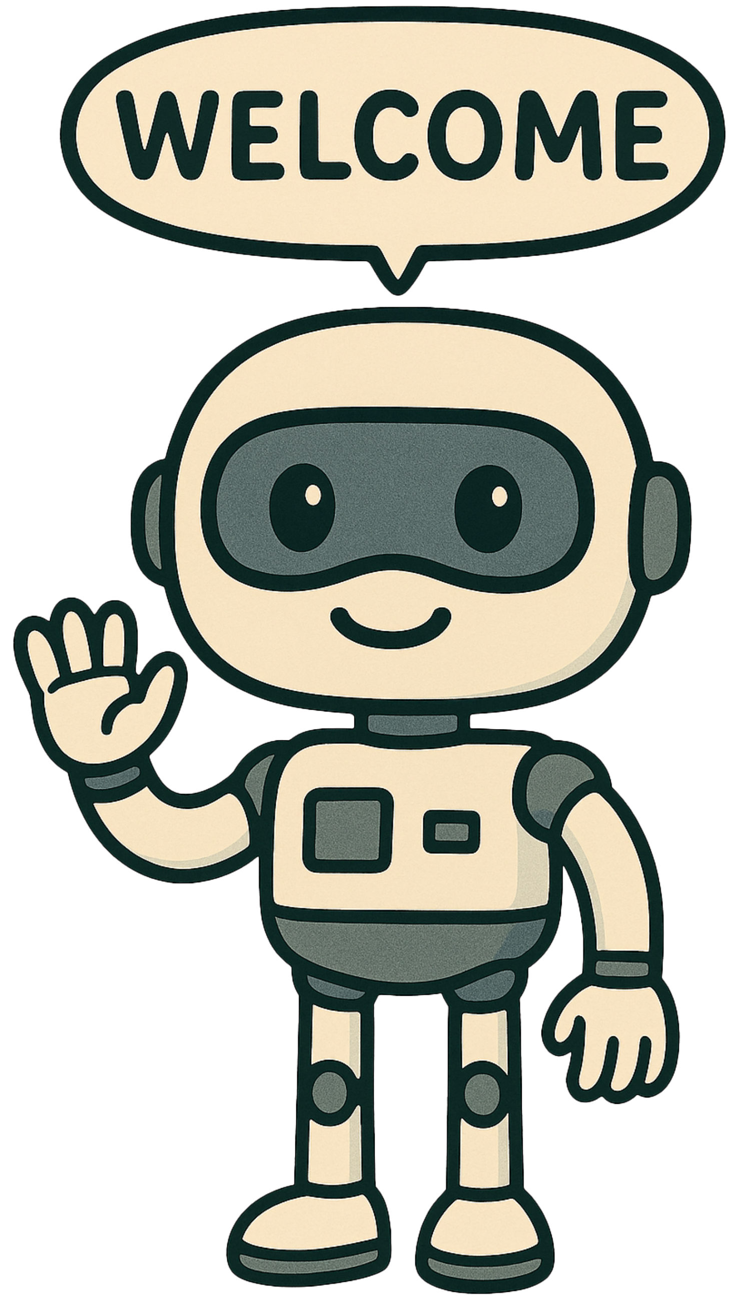
L1: Course Introduction

ELEC 341 | Systems and Control | Spring 2026

Cyrus Neary | cyrus.neary@ubc.ca



Who am I? A bit about your instructor!

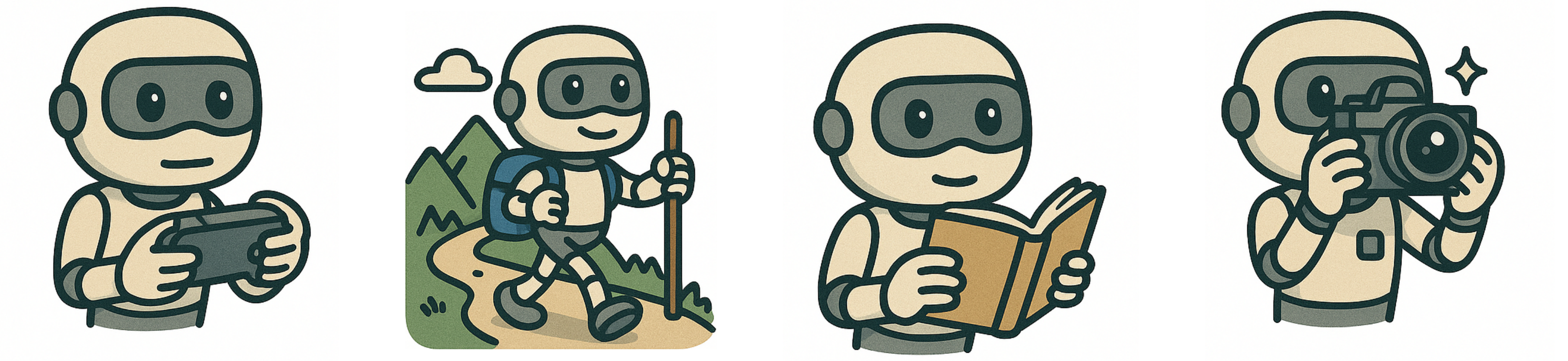
Undergrad from UBC in Engineering Physics and Mathematics.

MSc and PhD from The University of Texas at Austin in Computational Science, Engineering, and Mathematics.

Postdoctoral research at Mila - The Québec AI Institute.

Assistant professor in Electrical and Computer Engineering.
Research: Theory, algorithms, and applications of AI-driven autonomous systems in robotics and engineering.

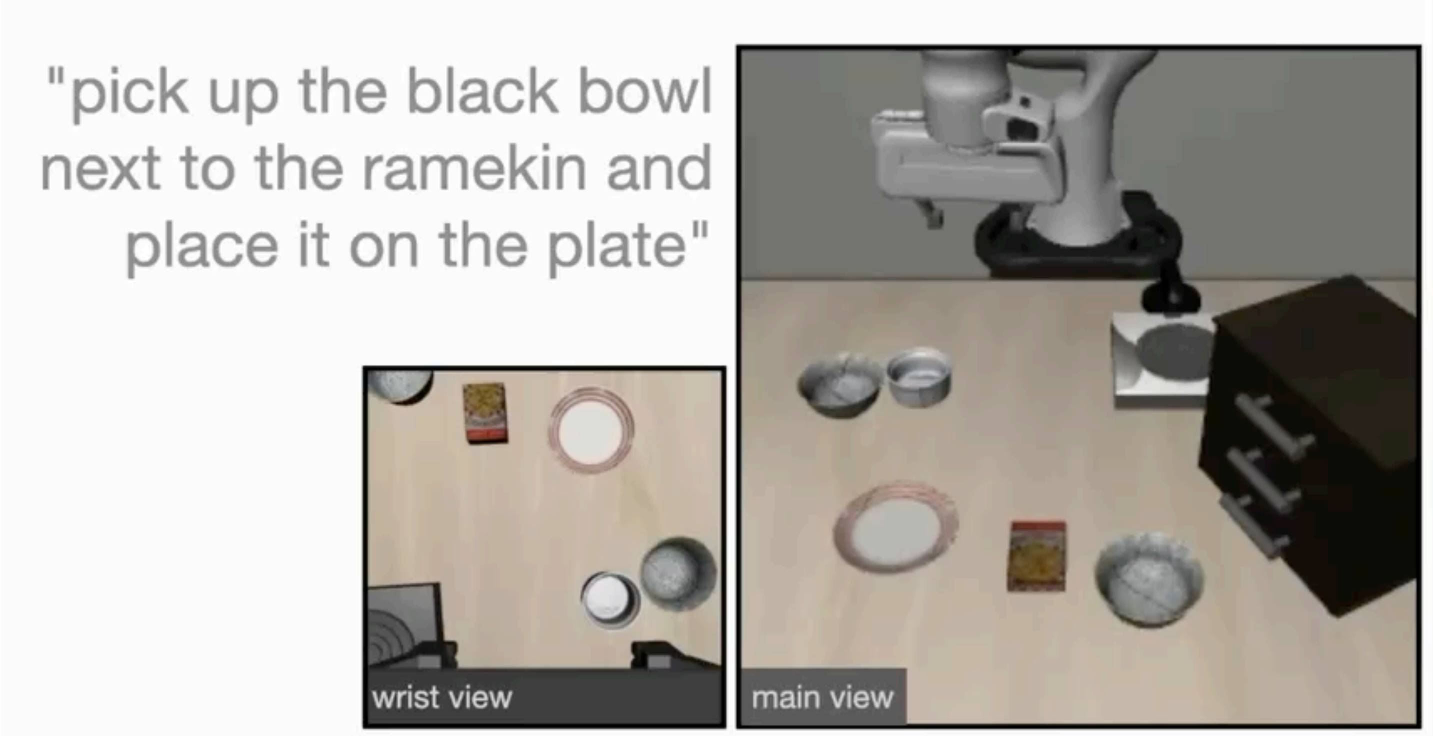
Hobbies: Reading (sci-fi/fantasy), photography, hiking, gaming.



Contact: cyrus.neary@ubc.ca. Include [Spring26ELEC341] in subject line.

What do I work on at UBC?

Robots that safely respond directly to natural language

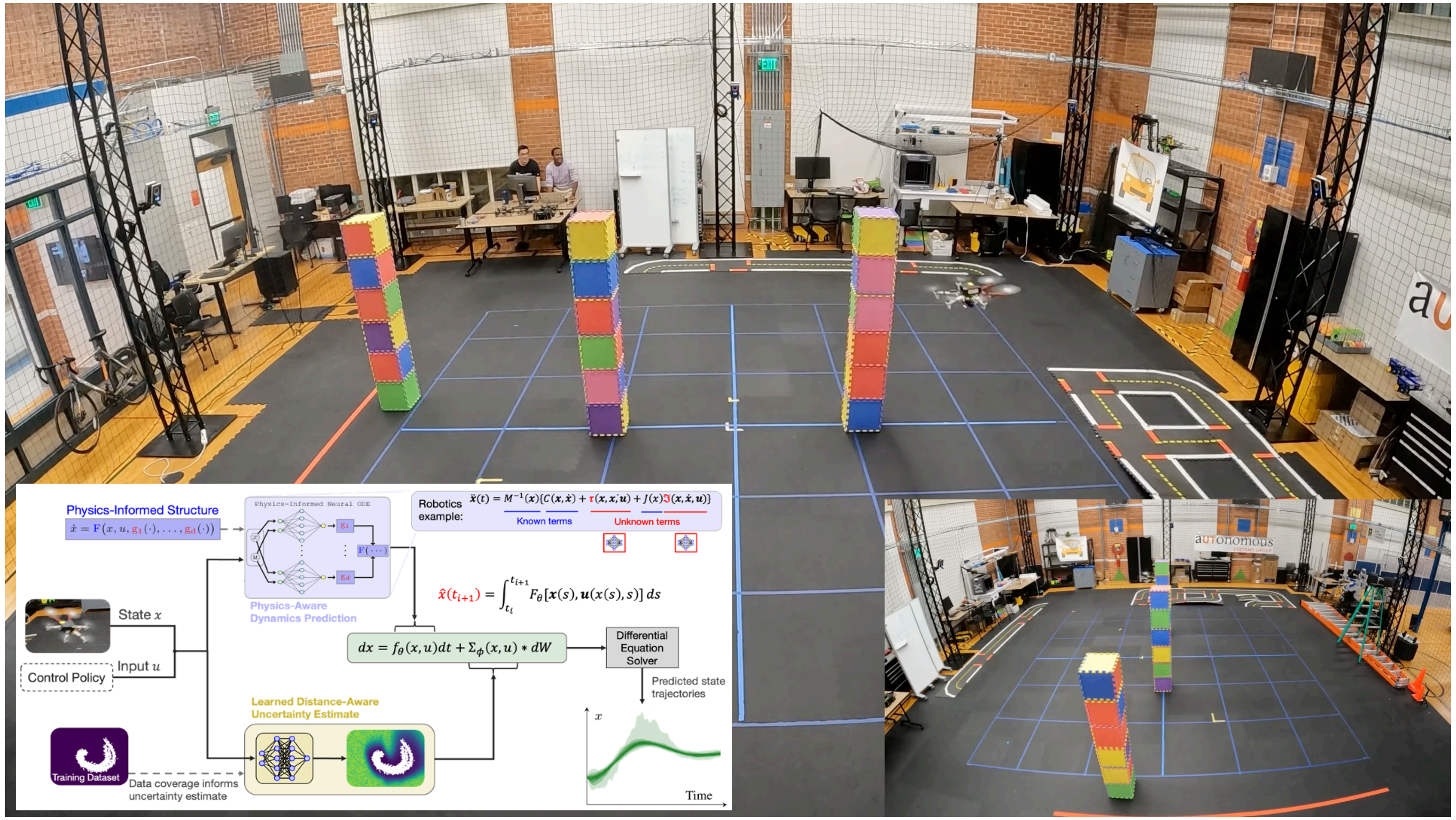


Robots that autonomously reason over complex, long-horizon objectives

AIRE

How can we transform **AI capabilities** into **autonomous engineering systems**?

Physics-informed deep learning for predictive modeling & control



Who are all of you?

How do we make drones fly autonomously?


Article

Champion-level drone racing using deep reinforcement learning

https://doi.org/10.1038/s41586-023-06419-4

Received: 5 January 2023

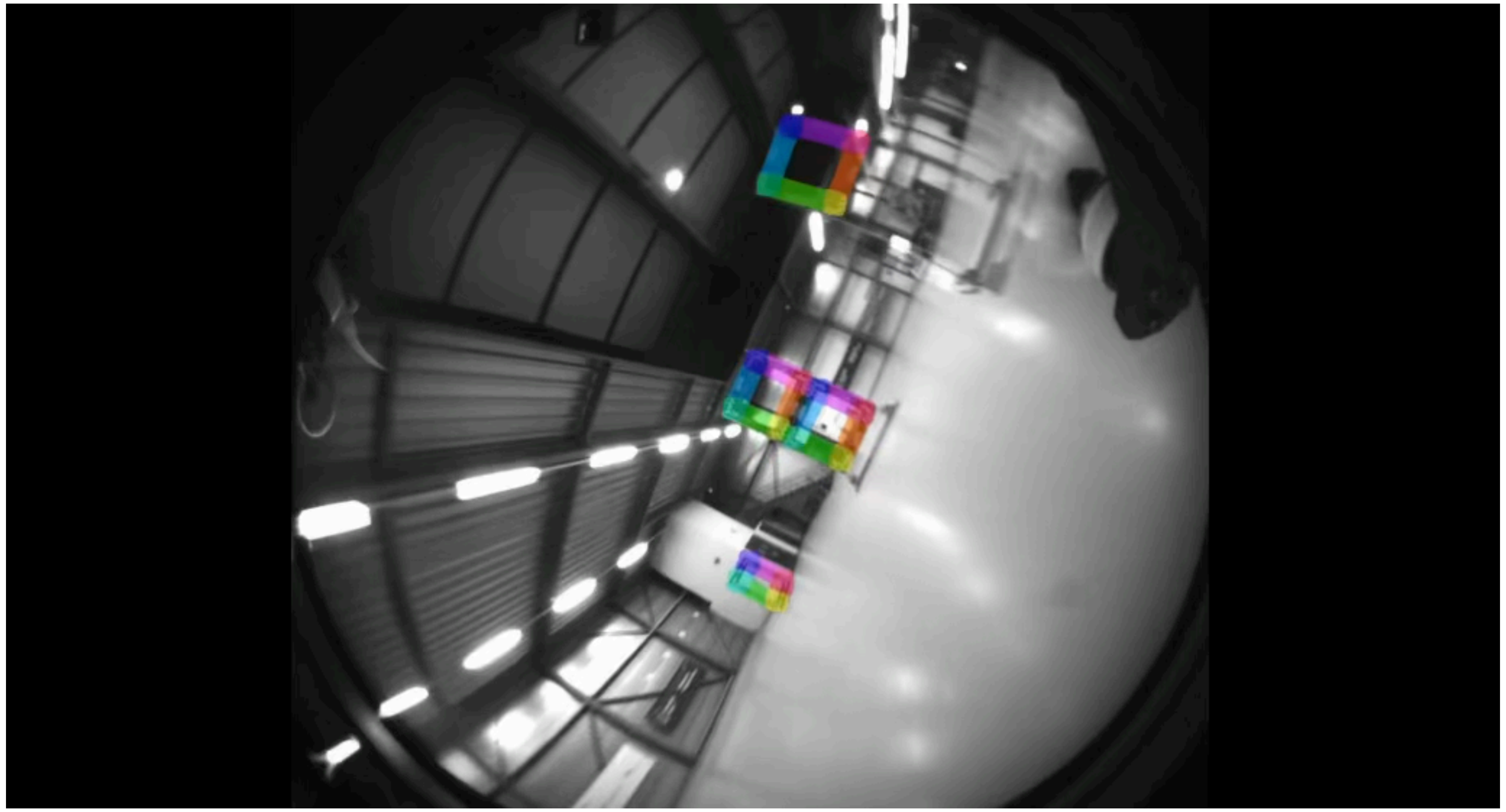
Accepted: 10 July 2023

Published online: 30 August 2023

Open access

Elia Kaufmann[1✉], Leonard Bauersfeld[1], Antonio Loquercio[1], Matthias Müller[2], Vladlen Koltun[3] & Davide Scaramuzza[1]

First-person view (FPV) drone racing is a televised sport in which professional competitors pilot high-speed aircraft through a 3D circuit. Each pilot sees the



How do we make drones ~~fly autonomously~~ hover?

$F_c = ???$

Desired hover height

$F_g = mg$

How do we pick the signals to send to the propellors?

How can we design an algorithm that automatically sends the right signals?

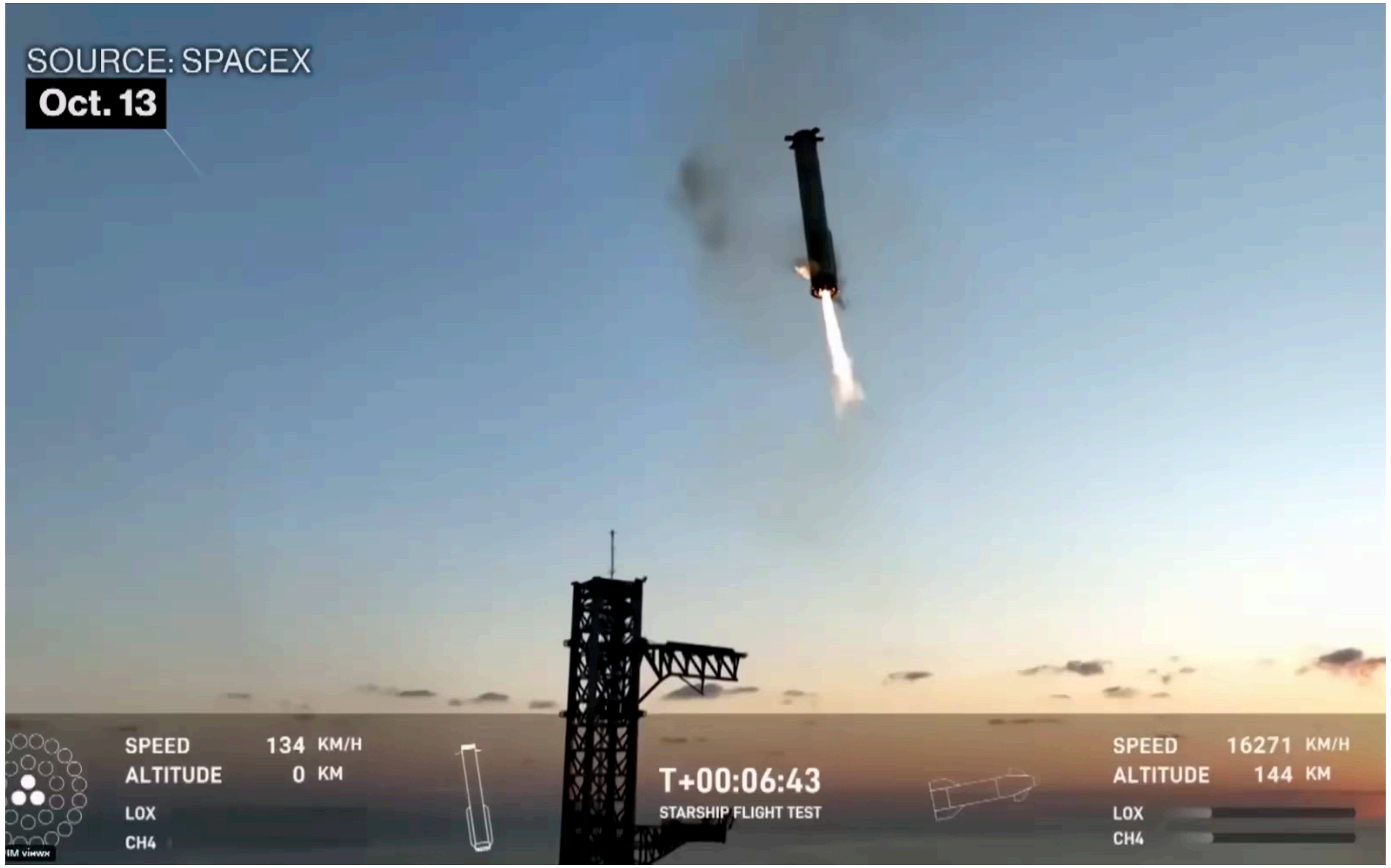
What information (e.g., setpoint, current drone height, dynamics model) would this algorithm need?

What design requirements should such an algorithm meet? How do we even articulate those requirements?

Why study control systems?

Aerospace engineering

How do you **land** a multi-story building falling from space??



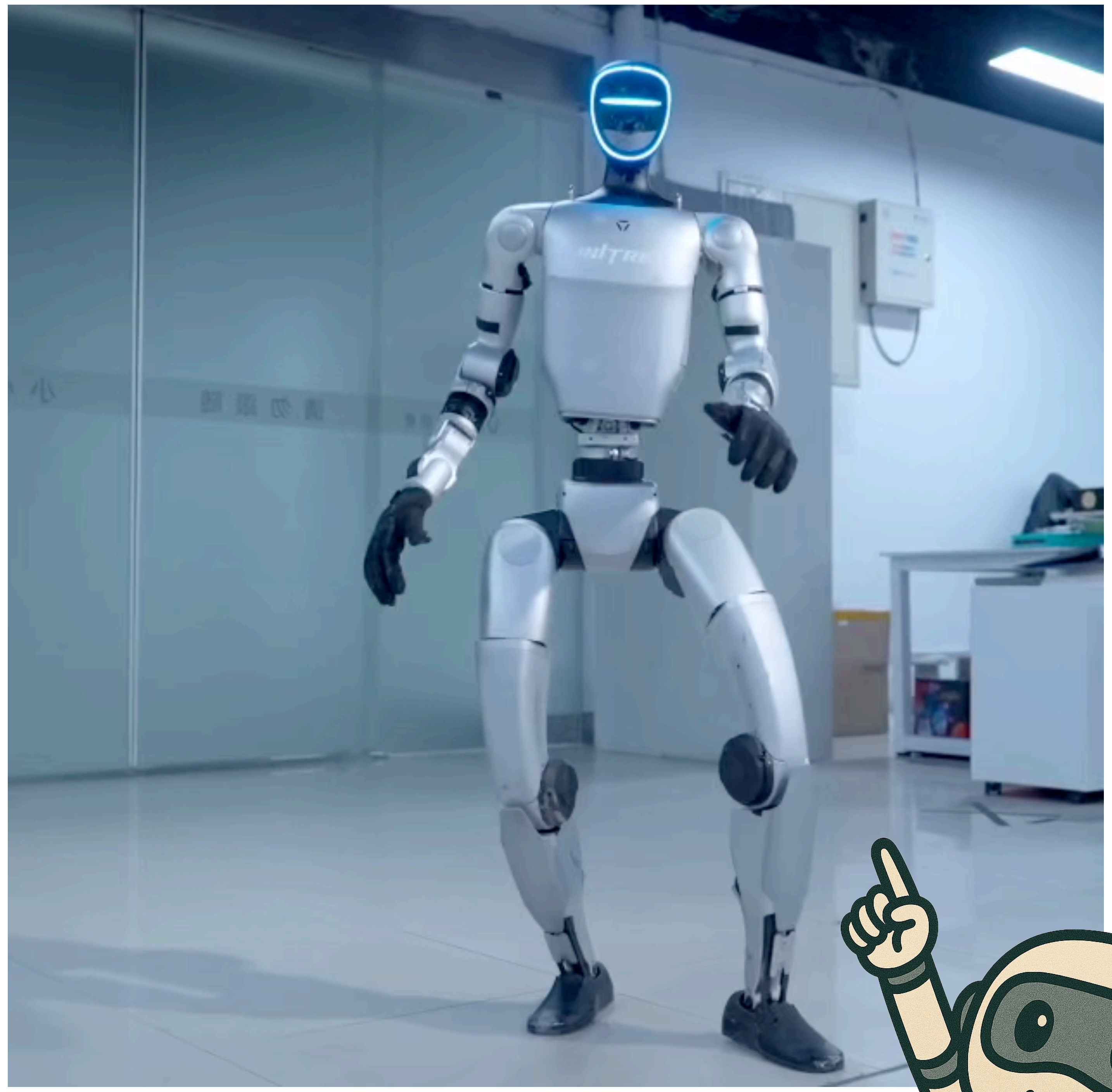
Or safely fly an airplane that's carrying hundres of passengers?

Why study control systems?

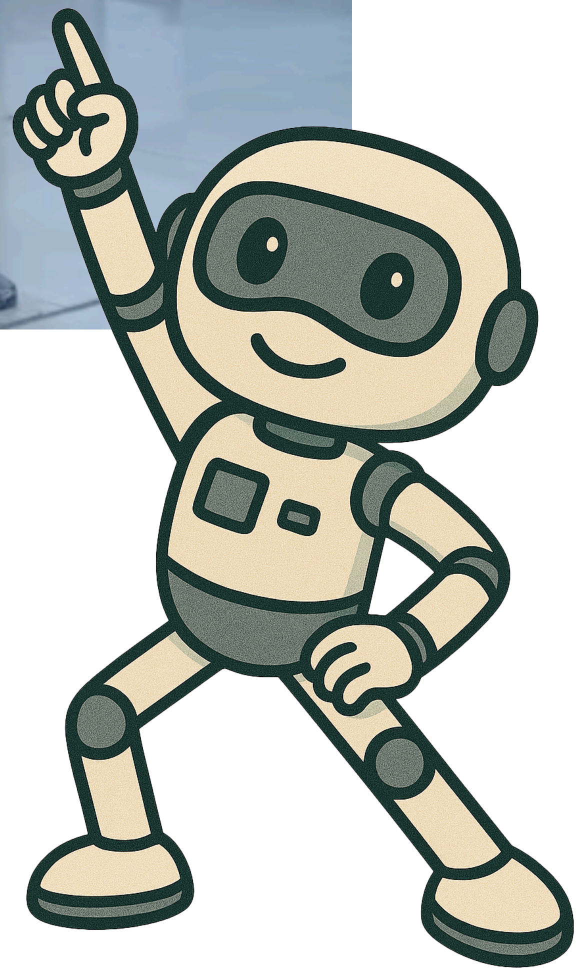
Robots…

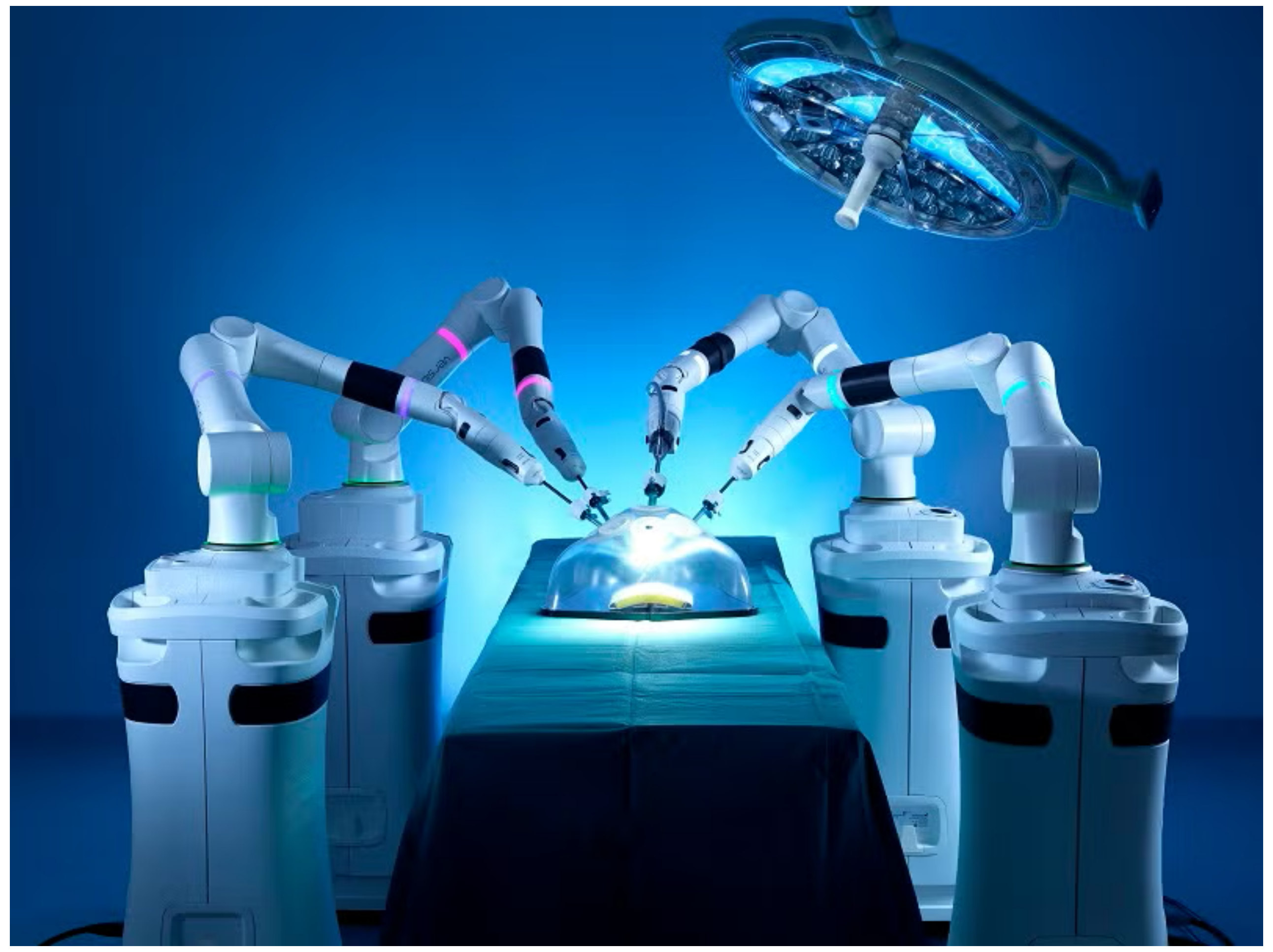
That optimize supply chains…

that explore the unknown…



and that dance!

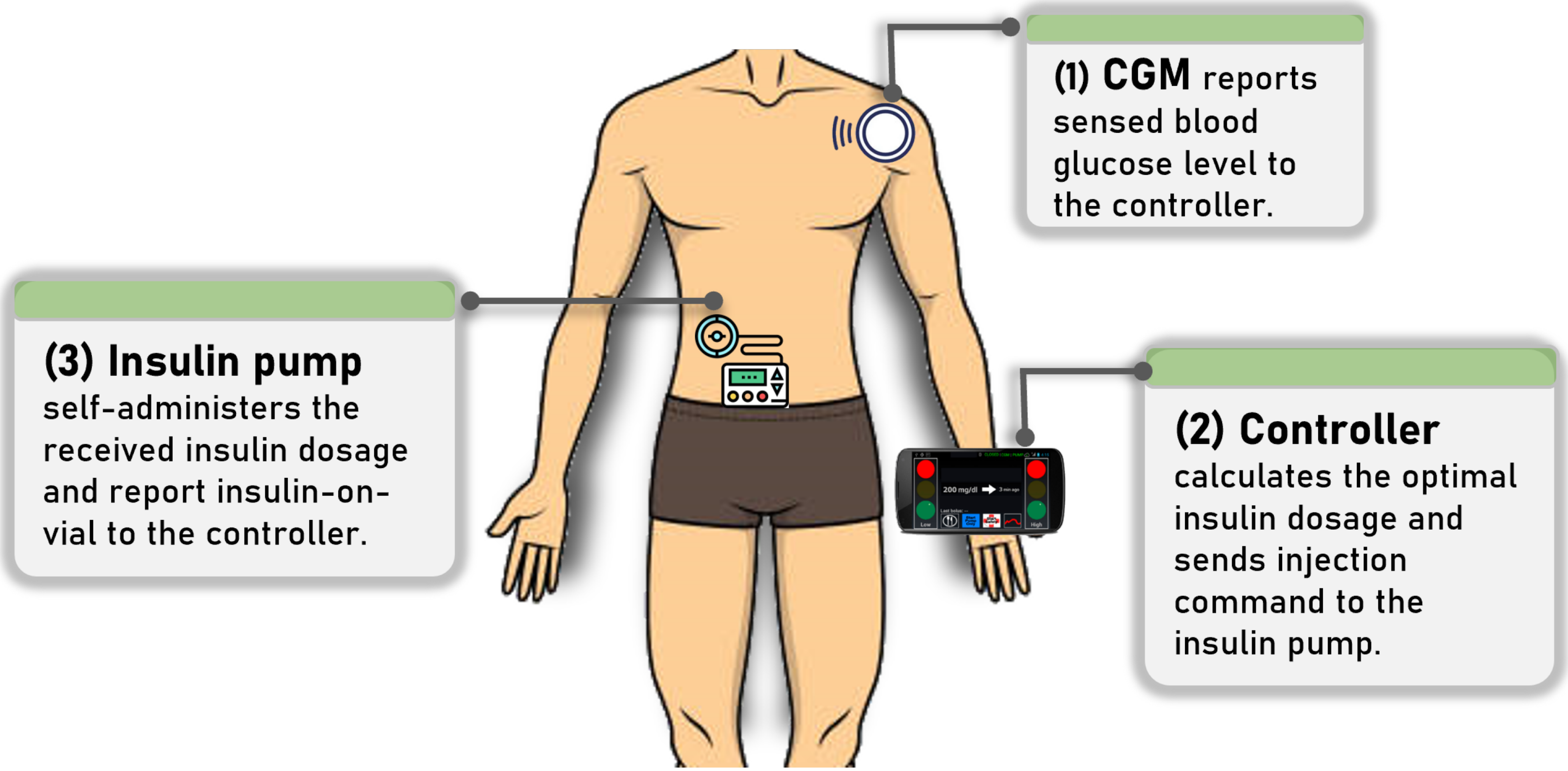




that save lives…

Why study control systems?

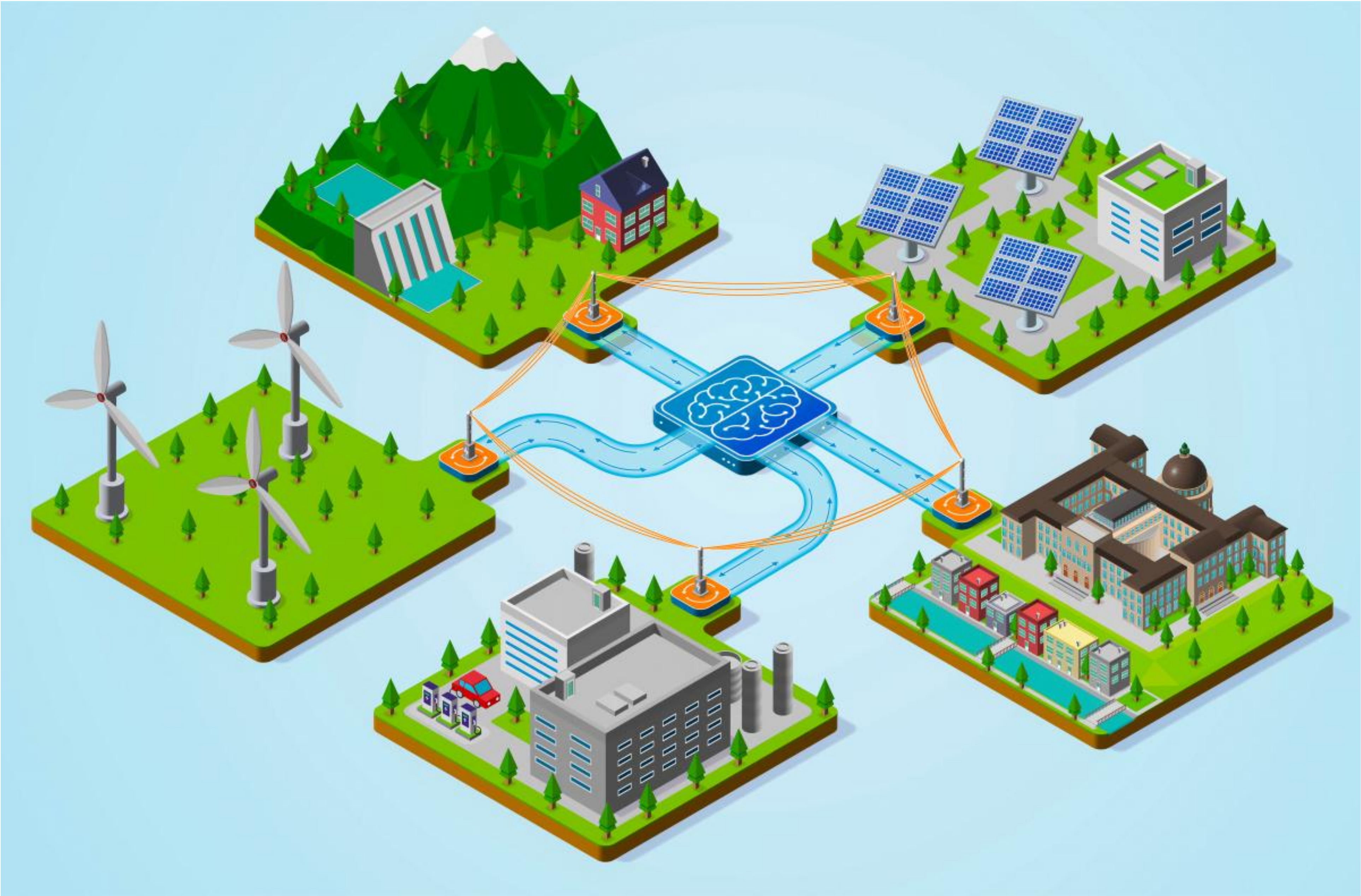
Biomedical engineering



How do you regulate the delicate balance of blood glucose in an artificial pancreas?
Robust control algorithms are the safeguard between a patient and life-threatening metabolic instability.

Why study control systems?

Controlling energy grids



Integrating renewable energy sources into the grid:

Renewable grids have no physical inertia to absorb load spikes.

We use control algorithms to mathematically introduce 'virtual' inertia, injecting power in milliseconds to catch the grid before it crashes.

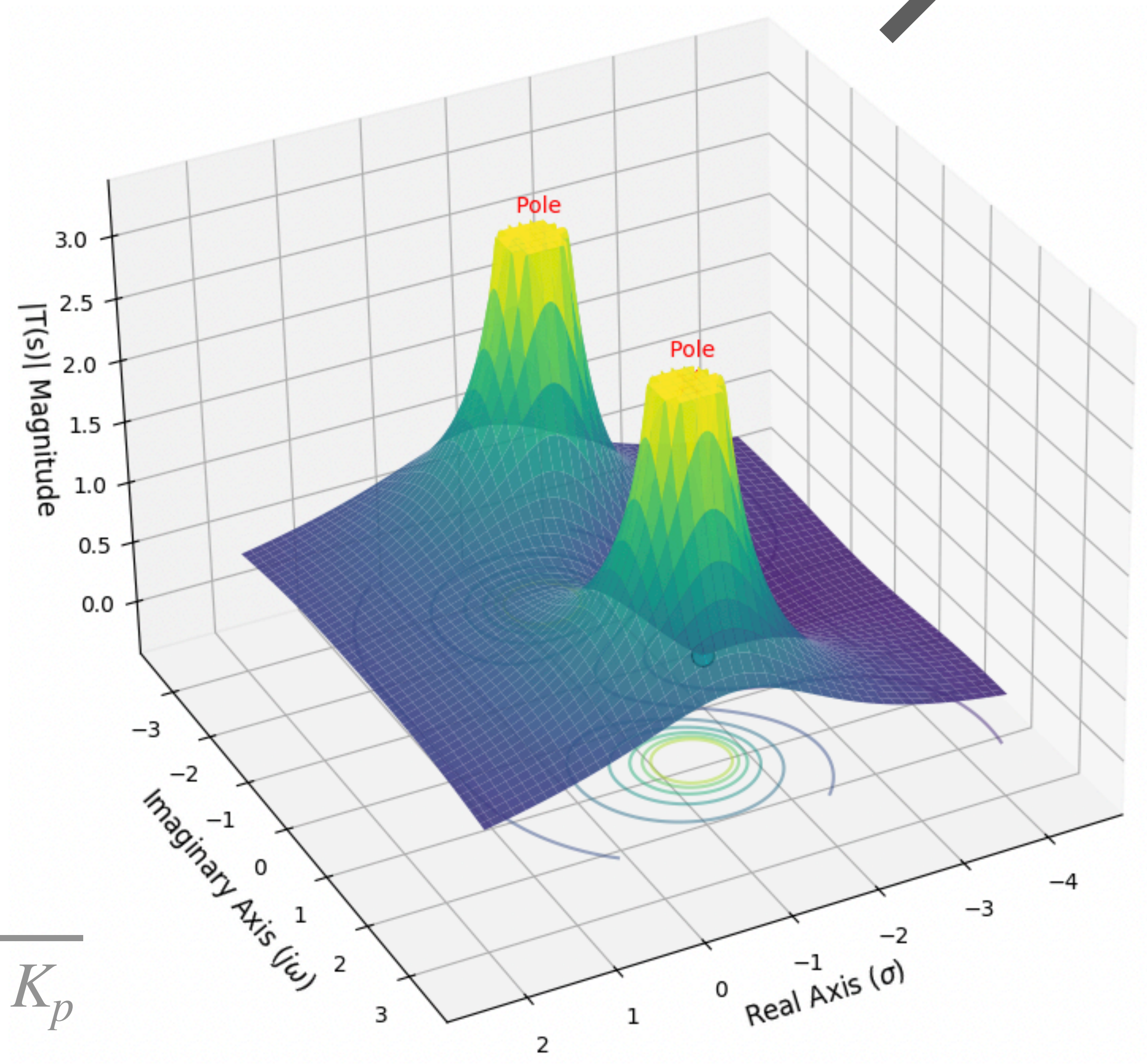
Why study control systems?

Because they are mathematically beautiful…

Complex real-world hardware

Specification of engineering design requirements

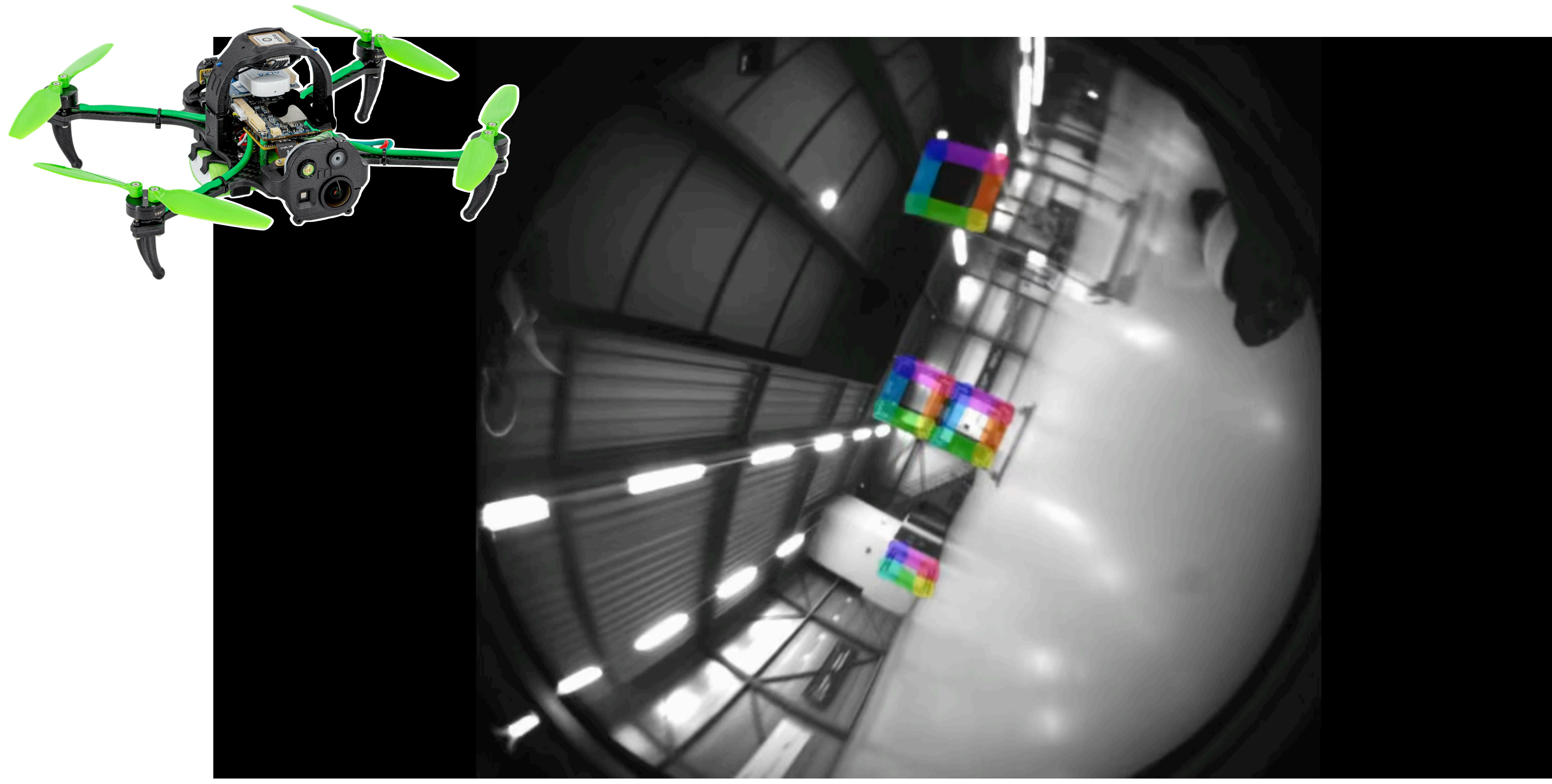
- Stability
- Transient response
- Control effort



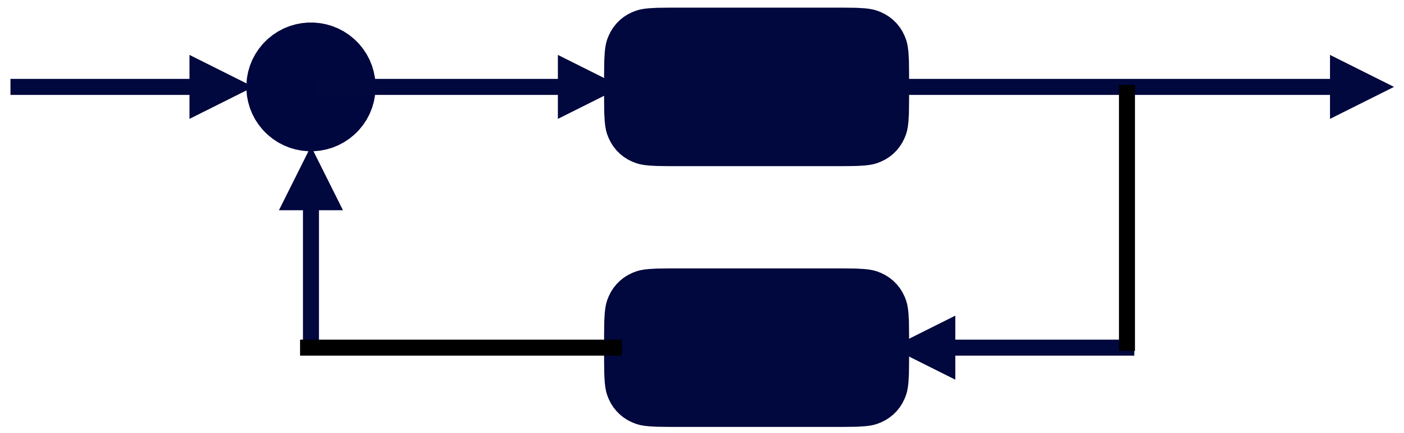
Abstract mathematical representation

$$T(s) = \frac{K_d s + K_p}{ms^2 + K_d s + K_p}$$

Performant real-world system



Why study control systems?



ELEC 341: Systems and Control

Mechanical engineering

Cars

Autonomous cars

VR headset tracking in gaming

Plasma stabilization in fusion reactors

Robotics

Controlling heating and cooling in your house

Electrical

This course: The common language (math foundations and engineering design principles) that connect all these applications.

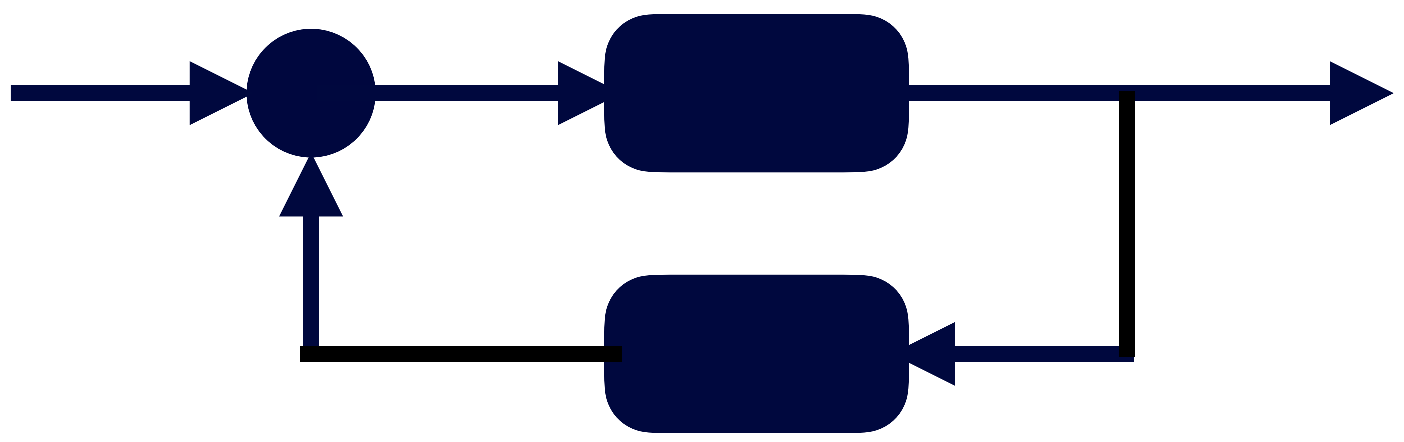
one

Biomedical engineering

Controlling heating and cooling in datacenters

Reinforcement learning (learning a controller)

Course learning objectives



ELEC 341: Systems and Control

Giving you the common language (math foundations and engineering design principles) that connect all these applications.

Understanding: feedback, stability, s-domain and frequency response analysis, system modeling & analysis, basic control design.

Course outline

Part 1

Mathematical tools

- Tools for modeling dynamic systems:
  - ODEs,
  - LTI systems,
  - Laplace transforms,
  - Transfer Functions,
  - Block diagrams

Part 2

Modeling control systems

- System responses
  - Impulse and step responses
  - first order systems,
  - second order systems,
- Stability analysis

Part 3

Introduction to control design

- Control objectives
  - Steady-state response
  - Performance metrics
- Control architectures
- Root locus diagrams
- Control design with the root locus method

Part 4

Frequency methods

- Frequency response
- Bode plots, gain and phase margins
- Nyquist stability criterion
- Designing controllers: lead-lag compensators
- Tuning PID controllers

Course logistics: evaluation

ELEC 341 201 2025W2, Systems and Control

Go to

join.iClicker.com

SRYF

- **10% - In class participation.**
    - IClicker Cloud polls and quizzes.
    - In-class tutorials and worked examples.

- **10% - Homework assignments.**

- **30% - Midterm exam. February 24th.**
    - Closed book. Pen, paper, calculator. Formula sheets will be provided.

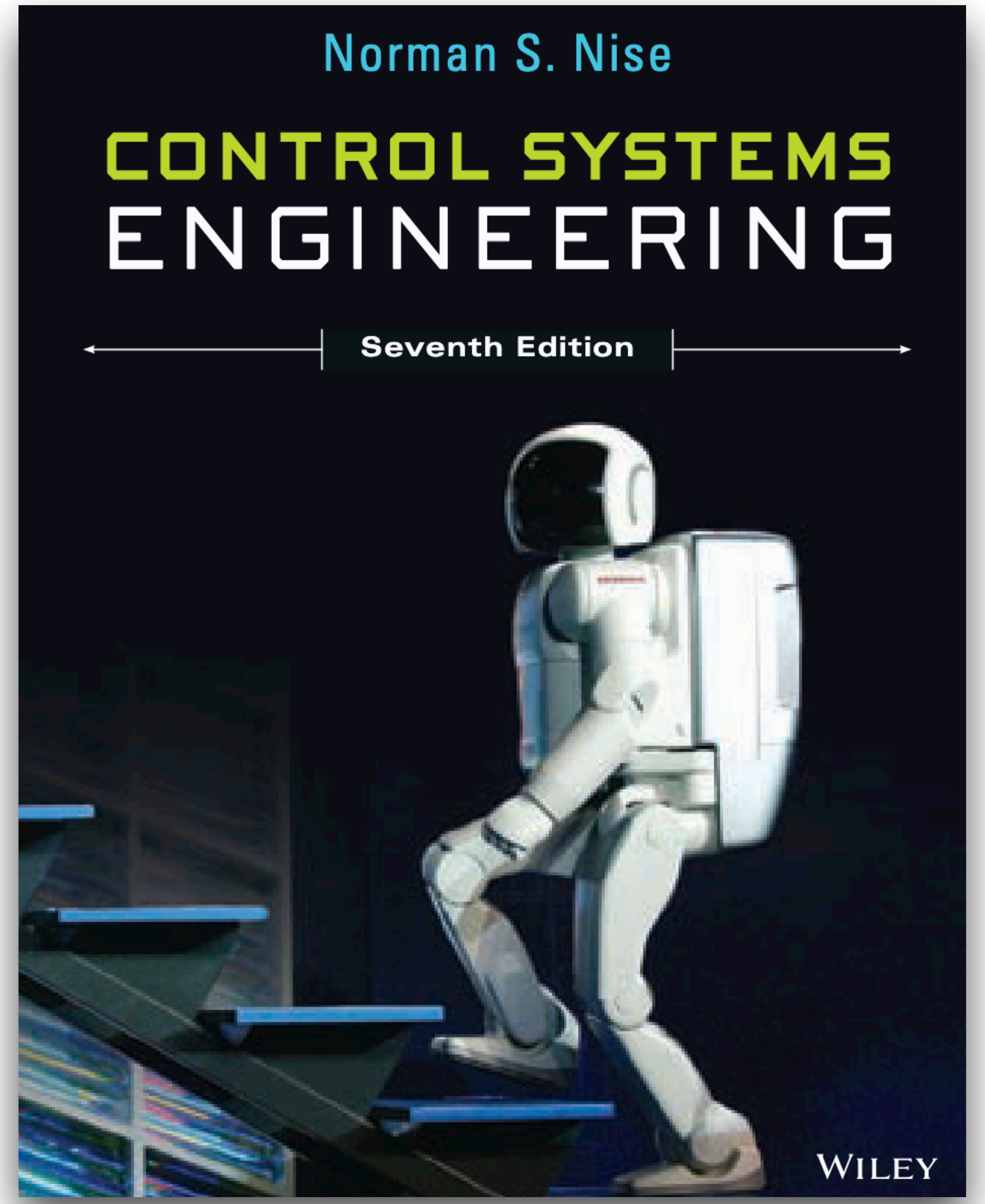
- **50% - Final exam.**
    - Closed book. Pen, paper, calculator. Formula sheets will be provided.

Course logistics: learning materials

Course content: Lectures
- Lecture notes will be provided as the course progresses.
- Lecture notes may not be entirely complete as provided. Attending class is necessary!!

Textbook: The following textbook is recommended as a useful second resource, but not strictly required.

- Norman S. Nise. Control Systems Engineering, 7th Edition. John Wiley & Sons; 2014.



Course logistics: the team of instructors

Instructor: Cyrus Neary

Office: Kaiser 4012

- Soon moving to Kaiser 3018.
- I will let you know in class and on Canvas when this happens.

Office hours: Thursdays after class.

- 15:00 - 16:00 PT in Kaiser 4012

Teaching assistant: Dilli Bhaskar

Office hours: TBD

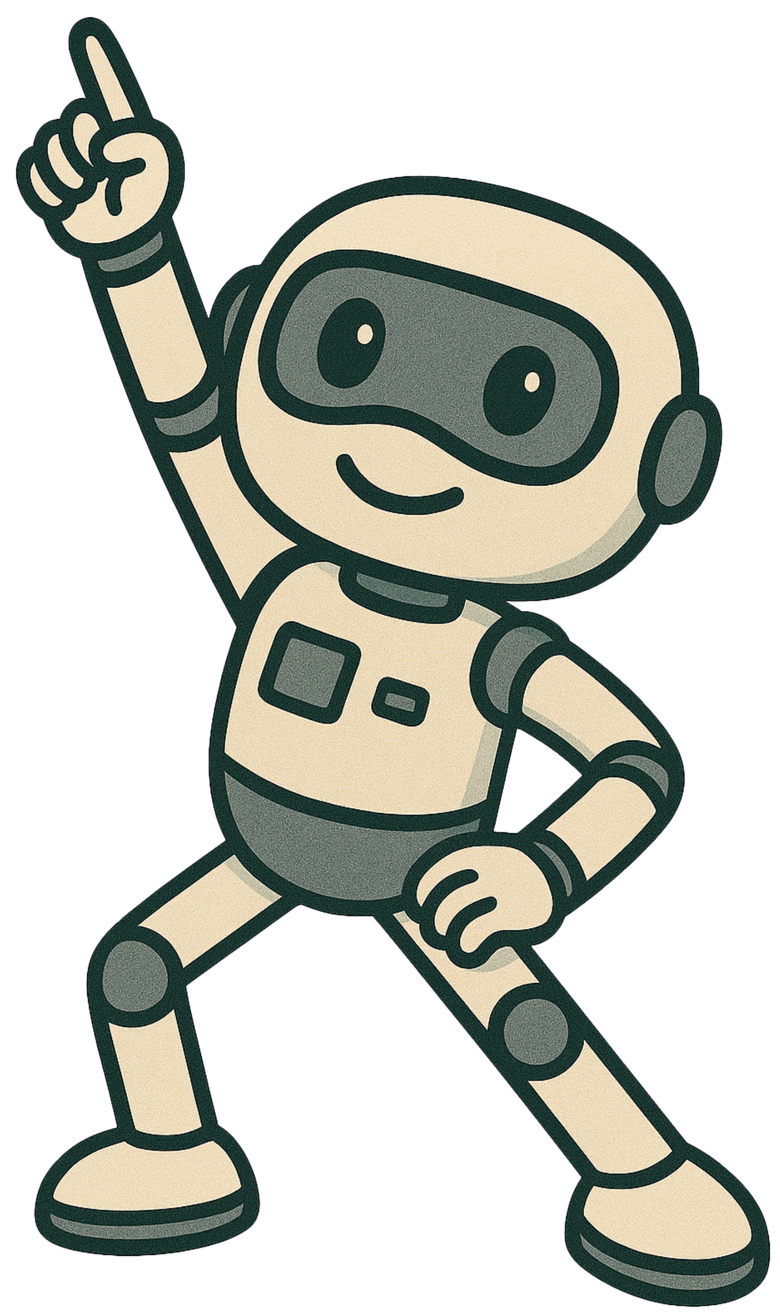
Teaching assistant: Ivan Cheung

Office hours: TBD

Contact: Please ask all course related questions in class, in office hours, or on Piazza (Piazza link is also in Canvas): https://piazza.com/ubc.ca/winterterm22026/elec_v3412012025w2

For confidential issues, or urgent concerns, please contact me directly by email: cyrus.neary@ubc.ca

Let’s get started!



What is a system? What is a control system?

System: Mathematically, it's a relationship between inputs u and outputs y.

$$\text{Input: } u(t) \longrightarrow \boxed{\text{System}} \longrightarrow y(t)\text{: Output}$$

Practically: Models some engineering system in the real world.

E.g. Drone: $u(t)$ — The current/voltage that we send to the motors

$y(t)$ — Motor speed, y velocity, height.

Car: $u(t)$ — The angle of gas pedal

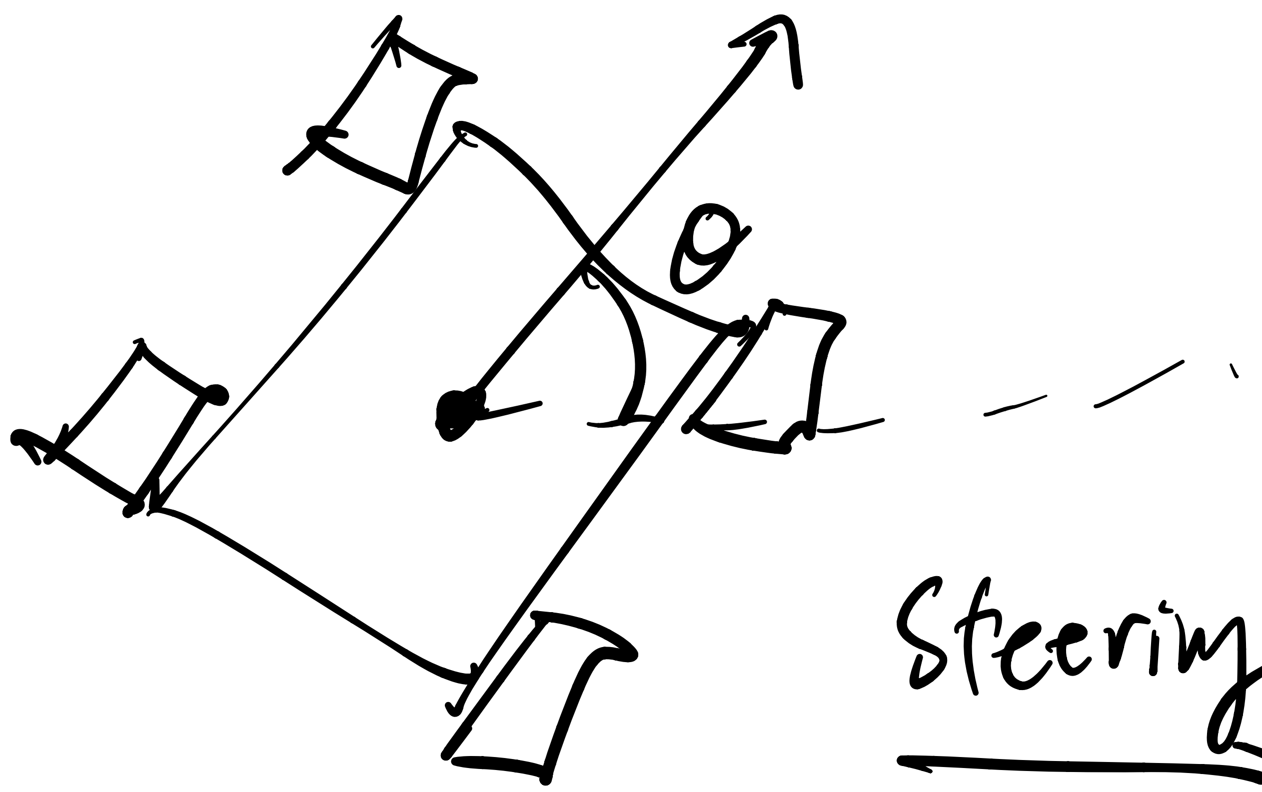
$y(t)$ — Acceleration, throttle position

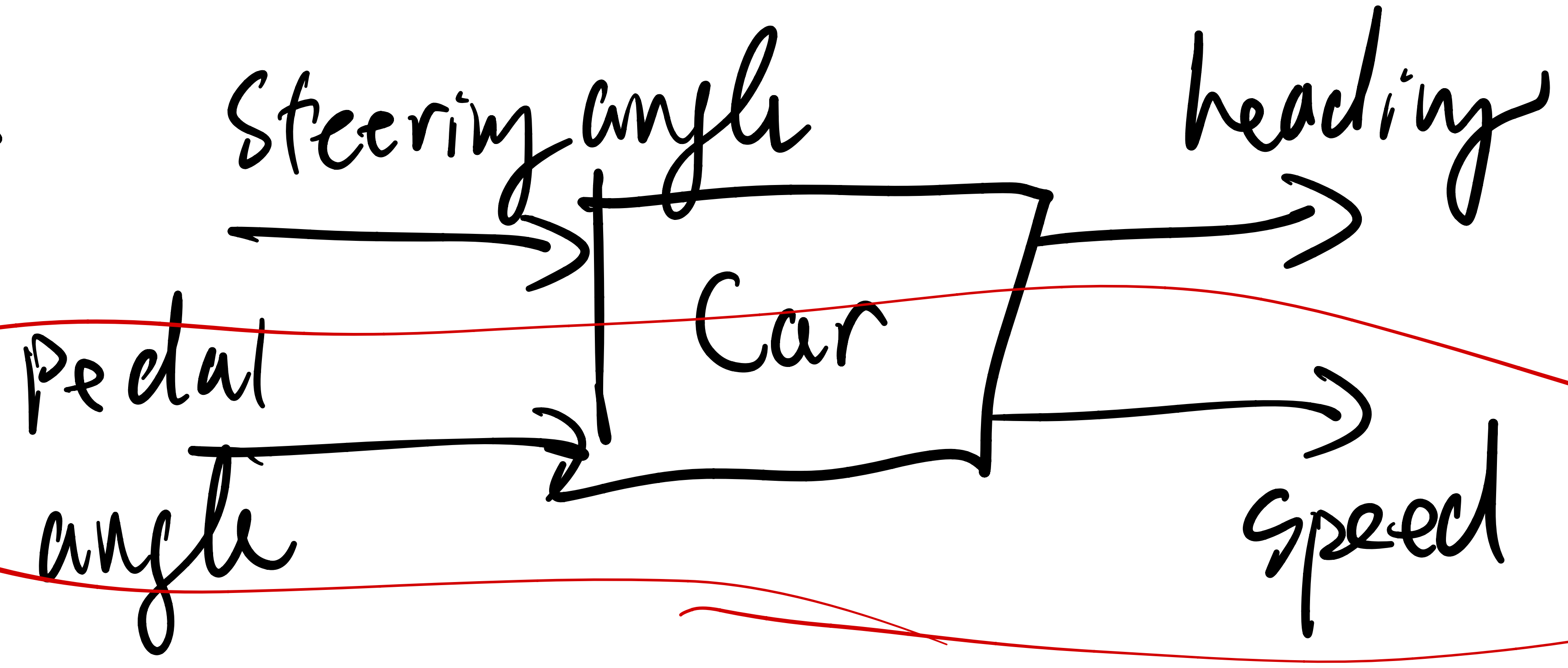
What is a system? What is a control system?

Control system: The goal of a control system is to "regulate its behavior", i.e. make the system do what we want.

↳ Choosing/designing the value of $u(t)$, $\forall t > 0$, to force the output $y(t)$ to follow a desired behavior.

Example



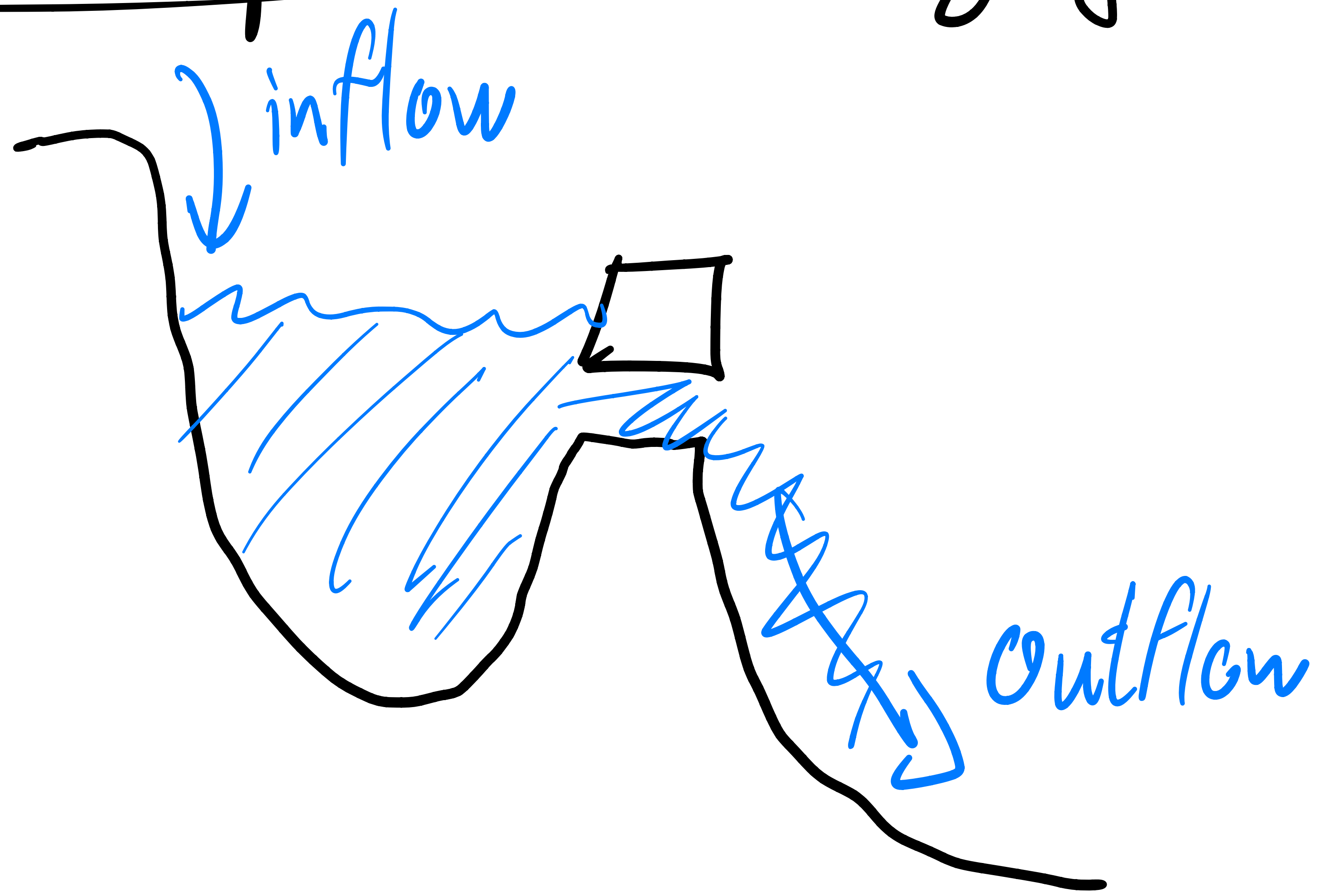


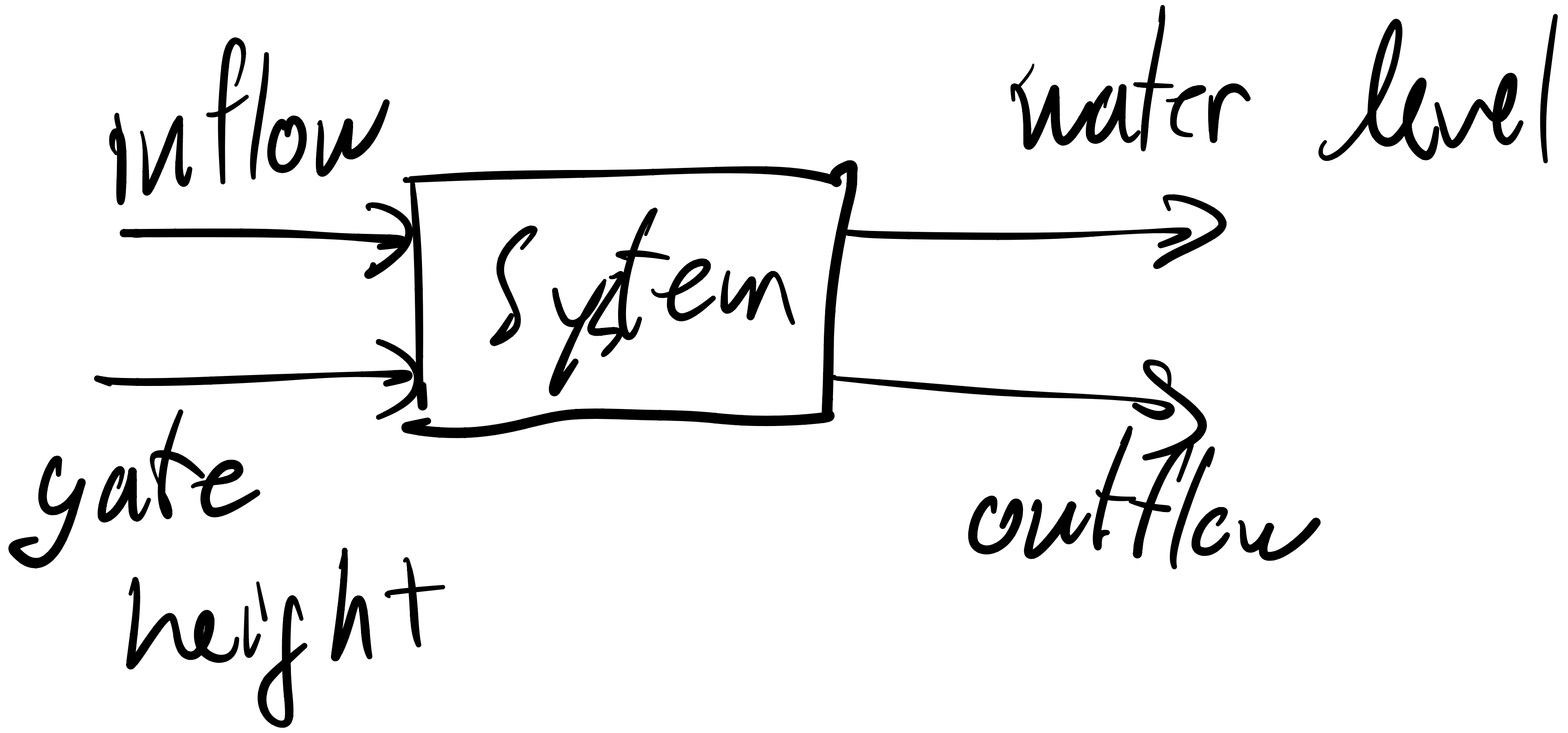
Objectives:

- Drive at consistent speed defined by pedal angle.
  ↳ More pedal ⇒ more speed
- Don't crash.

What is a system? What is a control system?

Example Managing water levels in a dam





Objectives:

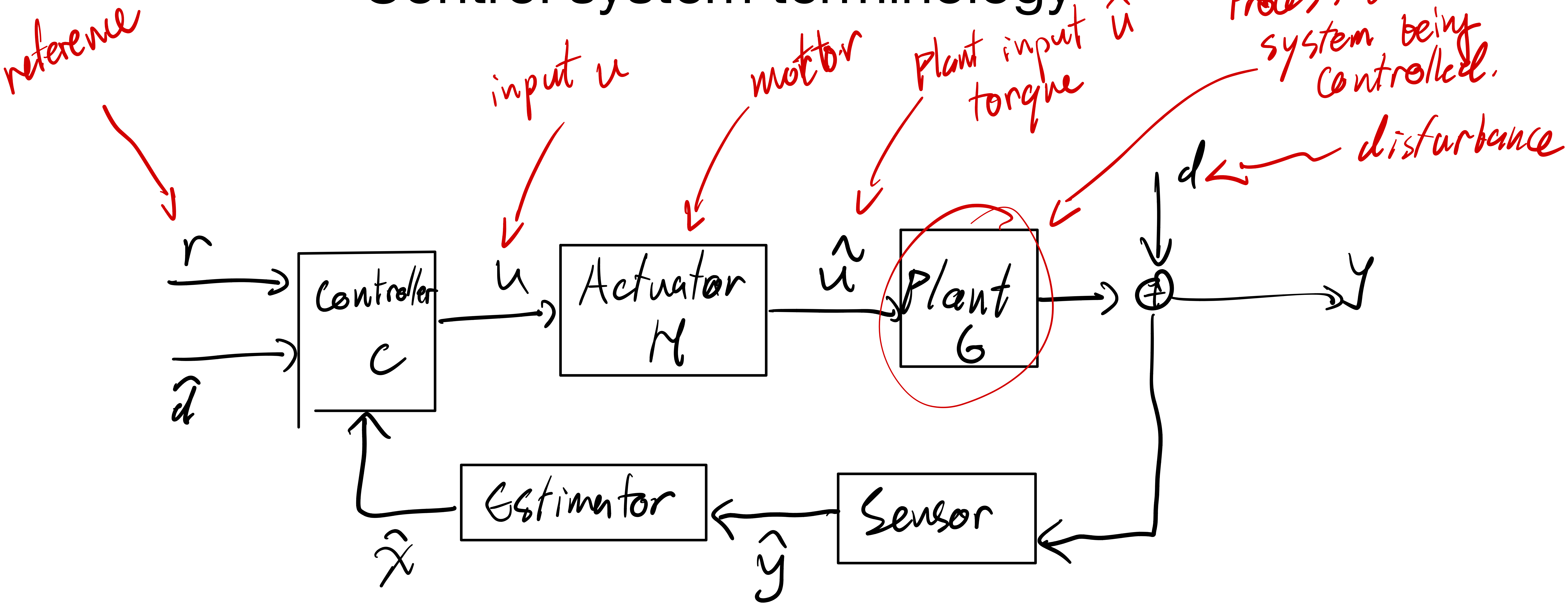
- Keep the water level constant
- Keep the water levels below a certain amount
- Maintain specific outflow for power generation.

What is a system? What is a control system?

Note: Control systems are very often safety critical

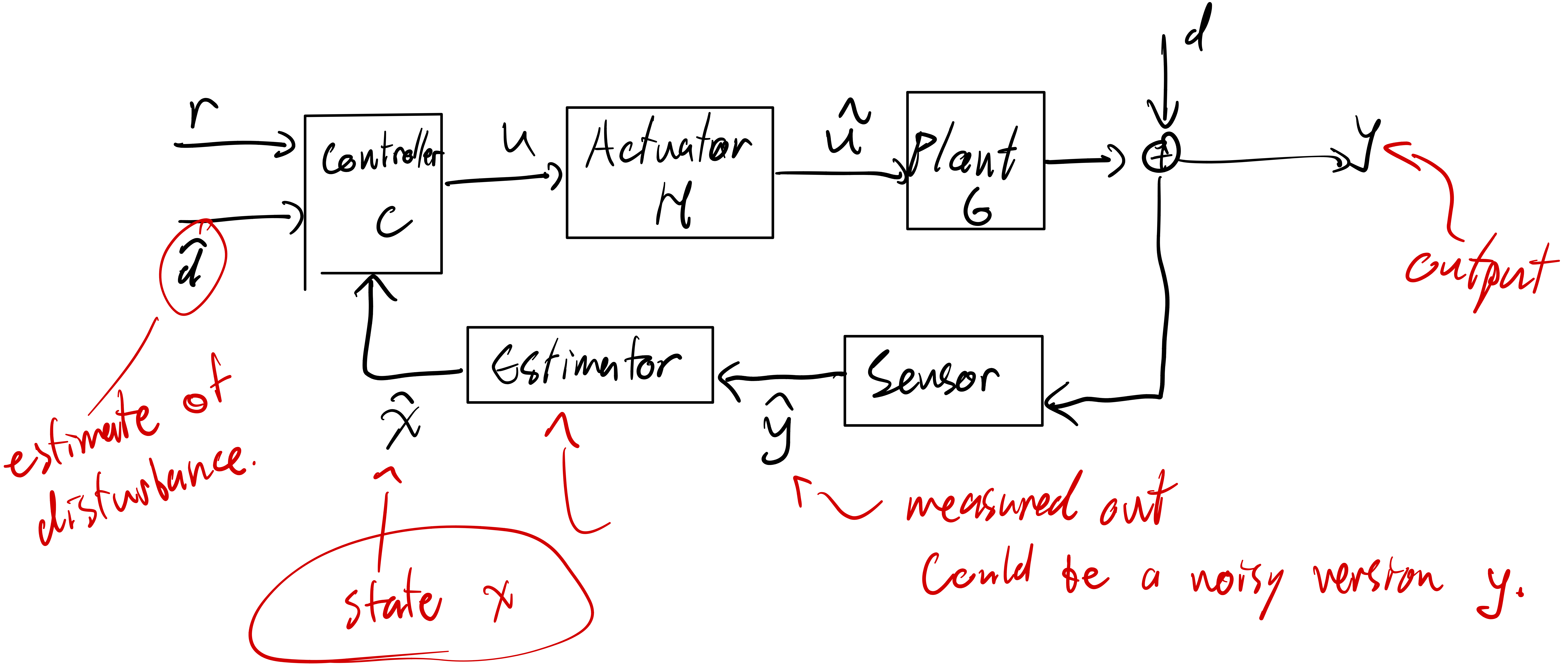
- If car controller fails $\longrightarrow$ car crashes
- If dam gets too full $\longrightarrow$ village below gets flooded
- Insulin pump controller fails $\longrightarrow$ Serious health risk.

Control system terminology

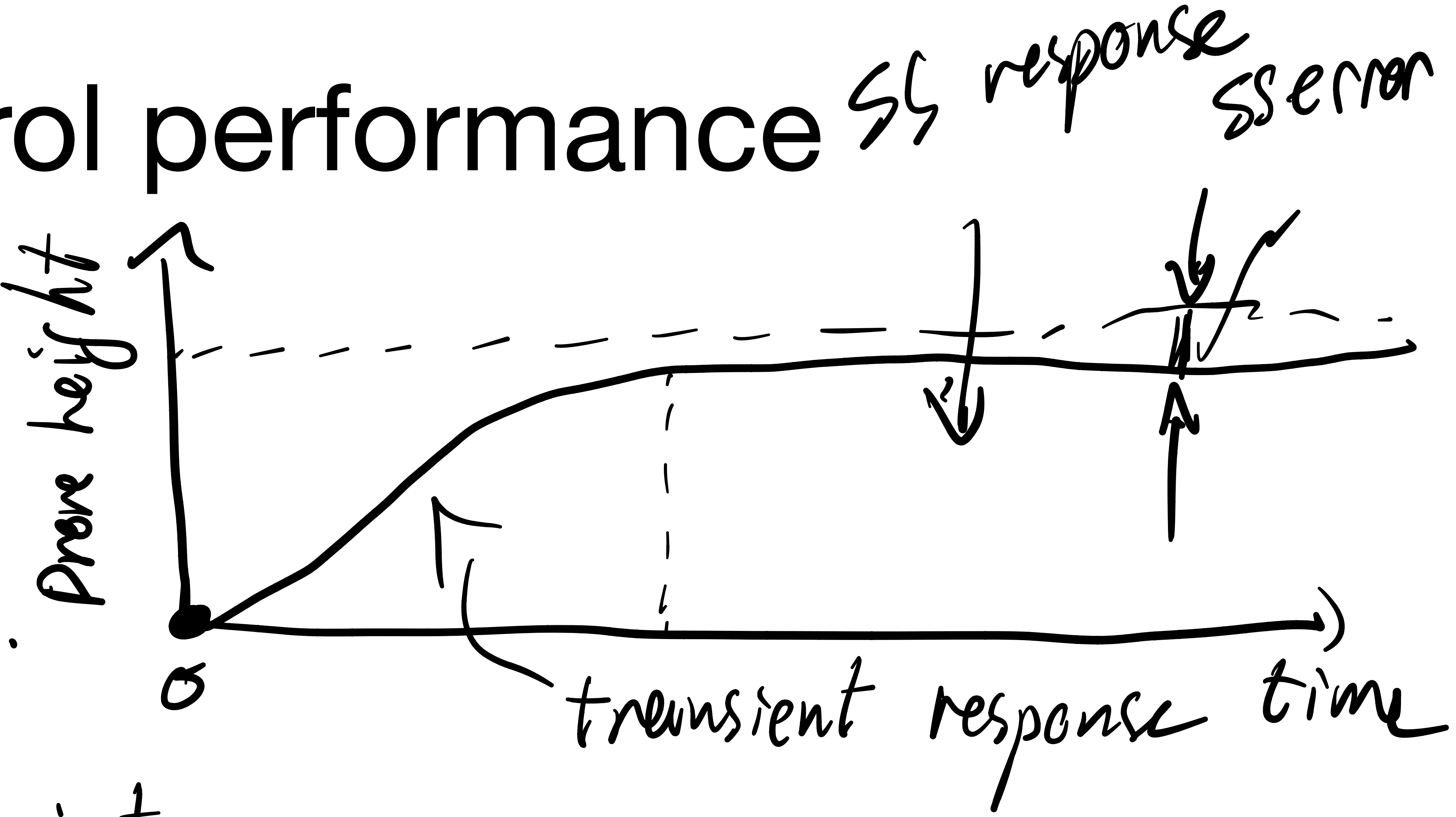


disturbance:
- Input that we can't control. wind.
- Often noisy/stochastic
- Something we did not account for in our model of the system/plant

Control system terminology



Important measures of control performance



Transient response: How does the system respond to a new input, before setting down to some steady state.

E.g. Elevator ride is smooth for passengers.

Steady state response: How close to our setpoint is the system output, after setting down.

E.g. the doors open and the floor is not aligned!

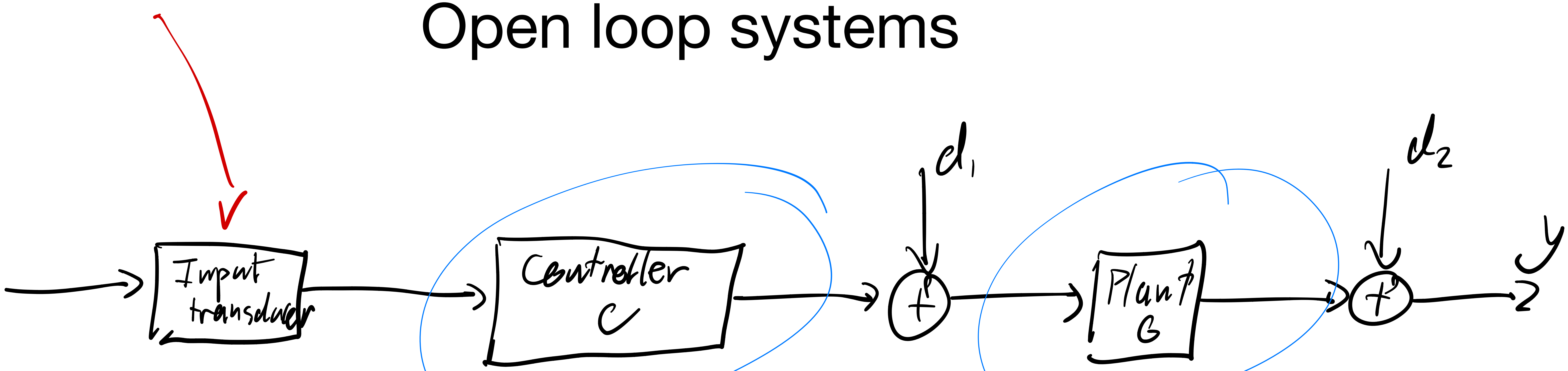
Stability: Will the response of the system remain "in our control"

→ If I give a bounded control input, does the output remain bounded as well?

Note: These are not the only measures of performance for control systems.

- Repeatability, energy use, tracking performance, robustness.

Open loop systems

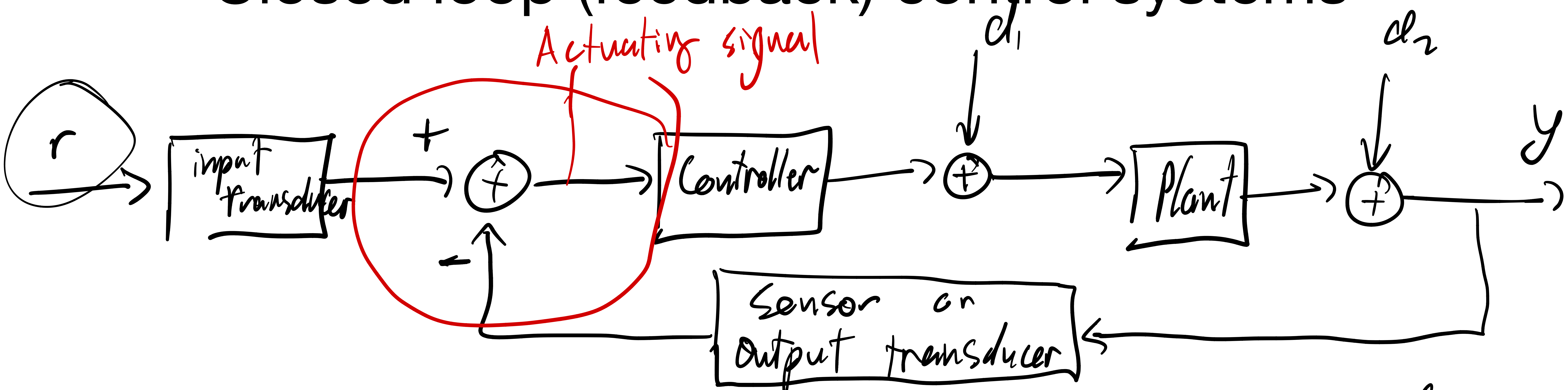


open loop system: No feedback loop! Simply follow the reference, without having an idea of the current output/state.

Example: Toaster oven

- output y – how toasty?
- Toaster doesn't measure this output ⟶ it's open loop.
- Can't account for disturbances.
  ↳ e.g. I changed my bread.

Closed loop (feedback) control systems



- The actuating signal is the difference between the desired reference and the actual measured output.
- Compensates for disturbances by measuring output, comparing it to ~~input~~ reference
  - ↳ Any difference between reference and output ⟶ actuating signal drive controller
  - ↳ If no difference ⟶ Nothing happens, :)

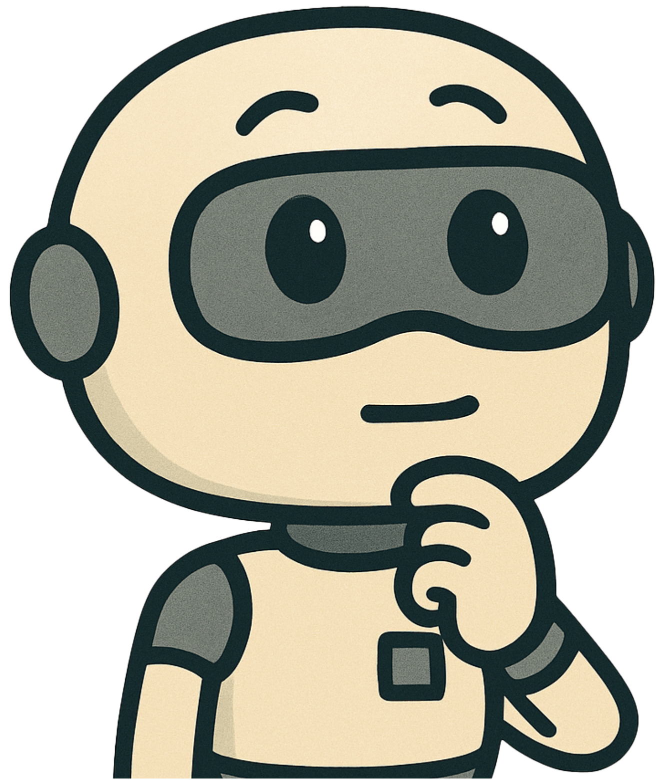
What are the strengths/weaknesses of open-loop/closed loop control?

Open loop

- Stength: Very simple
- Cons: Can't account for disturbances

Closed loop

- Strength: Can account for disturbances
- Con: Complex. We need sensors.



Manual vs. automatic control

Manual control: A human operator/pilot/driver adjusts the input to the plant.

Automatic control: A computer or machine that adjusts the inputs.

↳ fast and efficient.

↳ More sensitive. Can go beyond human capabilities.

↳ Safety.

Note: Many systems are hybrids of these definitions.

An overview of the control systems design process

what steps does the control engineer typically take?

Parts 1–3 of course

Step 1 Define the problem, Articulate the goals/objectives and the performance specifications.

↳ From abstract descriptions of the problem and hardware.

Step 2 Draw block diagrams. Specify how subsystems interconnect. How does information flow?

Step 3 Create a mathematical model of the whole control system.

Parts 3 and 4

Step 4 Iteratively analyze, design, and test to verify/validate that the specifications are met.