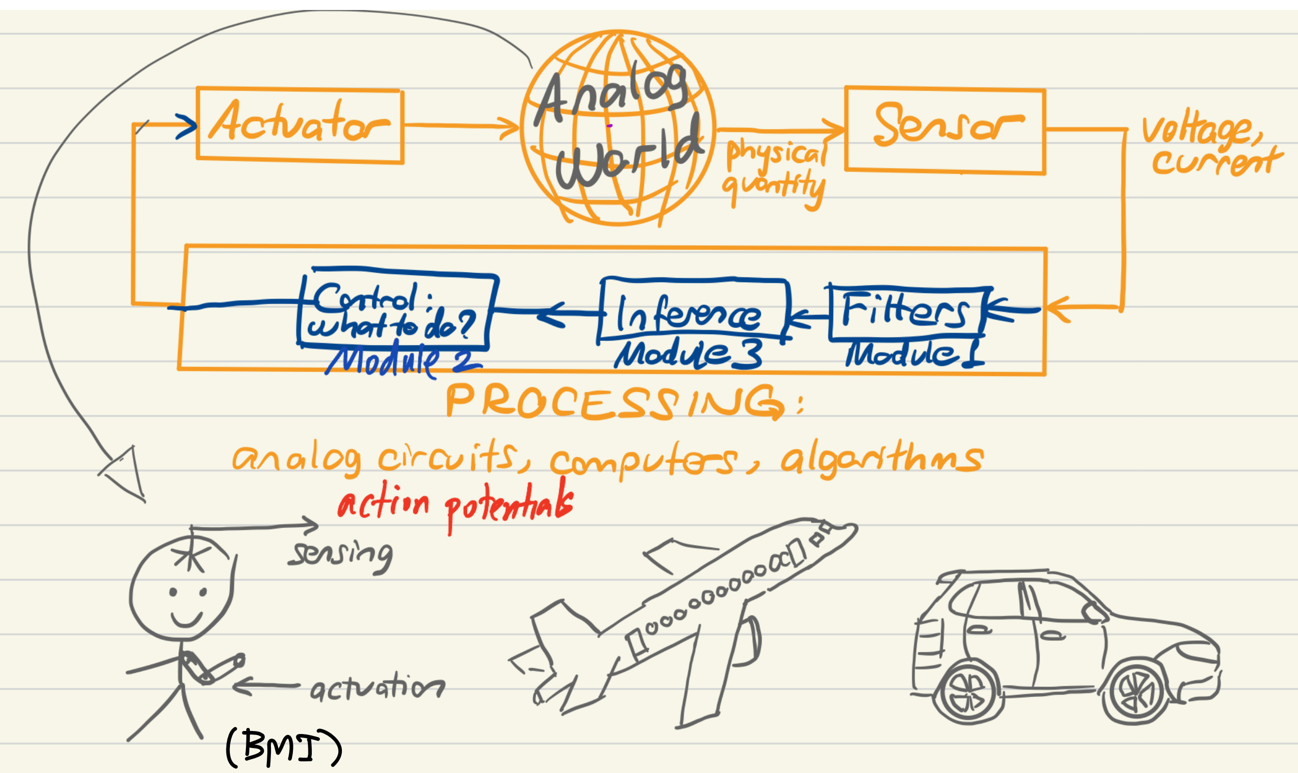
ROBOTICS & CONTROL ("Module" 2)

Today:
- Intro to control
- Discrete-time & continuous-time model conversion
- System identification

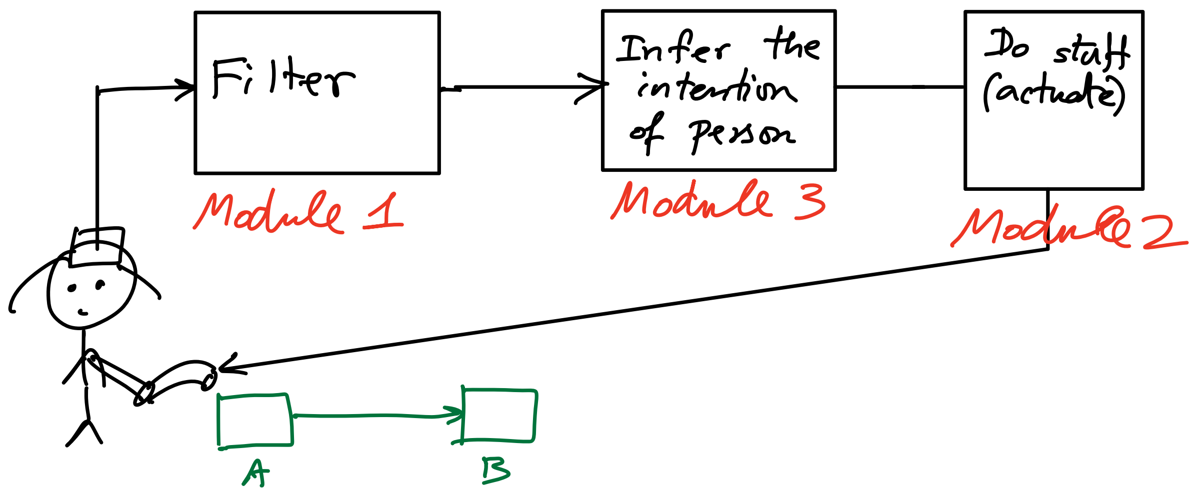
"BIG PICTURE" (EECS 16B view of the world: control perspective)

CONTROL THEORY: "Field of engineering & applied math dealing with the control of dynamical systems in machines & engineered processes." (WIKI):
- Foundations of Robotics & AI
- "Foundations of ML for a dynamic world."



Brain Machine Interface (BMI)

Example



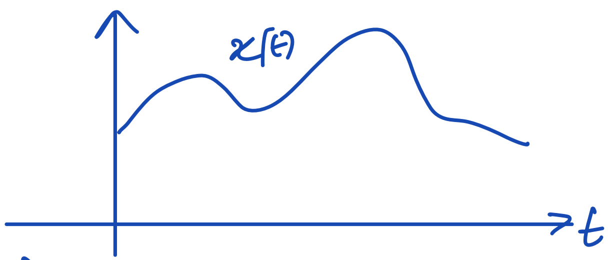
Other examples:

- cruise-control in cars designed to maintain a constant speed target. control $\Rightarrow$ throttle position of engine which determines how much power the engine delivers.
- temperature control through thermostat: switch boiler on/off depending on Δ between set temp. & measured temp.
- AI (Reinforcement Learning)
- Robotics (Lab: "robo car")
- Biological systems (regulating sugar levels)
- Space-Shuttle (Perseverence Mars Rover)

⋮

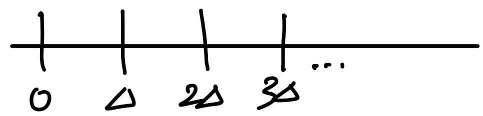
see Sp'21 EECS 16B Lec. 6B

Control & inference blocks are algorithms executed digitally in a computer, in discrete time. Rest of the system evolves in continuous time. How do we connect the two worlds?



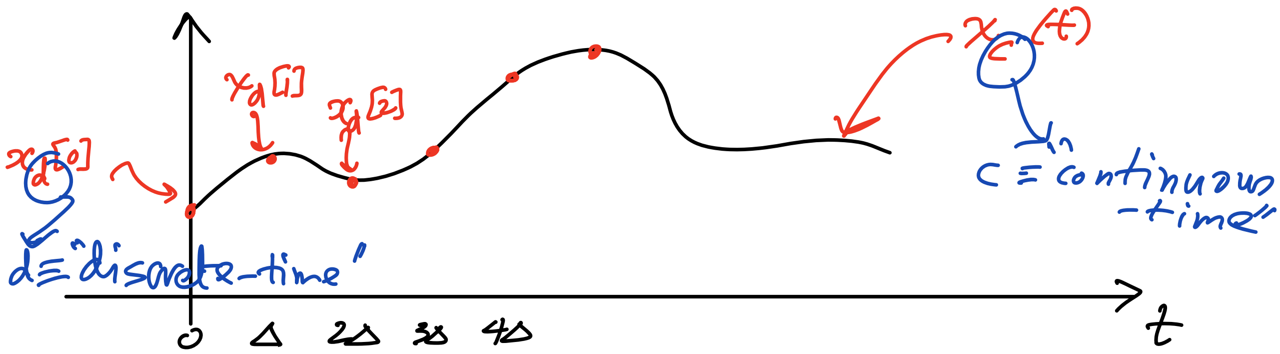
"Analog World"

"control" world



(Finite clock-ticks)

Interface through "SAMPLING"



sample i corresponds to CT sys. time $\boxed{i\Delta}$.

Physics Laws: e.g. $F = ma$

$$\frac{d\vec{x}_c(t)}{dt} = A \underbrace{\vec{x}_c(t)}_{\text{"state" of system}} + B \underbrace{\vec{u}_c(t)}_{\text{"control" input}}$$

design choice of engineer

var. you "care about"

x: "state" variable for control system

e.g. (i) inductor current, capacitor voltage, etc. in RLC circuits

(ii) position (x, y, z), velocity (v_x, v_y, v_z), acceleration (a_x, a_y, a_z) etc. in automobile/aircraft control sys., etc.

u: "control" variable for control system

e.g. (i) thrust power in aircraft control

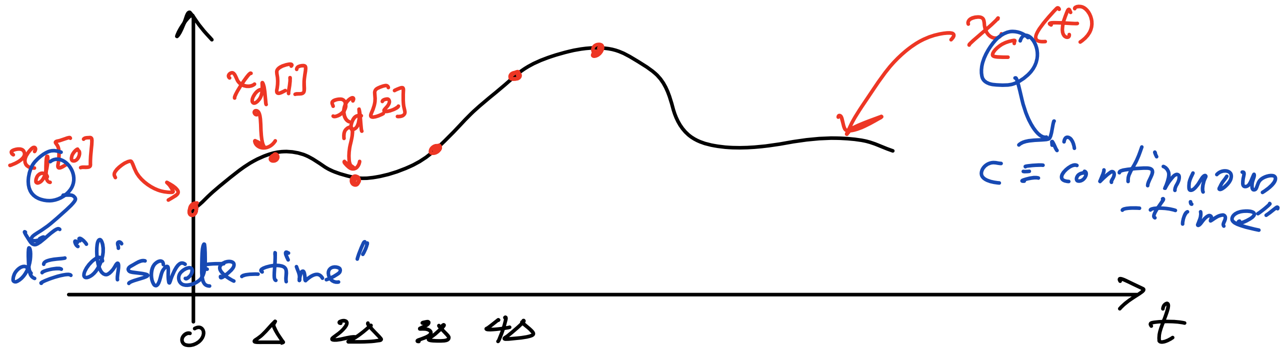
(ii) throttle position in cruise control

(iii) independent voltage or current source in RLC circuits.

<u>Note:</u> $\vec{x}(t)$ and $\vec{u}(t)$ can have different dimensions!

Conventionally, $\vec{x}(t) = \underbrace{\begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}}_{(n \times 1)}$; $\vec{u}(t) = \underbrace{\begin{bmatrix} u_1(t) \\ \vdots \\ u_m(t) \end{bmatrix}}_{(m \times 1)}$

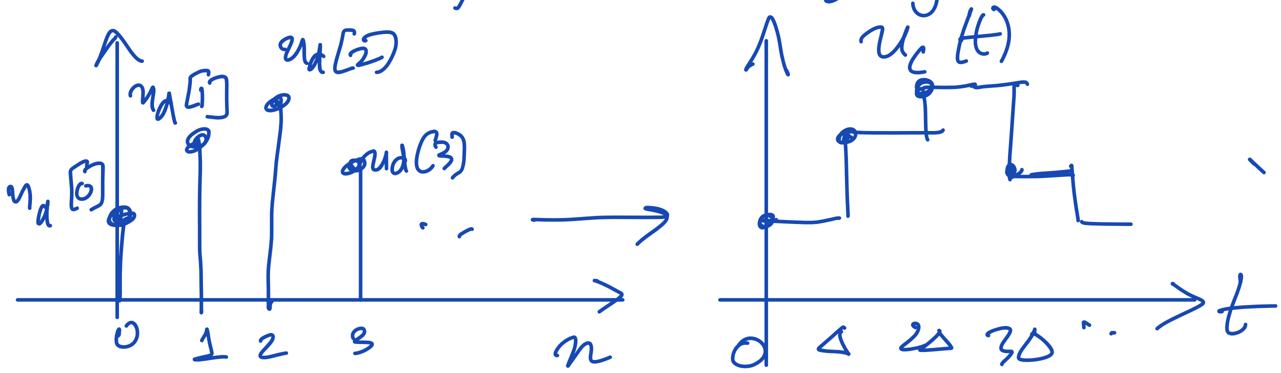
Interface through "SAMPLING"



sample i corresponds to CT sys. time $\boxed{i\Delta}$.

$$\boxed{\vec{x}_d[i] = \vec{x}_c(i\Delta)} \; ; \; i = 0, 1, \ldots$$

How is the input $u_c(t)$ going to work?

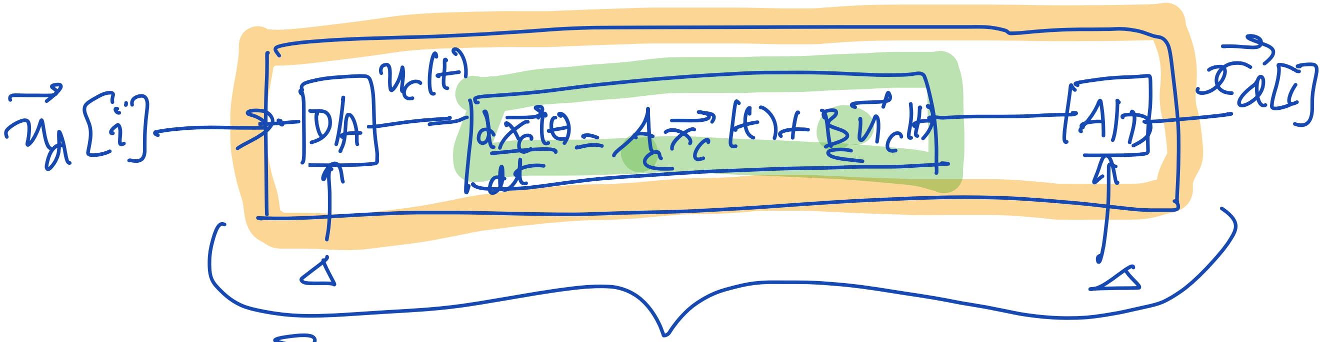


ZOH (zero-order hold)

piecewise-constant (p.c.)

D/C block
or D/A

(ZOH) $\boxed{u_c(t) = u_d[i] \text{ for } t \in [i\Delta, (i+1)\Delta)}$



difference eqn.

$$\vec{x}_d[i+1] = A_d\,\vec{x}_d[i] + B_d\,\vec{u}_d[i]$$

- Recurrence relations that describe how the sys. evolves from sample to sample.

Q) How to find A_d, B_d, given A_c, B_c (and Δ)

i.e. given $\vec{x}_d[i]$ and $\vec{u}_d[i]$, what is $\vec{x}_d[i+1]$ s.t. it represents $x_c((i+1)\Delta)$ of the CT system?

$\vec{x}_d[i] = x_c(i\Delta)$; $\vec{x}_d[i+1] = x_c((i+1)\Delta)$

$\vec{x}_d[i+1]$ is the soln. to the DE:

$$\frac{d\vec{x}_c(t)}{dt} = A\vec{x}_c(t) + B\vec{u}_c(t)$$

@ time $t = (i+1)\Delta$ from

I.C. $\vec{x}_c(i\Delta) = \vec{x}_d[i]$

@ time $t = i\Delta$