Problem 1:** No circuit exactly samples at one instant
 - Most circuits integrate over some period
 - Ex: Charge fills a capacitor at the input to the A/D
- ❑ **Problem 2:** Out-of-band emissions (“blockers”)
 - The received signal may contain signals at neighboring frequencies
 - Ex: Transmissions in other wireless channels
 - The system may not have control over these
 - Without filtering, these will be aliased into $r[n]$
 -

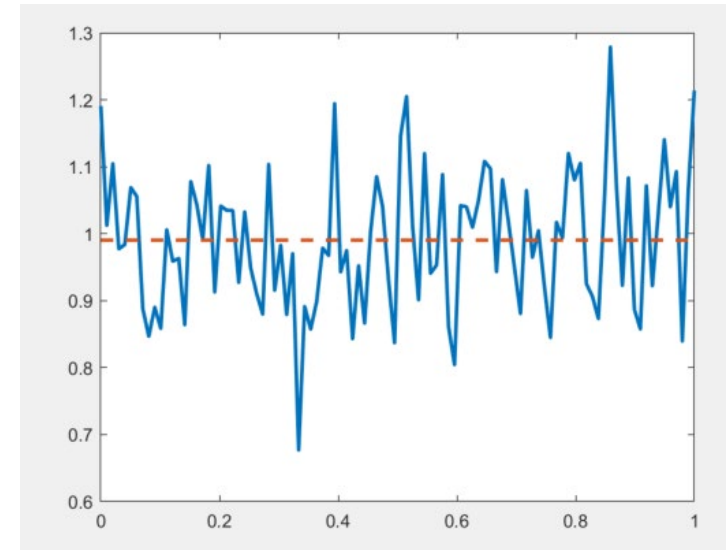
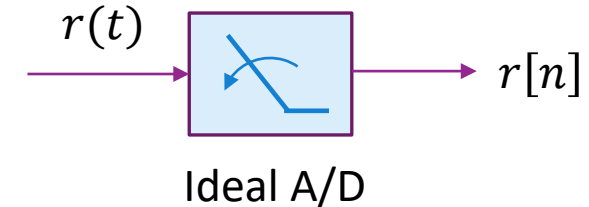


Problems with Ideal Sampling

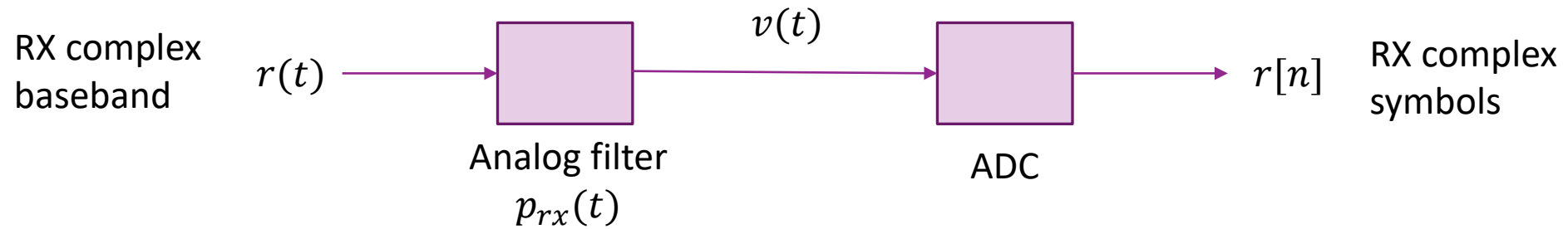
❑ Three problems in implementing an ideal A/D

❑ **Problem 3:** No noise filtering

- Suppose that in some symbol period:
$$r(t) = s[n] + w(t), \quad t \in [nT, (n+1)T)$$
- $r(t)$ = “desired signal” + “noise”.
- Noise $w(t)$ will appear in any sample.
- But suppose we average: $r_{avg}[n] = \frac{1}{T} \int_{nT}^{(n+1)T} r(t) dt = s[n] + v[n]$
- Effective noise is $v[n] = \frac{1}{T} \int_{nT}^{(n+1)T} w(t) dt$
- This will, in general, have a lower variance
- We will describe this more next unit




Two Step Receiver

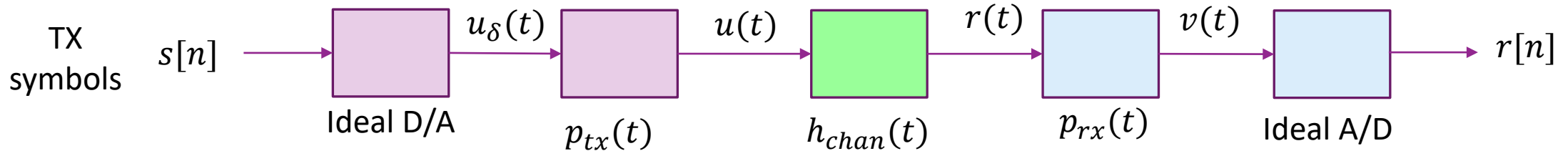


- ❑ Discussion motivates a two-step process
- ❑ Step 1: Receive filter: $v(t) = p_{rx}(t) * r(t)$
 - $p_{rx}(t)$ is the RX filter response
- ❑ Step 2: Sample $r[n] = v(nT)$
- ❑ Filter is useful to:
 - Model imperfections in the sampling
 - Filter out blockers. Anti-aliasing
 - Average out noise (more on this later)

Outline

- ☐ Receiver filtering and sampling
-  ☐ Perfect reconstruction with orthonormal modulation
- ☐ General channels: Time-domain analysis
- ☐ General channels: Frequency-domain analysis
- ☐ PSD Analysis
- ☐ Practical RX filter design
- ☐ Channel sounding

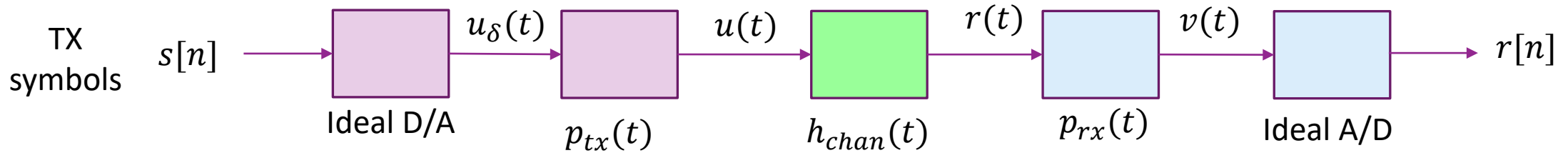
End-to-end TX and RX Chain so Far



□ Steps:

- Impulse D/A: $u_\delta(t) = \sum s[n]\delta(t - nT)$
- TX pulse shape: $u(t) = u_\delta(t) * p_{tx}(t) = \sum s[n]p_{tx}(t - nT)$
- Channel: $r(t) = u(t) * h_{chan}(t)$
- RX filter: $v(t) = r(t) * p_{rx}(t)$
- Sampling A/D: $r[n] = v(nT)$

Basic Questions



- ❑ Under what circumstances can we construct transmitted signals.
- ❑ That is, how do we select $p_{rx}(t)$ such that $r[n] = s[n]$?
- ❑ We first analyze this for a simple case:
 - Orthonormal pulse shapes
 - No channel impairments

Inner Products and Orthonormal Signals

□ Let $f(t), g(t)$ be two complex-valued signals

□ **Definition 1:** The **inner product** of f, g is:

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f^*(t)g(t)dt$$

◦ Here $f^*(t)$ = complex-conjugate of $f(t)$

□ **Definition 2:** We say $f(t), g(t)$ are **orthogonal** if $\langle f, g \rangle = 0$

◦ We will write this as $f \perp g$

□ **Definition 3:** The **signal energy** is $\|f\|^2 := \langle f, f \rangle = \int_{-\infty}^{\infty} |f(t)|^2 dt$

□ We will discuss this in much more detail in the next unit on signal spaces

Example Problem

□ Suppose $f(t) = a \text{Rect}\left(\frac{t}{T}\right)$, $g(t) = (b + ct) \text{Rect}\left(\frac{t}{T}\right)$

- Complex a, b, c with $a \neq 0$

□ Compute $\langle f, g \rangle$

□ When is $f \perp g$?

□ Solution:

- $\langle f, g \rangle = \int_{-\infty}^{\infty} f^*(t)g(t)dt = \int_{-T/2}^{T/2} a^*(b + ct)dt = a^*bT$
- Therefore $f \perp g = 0 \iff \langle f, g \rangle = a^*bT = 0$.
- Since $a, T \neq 0$, $f \perp g = 0 \iff b = 0$

Orthogonality in Frequency Domain

□ Sometimes it is more convenient to evaluate inner products in frequency domain

□ Parseval's Theorem: Let $f(t), g(t)$ be any two signals. Then:

$$\langle f, g \rangle = \int f^*(t)g(t)dt = \int F^*(f)G(f)df$$

□ This is useful whenever the Fourier transforms $F(f), G(f)$ are simple to work out.

Example

□ Suppose $f(t) = A \operatorname{sinc}\left(\frac{t}{T}\right)$, $g(t) = B \operatorname{sinc}\left(\frac{t-\tau}{T}\right)$

□ Compute $\langle f, g \rangle$. When are they orthogonal?

□ Solution:

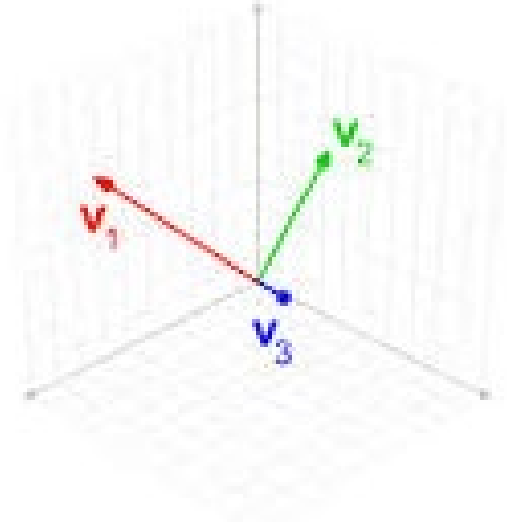
- Do this in frequency domain
- $F(f) = AT \operatorname{Rect}(fT)$, $G(f) = BT \operatorname{Rect}(fT) e^{-2\pi i f \tau}$
- From Parseval's Theorem: Let $f_0 = 1/(2T)$

$$\begin{aligned} \langle f, g \rangle &= \langle F, G \rangle = \int F^*(f) G(f) df = ABT^2 \int_{-f_0}^{f_0} e^{-2\pi i f \tau} df \\ &= \frac{ABT^2}{2\pi i \tau} [e^{2\pi i f_0 \tau} - e^{-2\pi i f_0 \tau}] = ABT \operatorname{sinc}(2f_0 \tau) = ABT \operatorname{sinc}\left(\frac{\tau}{T}\right) \end{aligned}$$

- $\langle f, g \rangle = 0$ when $\tau = kT$ for some integer k

Orthonormal Signals

- Let $\phi_n(t), n = 0, 1, \dots$ be a set of signals
 - This can be indexed from $n = -\infty$ to ∞ as well
- **Definition:** The set $\phi_n(\cdot)$ is **orthonormal** if:
 - $\|\phi_n\| = 1$ for all n (all signals have unit energy)
 - $\langle \phi_n, \phi_m \rangle = 0$ for all $n \neq m$ (different signals are orthogonal)
- This generalizes the concept of orthonormal vectors
- We will discuss orthonormal sets much more in signal space theory



Orthonormal Pulses and Matched Filtering

□ Consider the linear modulation: $u(t) = \sum_n s[n]p_{tx}(t - nT)$

□ **Definition 1:** We will say that the modulation is orthogonal if

$$\phi_n(t) = p_{tx}(t - nT), \quad n = \dots, -2, -1, 0, 1, 2, \dots$$

is an orthonormal set.

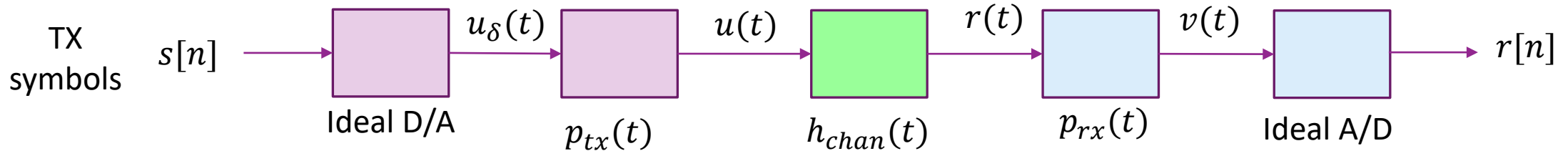
- We say $p_{tx}(t)$ is an orthonormal pulse at sample rate $\frac{1}{T}$.

□ **Definition 2:** Given any transmit pulse, $p_{tx}(t)$, the matched filter RX pulse is:

$$p_{rx}(t) = p_{tx}^*(-t)$$

- The pulse that is complex conjugate and flipped in time
- Note that if TX filter is causal, RX filter is anti-causal

Reconstruction With Orthonormal Pulses



□ **Theorem:** Suppose that:

- $p_{tx}(t)$ is an orthonormal pulse at sample rate $\frac{1}{T}$
- $h_{chan}(t) = \delta(t)$ (i.e., $r(t) = u(t)$ so there are no channel impairments)
- $p_{rx}(t) = p_{tx}^*(-t)$ (RX uses matched filter)

Then the receiver will exactly recover the TX samples in that $r[n] = s[n]$

□ **Theorem** answers our question. We can reconstruct the RX samples

- Under several assumptions

Proof of Reconstruction Theorem

- TX signal is: $u(t) = \sum_n s[n]p_{tx}(t - nT)$
- Since $h_{chan}(t) = \delta(t) \Rightarrow r(t) = h_{chan}(t) * u(t) = u(t) = \sum_n s[n]p_{tx}(t - nT)$
- RX filtered signal is: $v(t) = p_{rx}(t) * u(t) = \sum_n s[n](p_{rx} * p_{tx})(t - nT)$
- Sampling is: $r[m] = v(mT) = \sum_n s[n](p_{rx} * p_{tx})((m - n)T)$
- Now look at convolution:
$$(p_{rx} * p_{tx})(t) = \int p_{rx}(t - s)p_{tx}(s)ds = \int p_{tx}^*(s - t)p_{tx}(s)ds$$
- At $t = kT$: $(p_{rx} * p_{tx})(kT) = \int p_{tx}^*(s - kT)p_{tx}(s)ds$
- But $\phi_k(t) = p_{tx}(t - kT)$ is an orthonormal set
- So, $(p_{rx} * p_{tx})(kT) = \langle \phi_k, \phi_0 \rangle = \delta_k$
- Hence: $r[m] = v(mT) = \sum_n s[n]\delta_{m-n} = s[m]$

“Practical” Orthogonal Pulses

- There are two important “practical” orthonormal pulses
- Rectangles: $p_{tx}(t) = \frac{1}{\sqrt{T}} \text{Rect}\left(\frac{t}{T}\right)$
 - Orthonormal since $p_{tx}(t - nT)$ and $p_{tx}(t - mT)$ do not overlap when $n \neq m$
 - Scaling by $\frac{1}{\sqrt{T}}$ ensures they are normalized
 - This can be achieved (with some scaling) by a zero-order hold ADC
- Sinc pulses: $p_{tx}(t) = \frac{1}{\sqrt{T}} \text{Sinc}\left(\frac{t}{T}\right)$
 - Use similar frequency domain calculation as before to prove these are orthonormal
 - This would arise with ideal filtering at the TX and RX.
 - No filter is exactly ideal.
 - But, practical filters get quite close to this response.

In-Class Exercise

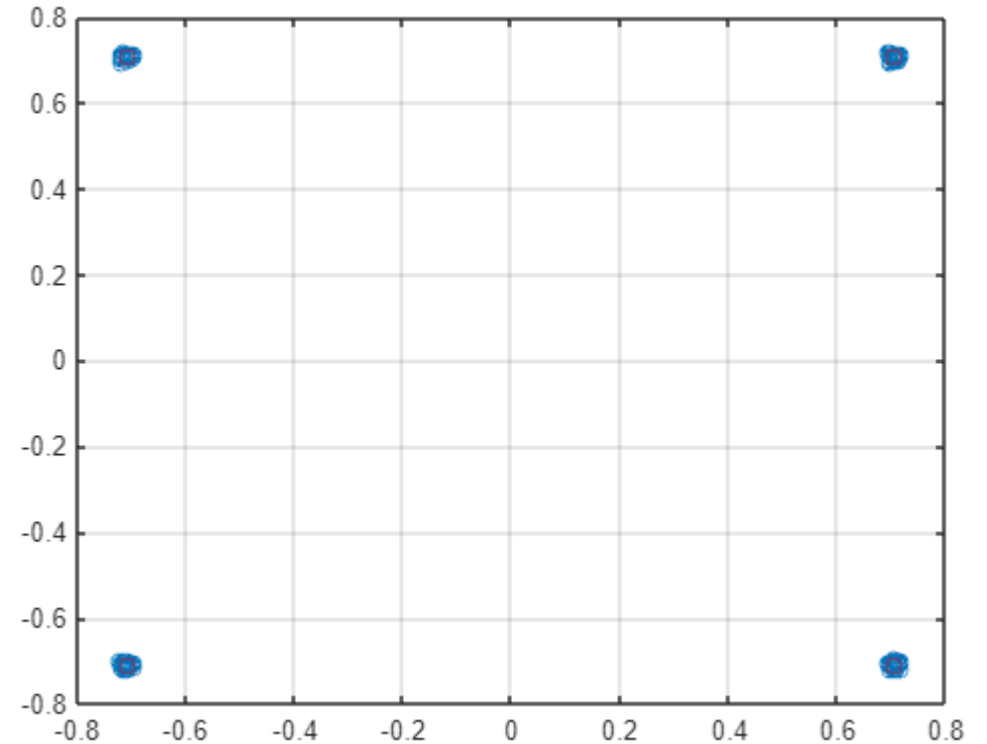
Simulating a Simple TX and RX Filter

We will first simulate a raised cosine TX and RX filter with no channel impairments.

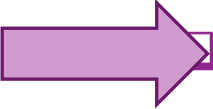
First, generate a set of QAM symbols with the following parameters.

```
nsym = 128;           % number of symbols
nbitsPerSym = 2;       % num bits / sym (2=QPSK)
nbits = nbitsPerSym*nsym; % num bits
fsym = 10e6;          % symbol rate
M = 2^nbitsPerSym;     % QAM order

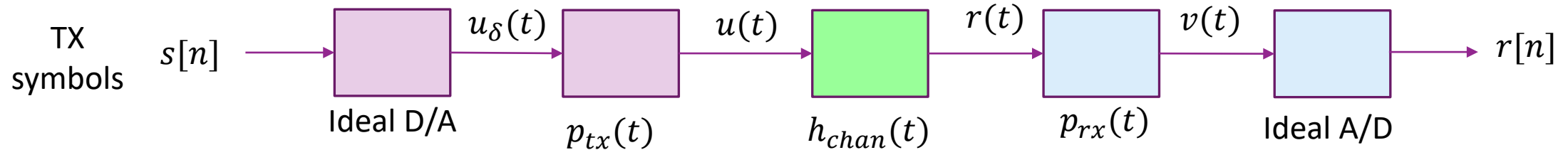
% TODO: Generate random bits and QPSK symbols
% h =
```



Outline

- ☐ Receiver filtering and sampling
- ☐ Perfect reconstruction with orthonormal modulation
-  ☐ General channels: Time-domain analysis
- ☐ General channels: Frequency-domain analysis
- ☐ PSD Analysis
- ☐ Practical RX filter design
- ☐ Channel sounding

Modeling the End-to-End System



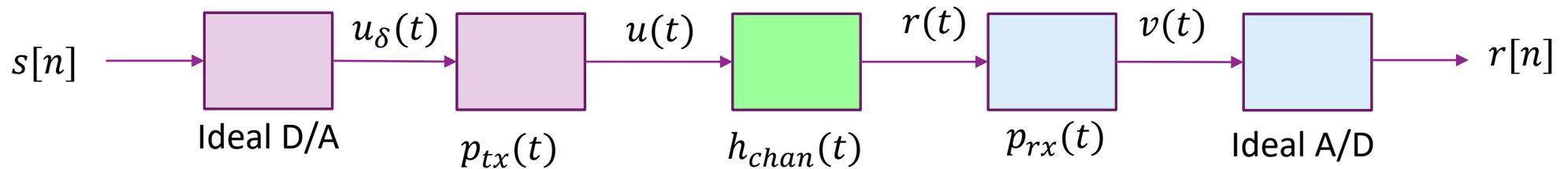
□ We have seen so far that when:

- Pulse shapes are matched
- Modulation is orthonormal
- No channel impairments

□ Then: $r[n] = s[n]$

□ What happens in the general case?

Key Result



- Define **channel impulse response with filtering** $g(t) = p_{tx}(t) * h_{chan}(t) * p_{rx}(t)$
 - Represents path from DAC output to ADC input

- **Theorem**: Mapping from $r[n]$ to $s[n]$ is LTI with impulse response is $h[n] = g(nT)$

$$r[n] = \sum_{k=-\infty}^{\infty} h[k]s[n-k]$$

- $h[n]$ is called the **effective discrete-time channel response**

Example 1: Rectangular Pulse

- Suppose that $p_{tx}(t) = p_{rx}(t) = \text{Rect}(t/T)$
- If channel is $r(t) = Gu(t - \tau)$, what is effective DT channel $h[n]$?
- Solution:
 - Impulse response is $h_{chan}(t) = G\delta(t - \tau)$
 - Impulse response from $u_\delta(t) \mapsto v(t)$ is $g(t) = p_{tx}(t) * p_{rx}(t) * h_{chan}(t) = GT \text{Tri}(t - \tau)$
 - Then $h[n] = g(nT)$

Example 1 Illustrated

- Channel response with filtering

$$g(t) = GT \operatorname{Tri}(t - \tau)$$

- Effective discrete-time channel:

$$h[n] = g(nT)$$

- Plot to right: $G = 1, T = 1$

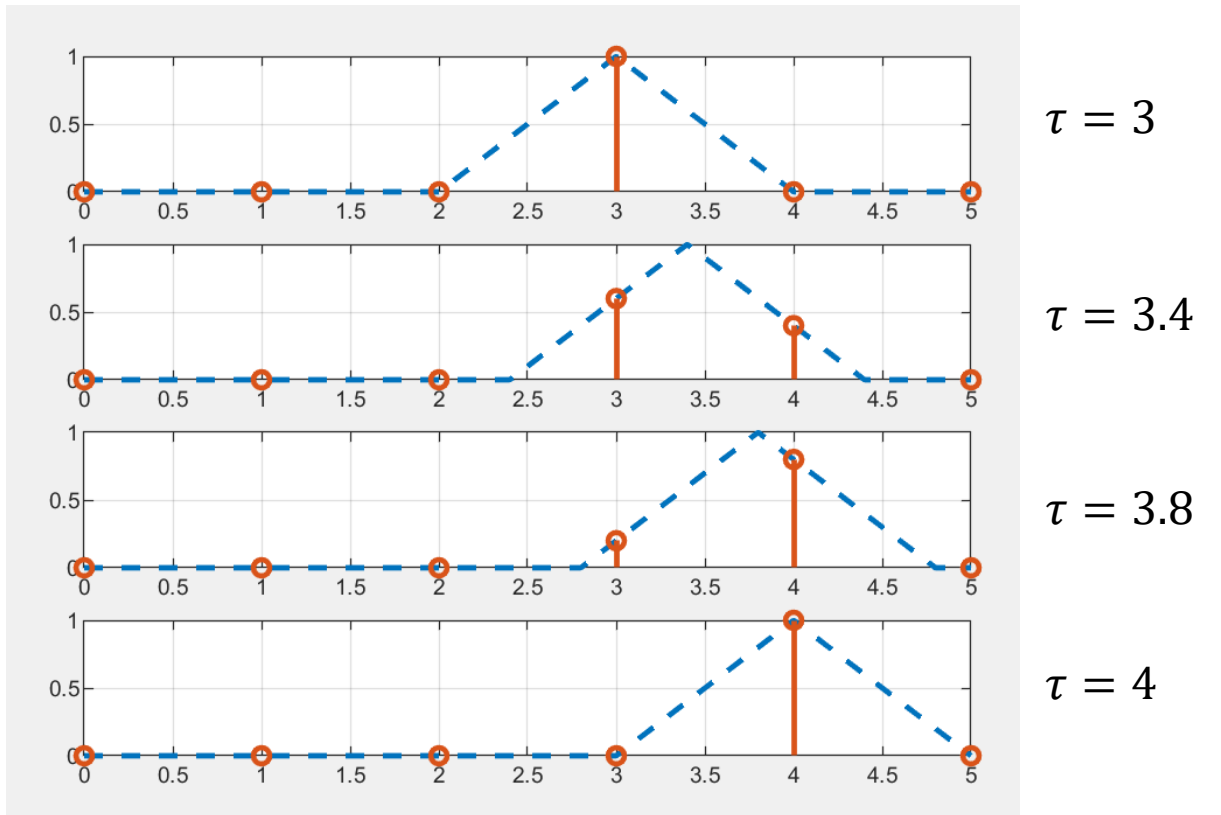
- $g(t)$: Blue dashed
- $h[n] = g(nT)$: Red stem

- When $\tau = kT$ is an integer:

- $h[n] = \delta_{n-k}$
- Single tap

- When $\tau \in (kT, (k+1)T)$

- $h[n]$ has two taps

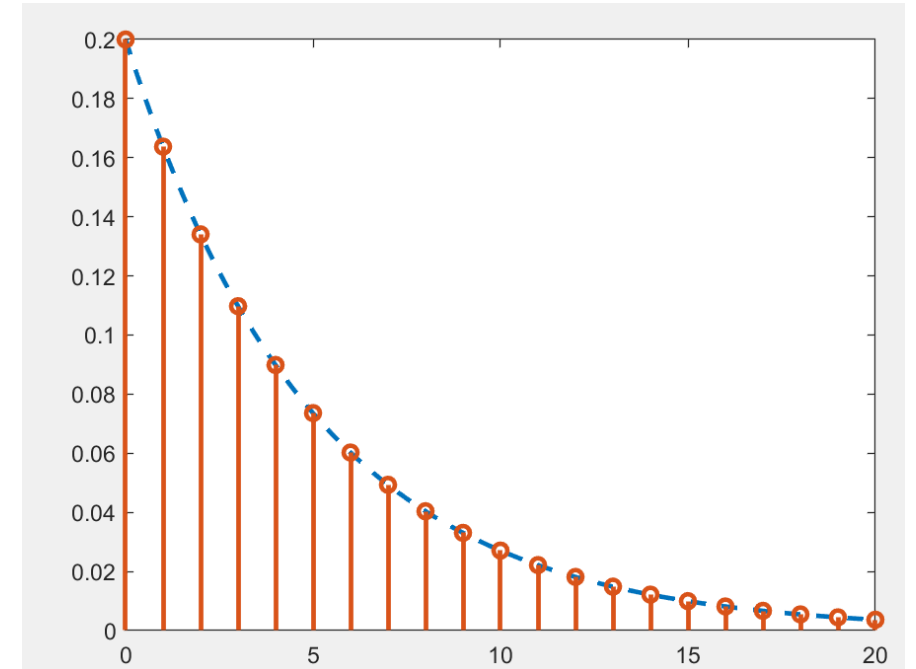


Example 2: Exponential

- Suppose that $p_{tx}(t) = \delta(t), p_{rx}(t) = T\delta(t)$
- Baseband channel is: $\frac{dr(t)}{dt} = \frac{\alpha}{T}(u(t) - r(t))$
- Find filtered channel response $g(t)$ and effective discrete-time response $h[n]$
- Solution:
 - Take Laplace transform of differential eqn: $R(s) = \frac{c}{s+c}U(s), c = \frac{\alpha}{T}$
 - Using inverse Laplace transform: $r(t) = h_{chan}(t) * u(t)$ where $h_{chan}(t) = ce^{-ct}1_{[0,\infty)}(t)$
 - Filtered channel response: $g(t) = p_{tx}(t) * p_{rx}(t) * h_{chan}(t) = cTe^{-ct}1_{[0,\infty)}(t)$
 - Discrete-time response: $h[n] = g(nT) = cTe^{-cnT}1_{\{n \geq 0\}} = \alpha e^{-\alpha n}1_{\{n \geq 0\}}$

Example 2: Exponential Illustrated

- Suppose that $p_{tx}(t) = \delta(t), p_{rx}(t) = T\delta(t)$
- Baseband channel is: $\frac{dr(t)}{dt} = \frac{\alpha}{T}(u(t) - r(t))$
- From previous slide:
 - Filtered channel response: $g(t) = cTe^{-ct}1_{[0,\infty)}(t)$
 - Discrete-time response: $h[n] = g(nT) = \alpha e^{-\alpha n}1_{\{n \geq 0\}}$



Inter-Symbol Interference

□ Effective discrete-time channel: $r[n] = \sum_{k=-\infty}^{\infty} h[k]s[n-k]$

□ **Inter-symbol interference:**

- Whenever $h[k] \neq 0$ for $k \neq 0$
- Other symbols interfere with one another

□ ISI occurs for many reasons:

- Multiple channel paths
- A single channel path that arrives between two sampling points

□ **Equalization:** Process to remove ISI

- Generally, find inverse filter $\hat{s}[n] = g[n] * r[n] \approx s[n]$
- Equalizing filter should roughly invert $h[n]$
- More on this later

In-Class Exercise

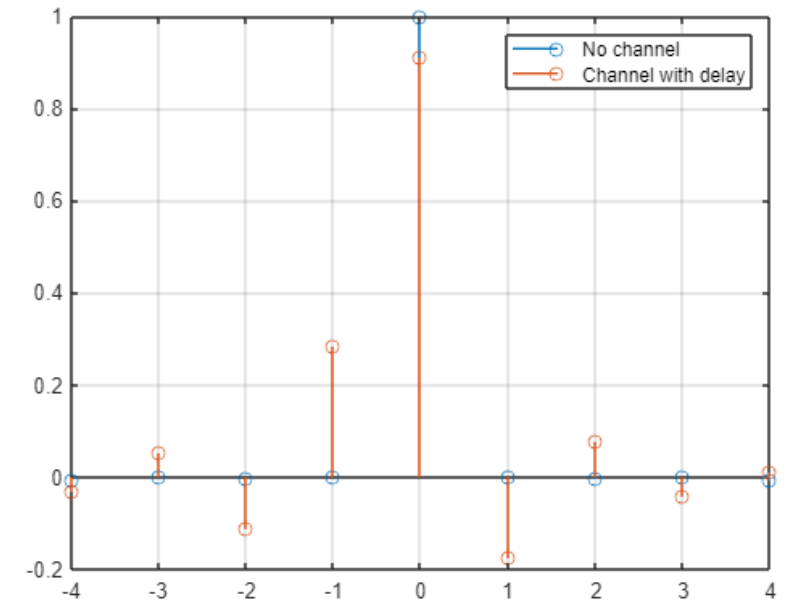
Computing an Effective Discrete-Time Channel

The baseband equivalent filter response is relative to some start time t_0 is:


$$h_{\text{dis}}[n] = g(t_0 + nT_{\text{sym}}), \quad g(t) = p_{\text{tx}}(t) * p_{\text{rx}}(t) * h_{\text{chan}}(t)$$

where T_{sym} is the symbol period. We can approximately compute by performing the convolutions in discrete time,

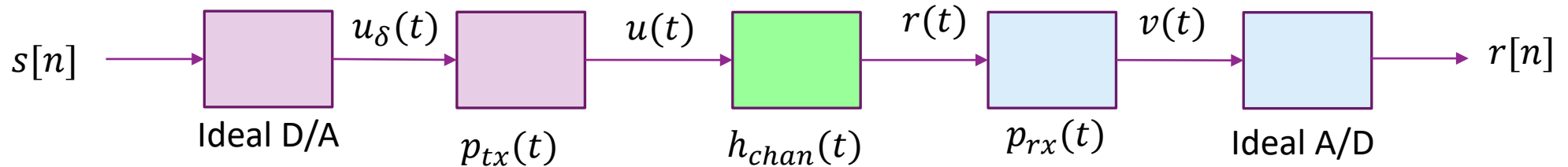
$$h_{\text{dis}}[n] = g[n * \text{nov} + 1], \quad g[k] = p_{\text{tx}}[k] * p_{\text{rx}}[k] * h_{\text{chan}}[k]$$



Outline

- ☐ Receiver filtering and sampling
- ☐ Perfect reconstruction with orthonormal modulation
- ☐ General channels: Time-domain analysis
-  ☐ General channels: Frequency-domain analysis
- ☐ PSD Analysis
- ☐ Practical RX filter design
- ☐ Channel sounding

Digital Channel Frequency Response



□ We saw that effective digital channel from $s[n] \mapsto u[n]$ is:

$$r[n] = \sum_k h[k]s[n-k], \quad h[k] = g(kT), \quad g(t) = p_{rx}(t) * h_{chan}(t) * p_{tx}(t)$$

□ Question: What is the frequency response?

$$R(\Omega) = H(\Omega)S(\Omega)$$

Frequency Response of DT Filter

□ Fact from signals and systems: If $h[n] = g(nT)$

$$H(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} G\left(\left(\frac{\Omega}{2\pi} + k\right)\frac{1}{T}\right)$$

□ Shifted copies of $G(f)$

□ Continuous frequency f mapped to DT frequency $\Omega = 2\pi f / f_s$, $f_s = 1/T$

□ Can obtain coefficients from inverse DTFT:

$$h[n] = \frac{1}{2\pi} \int_0^{2\pi} H(\Omega) e^{jn\Omega} d\Omega$$

Computing Effective DT Frequency Response Summary

□ Compute frequency response of channel with filtering $G(f) = P_{rx}(f)H_{chan}(f)P_{tx}(f)$

□ Effective DT channel response is:

$$H(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} G\left(\left(\frac{\Omega}{2\pi} + k\right)\frac{1}{T}\right)$$

- Scale $G(f)$ vertically by $1/T$
- Map continuous-time frequency f to $\Omega = 2\pi fT = \frac{2\pi f}{f_s}$
- Create shifted versions every 2π
- Shifted versions may overlap if there is aliasing

Bandlimited Channel

□ Suppose one of P_{rx} , P_{tx} or G bandlimited to $|f| < \frac{1}{2T}$,

□ Effective discrete-time channel reduces to:

$$H(\Omega) = \frac{1}{T} P_{rx} \left(\frac{\Omega}{2\pi T} \right) P_{tx} \left(\frac{\Omega}{2\pi T} \right) H_{chan} \left(\frac{\Omega}{2\pi T} \right)$$

for $|\Omega| < \pi$

□ Sampled version of the frequency response

Example: Sinc Pulses

□ Suppose that

- $p_{rx}(t) = p_{tx}(t) = \frac{1}{\sqrt{T}} \text{sinc}\left(\frac{t}{T}\right)$

□ Then,

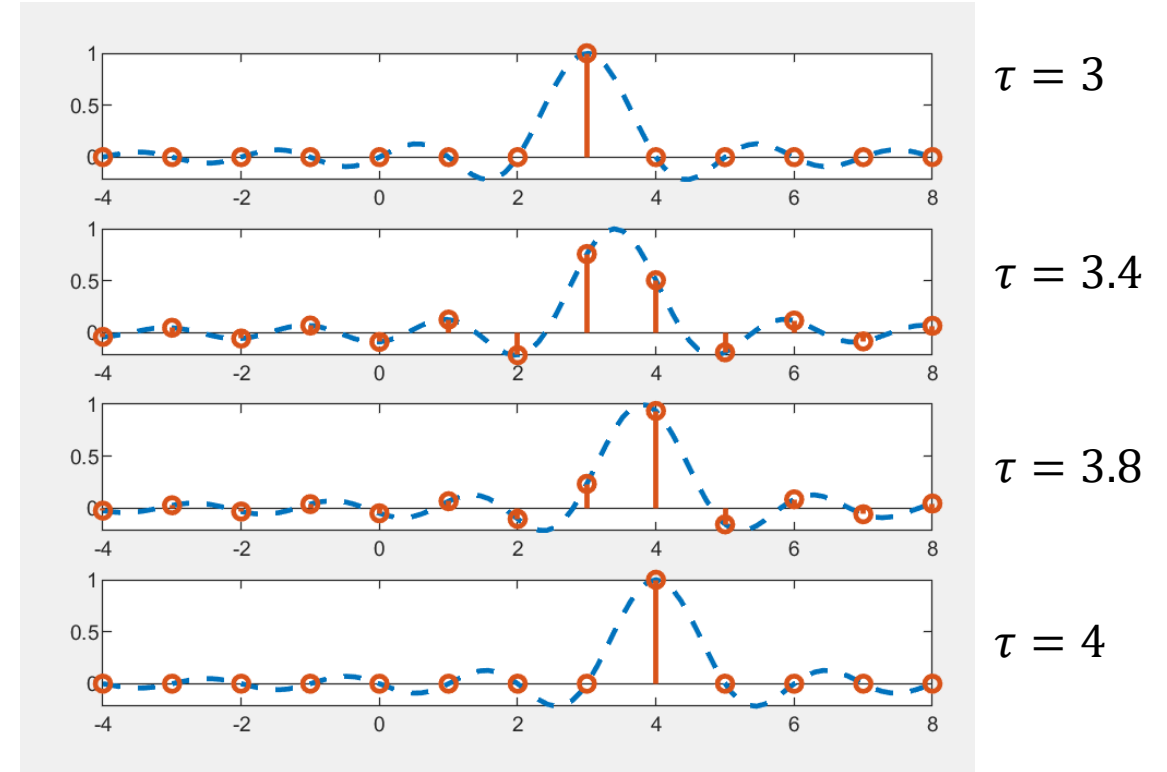
- $P_{rx}(f) = P_{tx}(f) = \sqrt{T} \text{Rect}(fT)$
- Hence $H(\Omega) = \frac{1}{T} P_{rx}\left(\frac{\Omega}{2\pi T}\right) P_{tx}\left(\frac{\Omega}{2\pi T}\right) H_{chan}\left(\frac{\Omega}{2\pi T}\right) = H_{chan}\left(\frac{\Omega}{2\pi T}\right)$
- Obtain sampled version of continuous-time channel

□ Suppose additionally $H_{chan}(f) = 1$ in $|f| \leq \frac{1}{2T}$.

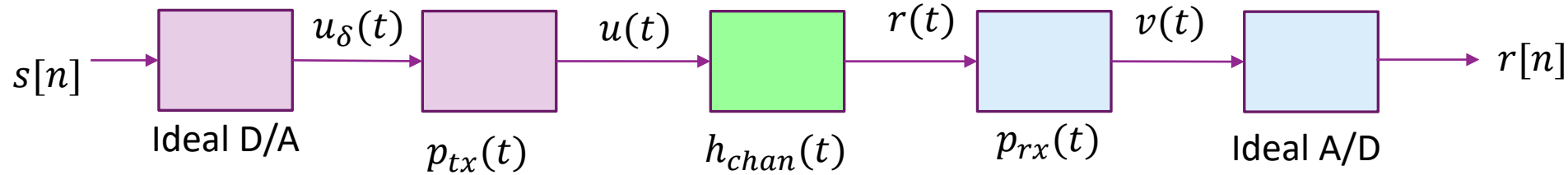
- Then $H(\Omega) = 1 \Rightarrow R(\Omega) = S(\Omega)$.
- Recover symbols exactly. No ISI

Example: Sinc Pulses with Delay

- Suppose that $p_{rx}(t) = p_{tx}(t) = \frac{1}{\sqrt{T}} \text{sinc}\left(\frac{t}{T}\right)$
- Channel has gain and delay: $r(t) = Gu(t - \tau)$
 - $H_{chan}(f) = Ge^{-2\pi jf\tau}$
- Then,
 - $P_{rx}(f) = P_{tx}(f) = \sqrt{T} \text{rect}(fT)$
 - $H(\Omega) = H_{chan}\left(\frac{\Omega}{2\pi T}\right) = Ge^{-j\Omega\tau/T}$
- Using DTFT table: $h[n] = G \text{sinc}\left(n + \frac{\tau}{T}\right)$
- Shown to right $G = 1, T = 1$



Numerically Computing the Channel



- ❑ For most real channels, we cannot analytically compute $h[n]$
 - Transmit and receive filters have complex frequency response
 - Channels can have many taps at arbitrary delays
- ❑ But we need $h[n]$ for proper simulation of communication systems
- ❑ Common solution: Approximately compute $h[n]$ numerically

Numeric Computation via Discretization

□ Baseband channel frequency response with filtering: $G(f) = P_{rx}(f)P_{tx}(f)H_{chan}(f)$

□ Effective discrete-time frequency response: $H(\Omega) = \frac{1}{T} G\left(\frac{\Omega}{2\pi T}\right)$

- Assume no aliasing.

□ Discrete-time impulse response: $h[n] = \frac{1}{2\pi} \int_0^{2\pi} H(\Omega) e^{i\Omega n} d\Omega$

□ Evaluate integral by discretization:

- Take N points

- $h[n] \approx \frac{1}{N} \sum_{k=-N/2+1}^{N/2} H(\Omega_k) e^{\frac{i2\pi kn}{N}}, \quad \Omega_k = \frac{2\pi k}{N}$

- Writing in terms of $G(f)$: $h[n] \approx \frac{1}{NT} \sum_{k=-N/2+1}^{N/2} G(f_k) e^{\frac{i2\pi kn}{N}}, \quad f_k = \frac{k}{TN}$

- Summation can be computed via an IFFT

Example: Two Path Channel

□ Sample rate = $\frac{1}{T} = 8(120)(1.024) = 983.04$ MHz

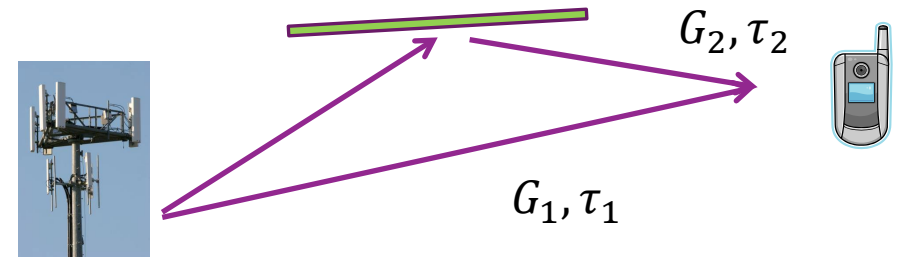
- Sample rate for an 8 channel 5G New Radio system at 120 sub-carrier spacing

□ $p_{rx}(t) = p_{tx}(t) = \frac{1}{\sqrt{T}} \text{Rect}\left(\frac{t}{T}\right)$ (zero-order hold ADCs)

□ $h_{chan}(t)$ has two paths:

- LOS path: Gain -10 dB, delay = 0
- NLOS path: Gain -15 dB, delay = 10 ns

□ In a wireless system, this would correspond to a path difference of $10(10)^{-9}3(10)^8 = 3$ m



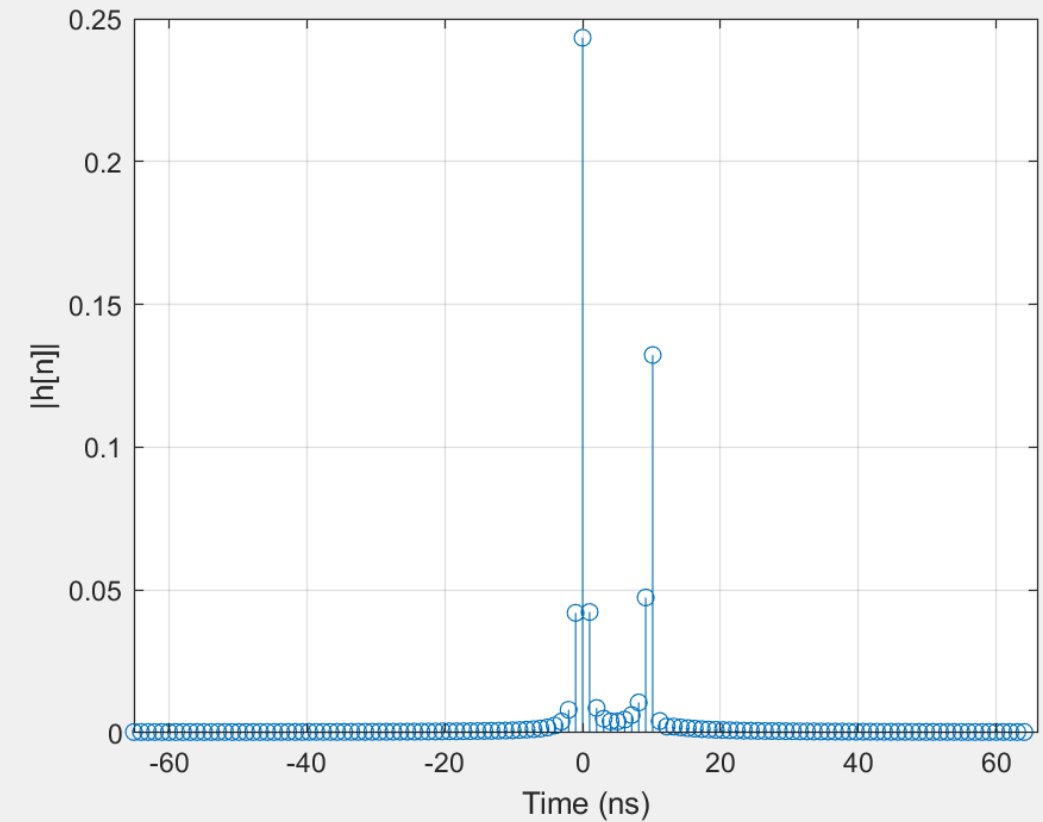
MATLAB Code

```
% Parameters
fsampMHz = 8*120*1.024;
Tsamp = 1/fsampMHz;

% Freq discretization points
npts = 128;
f = fsampMHz*(-npts/2+1:npts/2)'/npts;


% TX and RX filter
Prx = sqrt(Tsamp)*sinc(f*Tsamp);
Ptx = Prx;

% Channel
gaindB = [-10,-15];
dlyus = [0,0.01];
npath = length(gaindB);
Hchan = zeros(npts,1);
for ip = 1:npath
    Hchan = Hchan + 10^(0.05*gaindB(ip))*exp(-1i*2*pi*f*dlyus(ip));
end
```



```
% Compute discrete-time channel
G = Hchan.*Prx.*Ptx;
t = (-npts/2+1:npts/2)*Tsamp;
h = 1/Tsamp*ifft(G);
h = fftshift(h);
stem(t,abs(h));
```

Outline

- ☐ Receiver filtering and sampling
- ☐ Perfect reconstruction with orthonormal modulation
- ☐ General channels: Time-domain analysis
- ☐ General channels: Frequency-domain analysis
-  ☐ PSD Analysis
- ☐ Practical RX filter design
- ☐ Channel sounding

PSD of a Sampled Signal

□ Suppose that $y[n] = x(nT)$

□ If there is no aliasing: $Y(\Omega) = \frac{1}{T} X\left(\frac{\Omega}{2\pi T}\right)$

□ Now, look at auto-correlation:

$$R_y[m] = E(y[n]y^*[n-m]) = E(x(nT)x^*((n-m)T)) = R_x(mT)$$

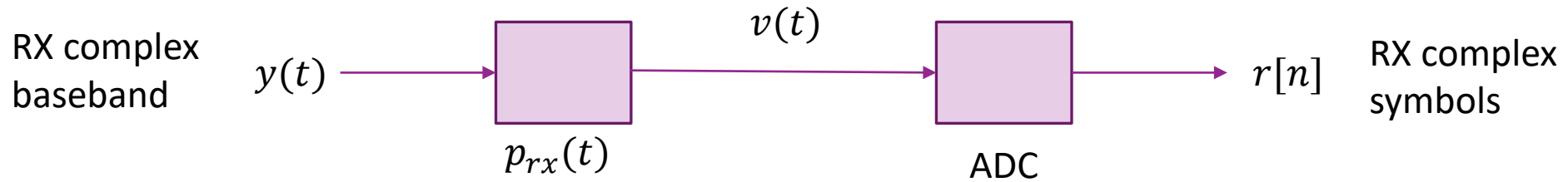
□ Hence, if there is no aliasing

$$S_y(\Omega) = \frac{1}{T} S_x\left(\frac{\Omega}{2\pi T}\right)$$

□ With aliasing:

$$S_y(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} S_x\left(\left(\frac{\Omega}{2\pi} + k\right)\frac{1}{T}\right)$$

PSD of the Receiver



□ We have $r[n] = v(nT)$

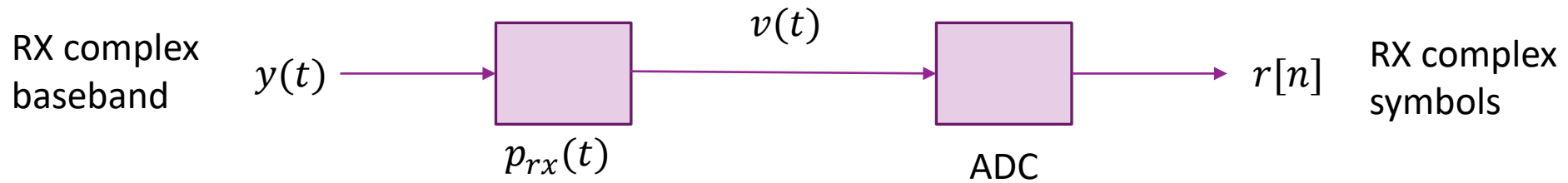
□ By sampling result, discrete-time PSD is:

$$S_r(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} S_v(f_k) = \frac{1}{T} \sum_{k=-\infty}^{\infty} |P_{rx}(f_k)|^2 S_y(f_k)$$

◦ $f_k = \left(\frac{\Omega}{2\pi} + k\right) \frac{1}{T}$

□ Without aliasing: $S_r(\Omega) = \frac{1}{T} \left| P_{rx} \left(\frac{\Omega}{2\pi T} \right) \right|^2 S_y \left(\frac{\Omega}{2\pi T} \right)$

Units in RX Chain: Time Domain



□ In time-domain: $v(t) = p_{rx}(t) * y(t)$, $r[n] = v(nT)$

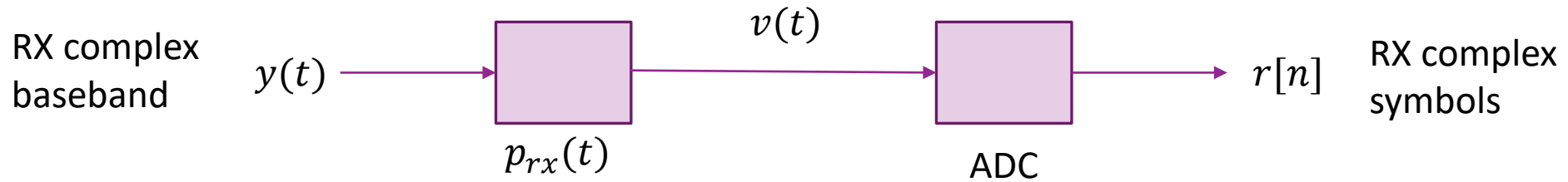
□ Units of $|y(t)|^2$ = power = energy / time

□ We take convention: $|p_{rx}(t)|^2 = 1 / (\text{time} * \text{sample})$

□ Then: $r[n] = \int r(t) p_{rx}(nT - t) dt = \sqrt{\frac{\text{Energy}}{\text{Time}}} \times \sqrt{\frac{1}{\text{Samples} \times \text{Time}}} \times \text{Time} = \sqrt{\frac{\text{Energy}}{\text{Sample}}}$

□ Hence: $|r[n]|^2 = \frac{\text{Energy}}{\text{Sample}}$

Units in RX Chain: PSD



$$\square S_r(\Omega) = \frac{1}{T} \left| P_{rx} \left(\frac{\Omega}{2\pi T} \right) \right|^2 S_y \left(\frac{\Omega}{2\pi T} \right)$$

$$\square \text{Units of } S_y(f) = \text{power/Hz} = \text{energy}$$

$$\square \text{Units of } |p_{rx}(t)|^2 = 1/(\text{samples} \times \text{time})$$

$$\square \text{Units of } |P_{rx}(f)|^2 = 1/(\text{samples} \times \text{Hz})$$

$$\square \text{Units of } S_r(\Omega) = \frac{1}{\text{Time}} \times \text{Energy} \times \frac{\text{Time}}{\text{Sample}} = \text{Energy per radian}$$

Sample problem (Solution on board)

Received PSD and filter as shown

Two components:

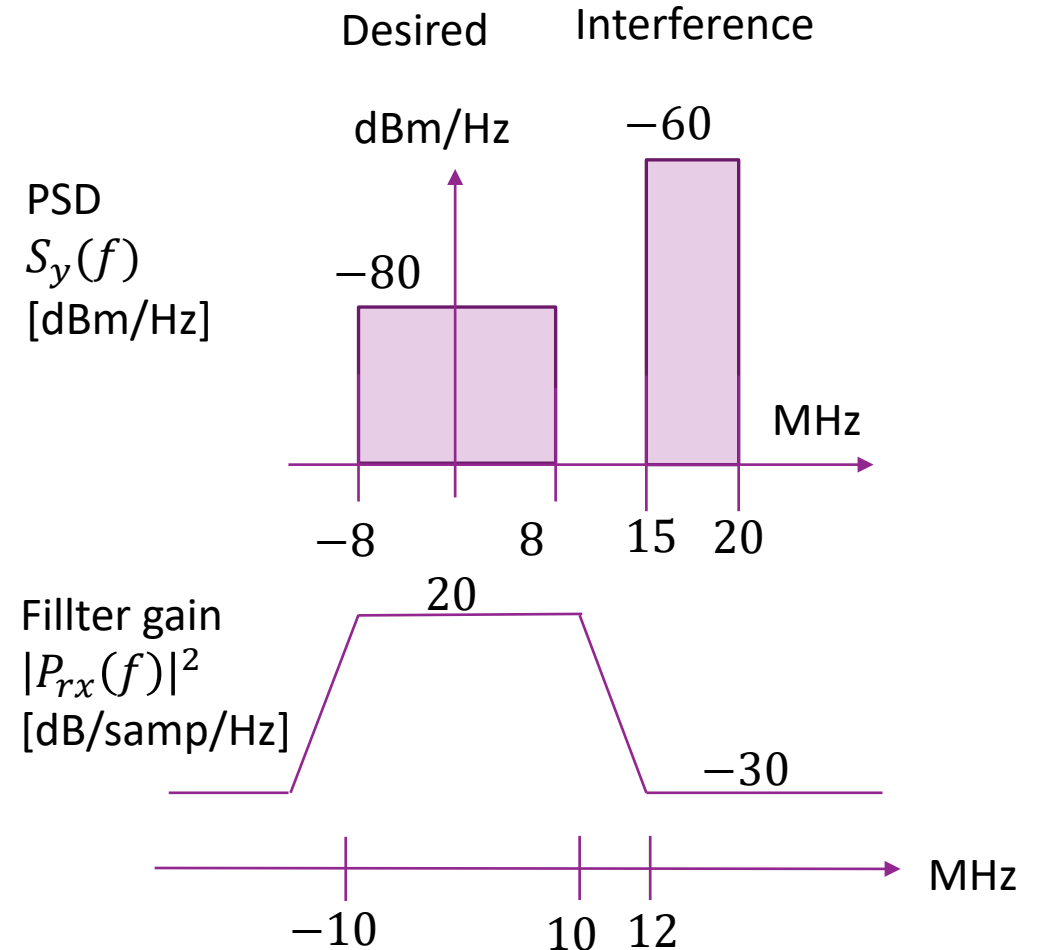
- Desired signal
- Nearby adjacent carrier signal

Assume they are uncorrelated


- Hence powers add.

Draw discrete-time PSD $S_r(\Omega)$

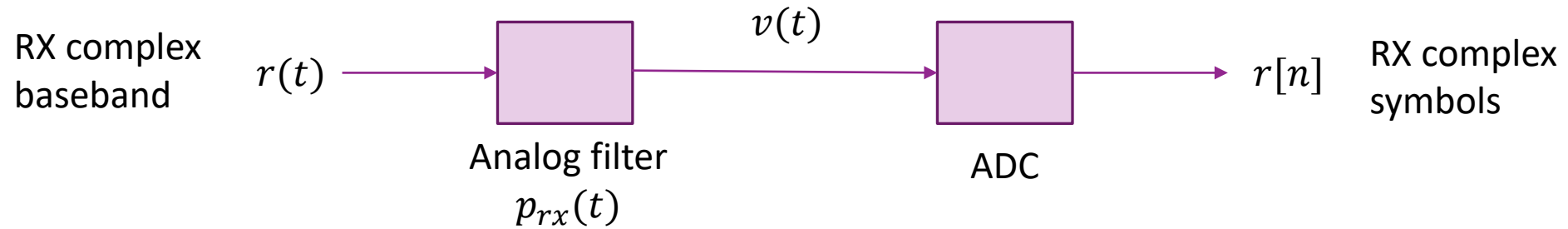
- Sample rate = 80 Msamples / sec
- Sample rate = 20 Msamples / sec



Outline

- ☐ Receiver filtering and sampling
- ☐ Perfect reconstruction with orthonormal modulation
- ☐ General channels: Time-domain analysis
- ☐ General channels: Frequency-domain analysis
- ☐ PSD Analysis
-  ☐ Practical RX filter design
- ☐ Channel sounding

Problems with Analog Filtering



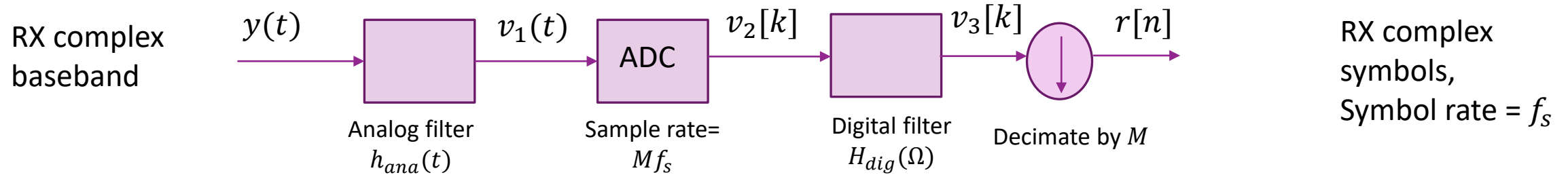
□ Up to now, we have considered two stage filtering

- Filtering: $v(t) = p_{rx}(t) * r(t)$
- Sampling / ADC: $r[n] = v(nT)$

□ Problem: Filtering is performed in analog

- Desire sharp filters to remove close adjacent carrier
- Difficult to design sharp filters

Typical Digital Implementation of RX Filtering



- ❑ Use combination of analog and digital filtering in four steps:
- ❑ Step 1. Analog filtering $v_1(t) = h_{ana}(t) * y(t)$
- ❑ Step 2. Sample at M times symbol rate: $v_2[k] = v_1(kT/M)$
 - M = oversampling ratio
- ❑ Step 3. Digitally filter: $v_3[k] = h_{dig}[k] * v_2[k]$
- ❑ Step 4. Decimate: $r[n] = v_3[nM]$.
 - Takes one every M samples

Frequency Domain Analysis

- Step 1: $V_1(f) = H_{ana}(f)Y(f)$
- Step 2: Sampling at $\frac{T}{M}$: $V_2(\Omega) = \frac{M}{T} \sum_{k=-\infty}^{\infty} V_1\left(\left(\frac{\Omega}{2\pi} + k\right)\frac{M}{T}\right)$
- Step 3: Digital filtering: $V_3(\Omega) = V_2(\Omega)H_{dig}(\Omega)$
- Step 4: Decimate: $R(\Omega) = \frac{1}{M} \sum_{m=0}^{M-1} V_3\left(\frac{\Omega + 2\pi m}{M}\right)$

- Assuming no aliasing: Effective RX pulse shape is $P_{rx}(f) = H_{ana}(f)H_{dig}\left(\frac{2\pi f}{f_s M}\right)$

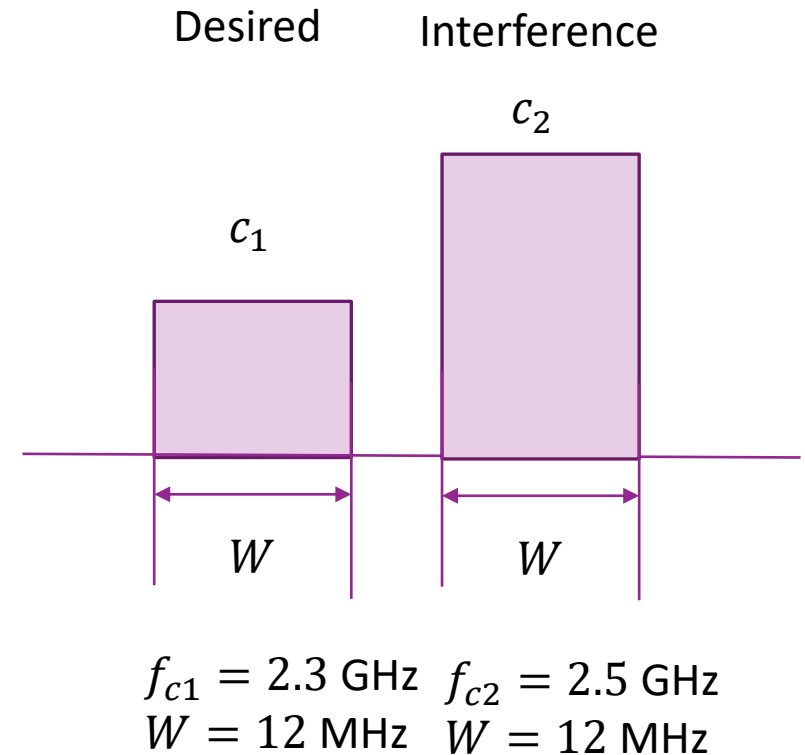
Frequency Domain Analysis: PSD

- Step 1: $S_1(f) = |H_{ana}(f)|^2 S_y(f)$
- Step 2: Sampling at $\frac{T}{M}$: $S_2(\Omega) = \frac{M}{T} \sum_{k=-\infty}^{\infty} S_1\left(\left(\frac{\Omega}{2\pi} + k\right)\frac{M}{T}\right)$
- Step 3: Digital filtering: $S_3(\Omega) = S_2(\Omega) |H_{dig}(\Omega)|^2$
- Step 4: Decimate: $R(\Omega) = \frac{1}{M} \sum_{m=0}^{M-1} S_3\left(\frac{\Omega + 2\pi m}{M}\right)$

- Assuming no aliasing: Effective RX pulse shape is $P_{rx}(f) = H_{ana}(f) H_{dig}\left(\frac{2\pi f}{f_s M}\right)$

Sample problem (Solution on board)

- ❑ Received passband signal $S_y(f)$ as shown
- ❑ Two components:
 - Desired signal
 - Nearby adjacent carrier signal
- ❑ Draw the following:
 - Complex baseband $R(f)$ after mixing at f_{c1}
 - Response after analog filtering with cutoff $|f| \leq 20$ MHz
 - Sampling at 40 Ms/s
 - Digital filtering with $|\Omega| \leq \pi/2$
 - Downsampling by 2



Filter Design

- ❑ Effective pulse shape: $P_{rx}(f) = H_{ana}(f)H_{dig}\left(\frac{2\pi f}{f_s M}\right)$
- ❑ Want $P_{rx}(f)$ to be low-pass with cutoff $f = \frac{1}{T}$
- ❑ Typical design for analog filter
 - $H_{ana}(f)$ passband up to $\frac{1}{2T}$, Stopband $\frac{2M-1}{2T}$
 - Removes images before sampling
 - Large transition region. Easy to design
- ❑ Design spec for digital filter
 - Low pass with digital cut-off frequency π/M
 - Typically very sharp to remove close adjacent carrier

Summary of Sampling Relations

Operation	Time domain	Frequency-domain	PSD
Ideal DAC	$u(t) = \sum_n s[n]p(t - nT)$	$U(f) = P(f)S\left(\frac{2\pi f}{f_s}\right)$	$S_u(f) = \frac{1}{T}S_s(2\pi fT) P(f) ^2$
Digital upsampling	$s[n] = \begin{cases} x[k] & n = kM \\ 0 & \text{else} \end{cases}$	$S(\Omega) = X(M\Omega)$	$S_s(\Omega) = \frac{1}{M}S_x(M\Omega)$
Ideal ADC	$r[n] = v(nT)$	$R(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} V(f_k)$ $f_k = \left(\frac{\Omega}{2\pi} + k\right)\frac{1}{T}$	$S_r(\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} S_v(f_k)$ $f_k = \left(\frac{\Omega}{2\pi} + k\right)\frac{1}{T}$
Ideal ADC. No aliasing	$r[n] = v(nT)$	$R(\Omega) = \frac{1}{T}V\left(\frac{\Omega}{2\pi T}\right)$	$S_r(\Omega) = \frac{1}{T}S_v\left(\frac{\Omega}{2\pi T}\right)$
Digital decimation	$r[n] = x(nM)$	$R(\Omega) = \frac{1}{M} \sum_{m=0}^{M-1} X\left(\frac{\Omega + 2\pi m}{M}\right)$	$S_r(\Omega) = \frac{1}{M} \sum_{m=0}^{M-1} S_x\left(\frac{\Omega + 2\pi m}{M}\right)$
Digital decimation. No aliasing	$r[n] = x(nM)$	$R(\Omega) = \frac{1}{M}X\left(\frac{\Omega}{M}\right)$	$S_r(\Omega) = \frac{1}{M}S_x\left(\frac{\Omega}{M}\right)$

Summary of Units

Operation	Quantity	Time domain units	PSD units
TX symbols	$s[n]$	$ s[n] ^2$: Energy per sample	$S_s(\Omega)$: Energy per sample per radian
TX modulation	$u(t) = \sum p_{tx}(t - nT)s[n]$	$ p_{tx}(t) ^2$: Samples / sec	$ P_{tx}(f) ^2$: Samp / Hz = Samp x sec
		$ u(t) ^2$: Energy / sec = power	$S_u(f)$: Power / Hz = Energy
RX signal	$y(t)$	$ y(t) ^2$: Energy / sec = power	$S_y(f)$: Power / Hz = Energy
RX filtered & sampled	$v(t) = p_{rx}(t) * y(t)$ $r[n] = v(nT)$	$ p_{rx}(t) ^2$: 1/(samp x time)	$ P_{rx}(f) ^2$: 1/(samp x Hz)=time/samp
		$ r[n] ^2$: Energy per sample	$S_r(\Omega)$: Energy per sample per radian

Example RX Filtering Circuit

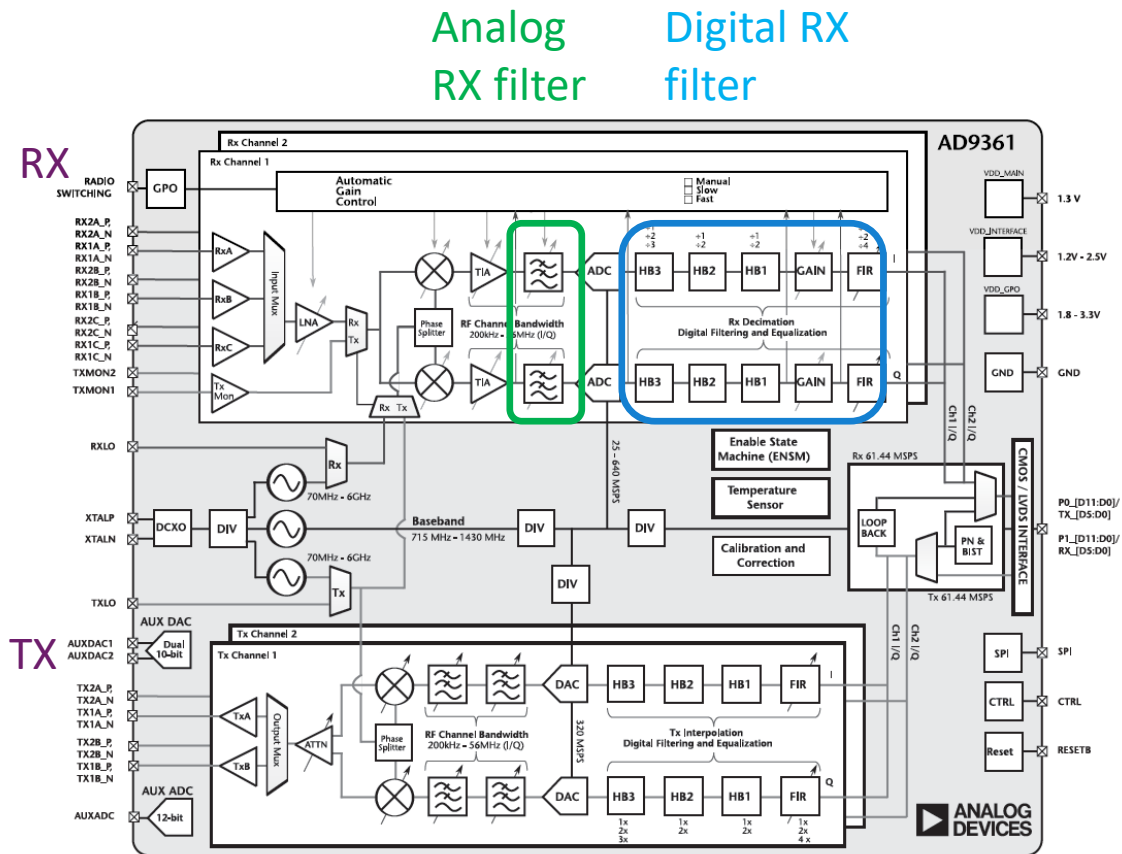


Figure 1.8 Integrated ZIF architecture used in the Pluto SDR.

❑ Analog Devices AD9361 Wideband TXCR

❑ RX filtering performed in two stages:


❑ Analog filtering (before ADC)

- Depends on channel bandwidth

❑ Digital filtering:

- Multiple stages of interpolation
- Programmable depending on sample rate

Outline

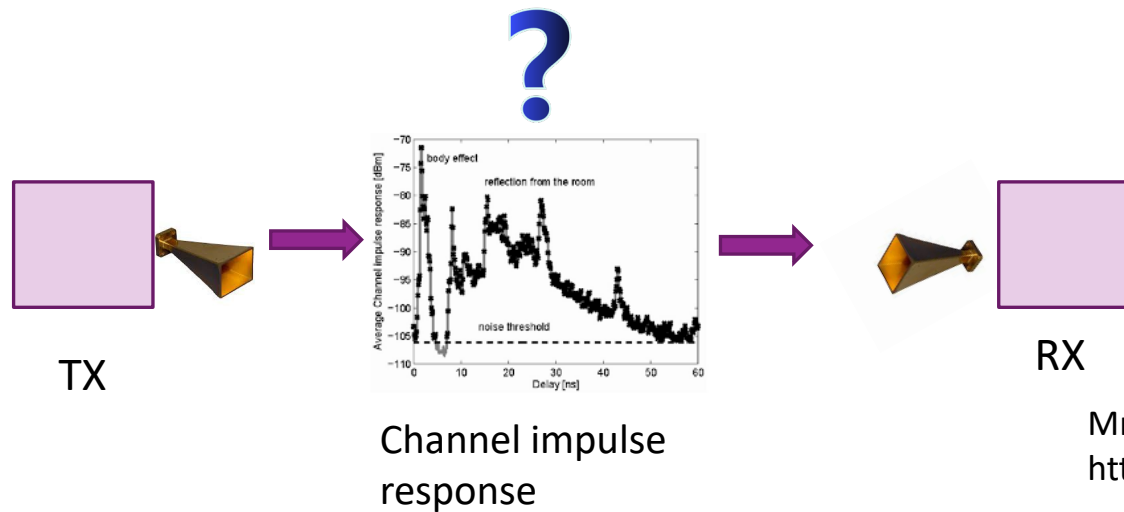
- ❑ Receiver filtering and sampling
- ❑ Perfect reconstruction with orthonormal modulation
- ❑ General channels: Time-domain analysis
- ❑ General channels: Frequency-domain analysis
- ❑ PSD Analysis
- ❑ Practical RX filter design
-  Channel sounding

Channel Sounder

□ Measure channel

□ Useful for:

- Wireless propagation analysis
- Measure multipath components, signal strengths, directions of arrival
- But, also good for debugging any front-end



MmWave and sub-THz channel sounder by Rappaport lab
<https://wireless.engineering.nyu.edu/mmwave-5g-and-6g-channel-sounder/>

FFT-Based Channel Sounding

- ❑ Effective discrete-time channel: $r[n] = h[n] * s[n]$
- ❑ Select period N
- ❑ Repeatedly transmit $s[n] = \sum_{k=0}^{N-1} S_k e^{2\pi i k n / N} = \text{FFT}(S_k)$
- ❑ Receiver will get: $r[n] = \sum_{k=0}^{N-1} R_k e^{2\pi i k n / N}$, $R_k = S_k H\left(\frac{2\pi k}{N}\right)$
- ❑ Recover $R_k = \text{IFFT}(r[n])$
- ❑ Get channel response: $H\left(\frac{2\pi k}{N}\right) = \frac{R_k}{S_k} = H_{chan}\left(\frac{k}{NT}\right) P_{tx}\left(\frac{k}{NT}\right) P_{rx}\left(\frac{k}{NT}\right)$
- ❑ Learn $P_{tx}\left(\frac{k}{NT}\right), P_{rx}\left(\frac{k}{NT}\right)$ from calibration
- ❑ Estimate $H_{chan}\left(\frac{k}{NT}\right)$
- ❑ Provides a discrete estimate of the channel.
- ❑ More in the lab