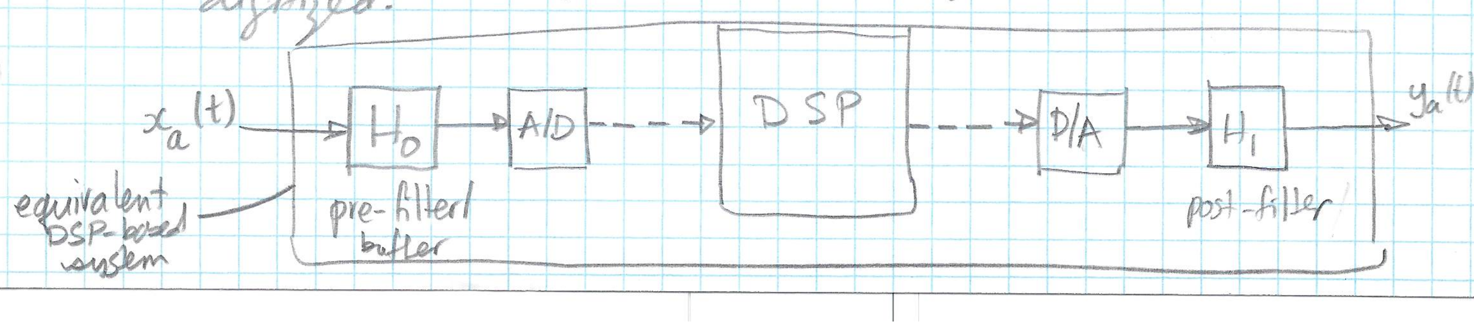
From Ingle & Proakis

- signals can be from various sources
  - natural
  - artificial (man-made)
- in engineering signals are carriers of information, both useful and unwanted
- a major goal of signal processing is to extract this useful information but more generally SP is an operation designed for:
  - extracting
  - enhancing
  - storing
  - or transmitting

  this useful information
- most signals are analog, and can be processed directly with analog signal processing.

  - but the range of mathematical operations available to us in analog form is limited
  - furthermore, there are typically noise issues owing to thermal and shot sources which limit the quality of the processing we can do w/ analog electronics.
- using digital techniques offers greater flexibility and provides potentially higher performance; however we require the analog input to be digitized:



two categories of DSP: analysis and filtering

e.g.)
- spectral analysis
- speech recognition
- speaker verification
- target detection

e.g.)
- noise removal
- spectral shaping
- separation of content by frequency range
- conditioning

DISCRETE-TIME SIGNALS

- analog signals denoted with t independent variable:

$$x_a(t) \quad , t \in \mathbb{R}$$

- discrete-time signals use n as the independent variable:

$$x(n)$$

$$x(n), \quad n \in \mathbb{Z}$$

IN MATLAB YOU'D DEFINE IT AS A ROW VECTOR:

e.g. $x = [-15, -6, +3, 4, -1, 6, 7];$

$\Rightarrow x(1) = -15,$
$x(2) = -6$
$\vdots$

① unit sample

$$\delta(n) = \begin{cases} 1, & n=0 \\ 0, & n \neq 0 \end{cases} = \{\ldots, 0, 0, \underset{\uparrow}{1}, 0, 0, \ldots\}$$

arrow used to indicate $n=0$.

② unit step sequence

$$u(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} = \{\ldots, 0, 0, \underset{\uparrow}{1}, 1, 1, \ldots\}$$

③ real-valued exponential sequence

$$x(n) = a^n, \quad n \in \mathbb{Z}, \ a \in \mathbb{R}$$

MATLAB:

```
>> n = [0:10];
>> x = (0.9).^n;
```

④ complex-valued exponential sequence:

$$x(n) = e^{(\sigma + j\omega_0)n}, \quad n \in \mathbb{Z}, \ j = \sqrt{-1}, \ \sigma, \omega_0 \in \mathbb{R}$$

MATLAB:

```
>> n = [-8:7];
>> x = exp((2+3*j)*n);
```

⑤ sinusoidal sequence:

$$x(n) = A\cos(\omega_0 n + \theta_0), \quad n \in \mathbb{Z}, \ A, \omega_0, \theta_0 \in \mathbb{R}.$$

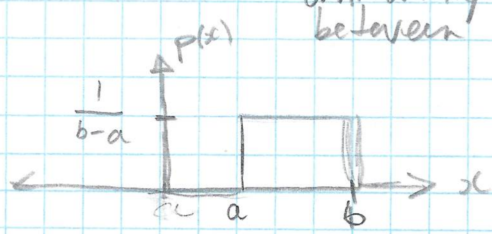
MATLAB:

```
>> n = [-5:5];
>> x = 3*cos(0.1*pi*n + pi/3) + 2*sin(0.5*pi*n);
```

⑥ random sequences

- need parameters associated w/ pdfs
- MATLAB: rand(1,N) ← random sequence of N numbers, uniformly distributed between 0 and 1

$\int p(x)\,dx = 1$



ESE 2014: Systems Review

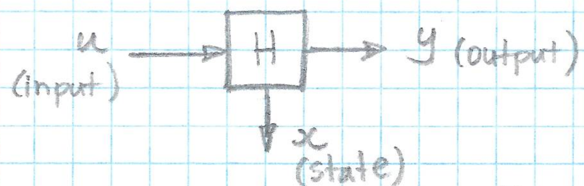
- recall the two major aspects of DSP:

  1) Analysis:
     - spectral/frequency-domain (FFT/DFT)
     - statistical (histogram, time averages, auto-correlat.)
     - comparative — e.g., difference between given signal and a known reference

  2) Filtering:
     - noise removal
     - shaping/equalization
     - data conversion
     - stabilization/control

- But how do we design filters?
  → filters are <u>dynamic systems</u>

- A general class of nonlinear dynamic systems:



- vector-valued quantities: $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$, $x \in \mathbb{R}^n$ } MIMO system w/ m inputs, p outputs, and n states
- in <u>continuous time</u>:

$$H: \begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{aligned}$$

where $f: \mathbb{R}^{n \times m} \to \mathbb{R}^n$ and $h: \mathbb{R}^{n \times m} \to \mathbb{R}^p$ are vector-valued functions of several variables

EX.: biological/physiological compartmental models, electronic systems, mechanical/robotic systems e.g.)

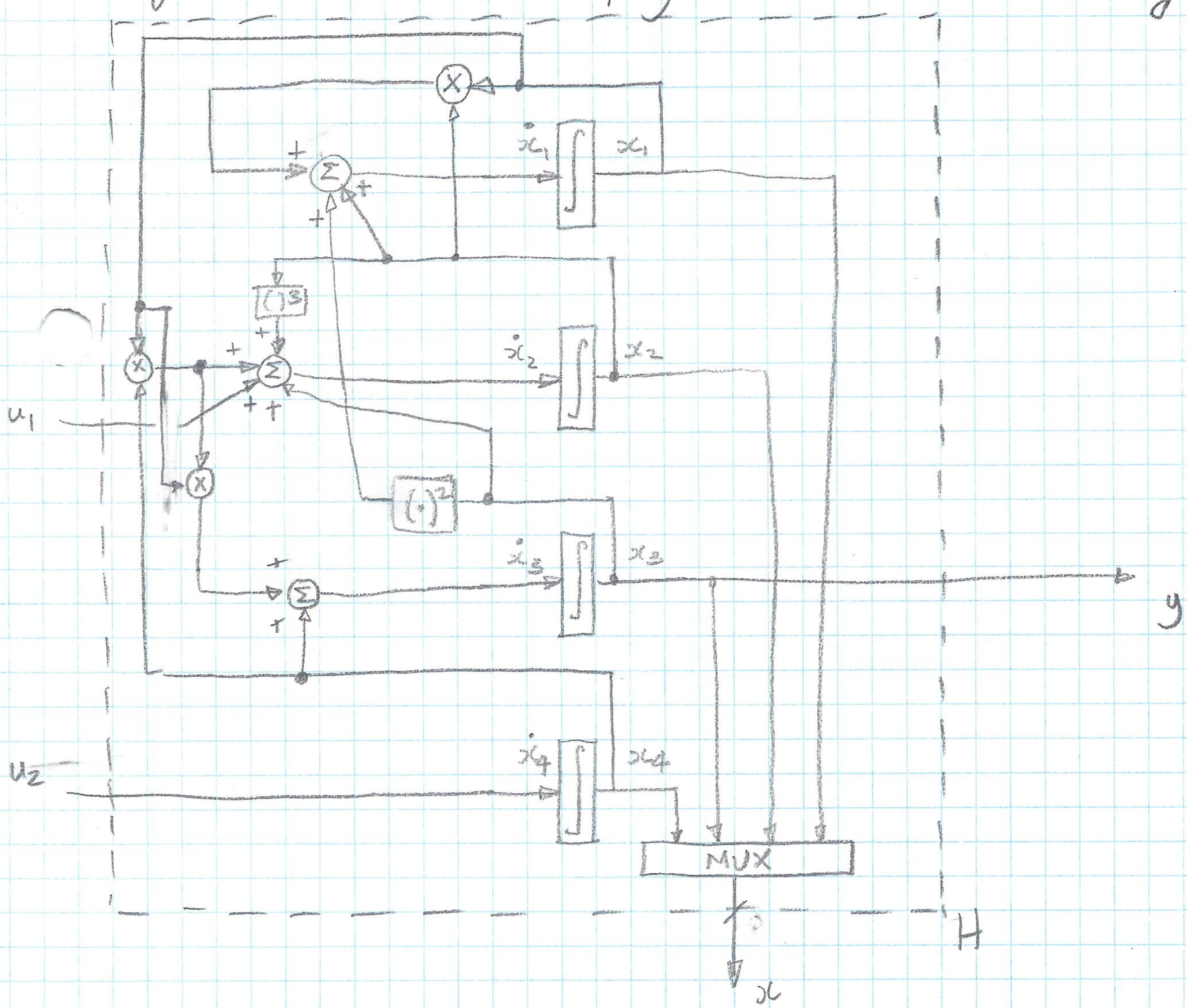
$$\begin{aligned}
\dot{x}_1 &= x_2 + x_3^2 + x_1 x_2 \\
\dot{x}_2 &= x_3 + x_4 x_1 + x_2^3 + u_1 \\
\dot{x}_3 &= x_4 + x_1^2 x_2 \\
\dot{x}_4 &= u_2 \\
y &= x_3
\end{aligned}
\quad \Bigg\} \quad
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \quad y = x_3, \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$n = 4$, $p = 1$, $m = 2$

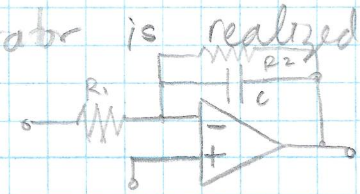
- this is a nonlinear state-space model.
- as opposed to other system descriptions, the state-space model gives an explicit <u>system realization</u>, that is, <u>a way to build it</u>!

<u>TO REALIZE:</u>

- for each state, simply introduce an integrator:



An integrator is realized in analog form as



$$\frac{1}{s} \sim \frac{1}{\tau s + \alpha_0}$$