Sensing, Actuation, Control

ENES 100

Prof. Bruce Jacob
Electrical & Computer Engineering
(with enormous thanks to Prof. Bill Levine)

OUTLINE:

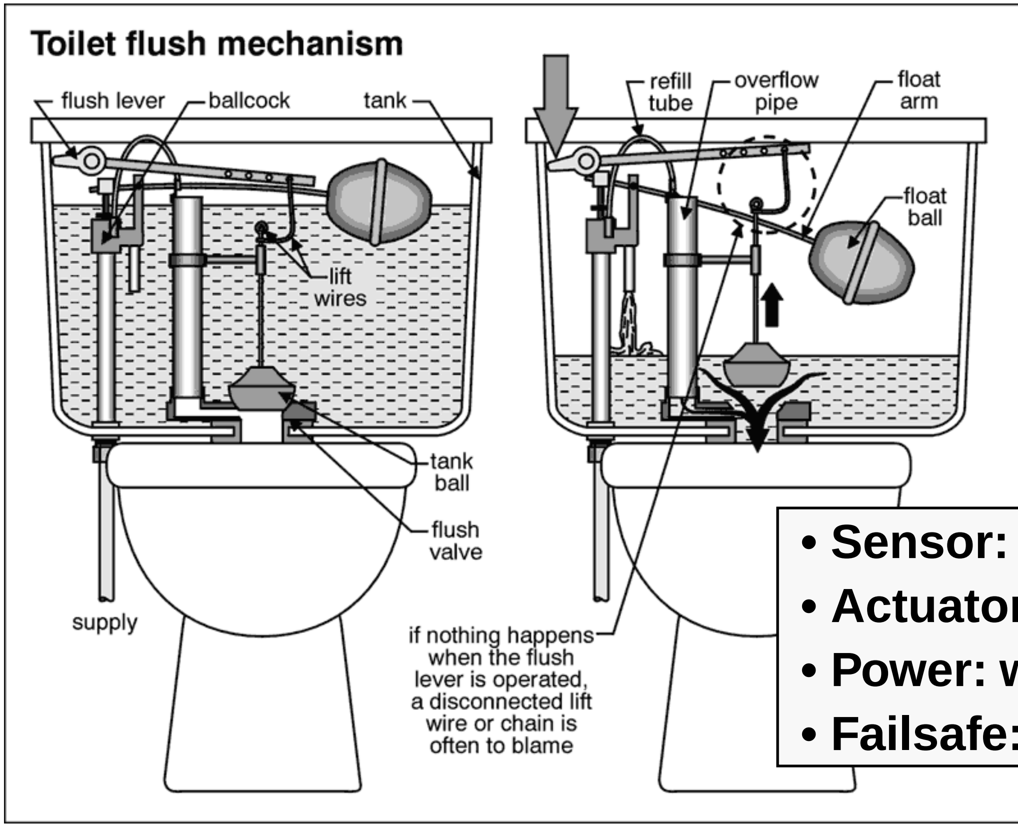
- **Some example control systems**
- **Feedback: Open loop vs. closed loop** (PID control)
- **Simple hovercraft circuits**
- **Hovercraft control issues**

The Toilet

- Sensor: float
- Actuator: valve
- Power: water level
- Failsafe: overflow tube

- **2000 year old control system**
- **System not used for present purpose until 19th century (cholera epidemics)**

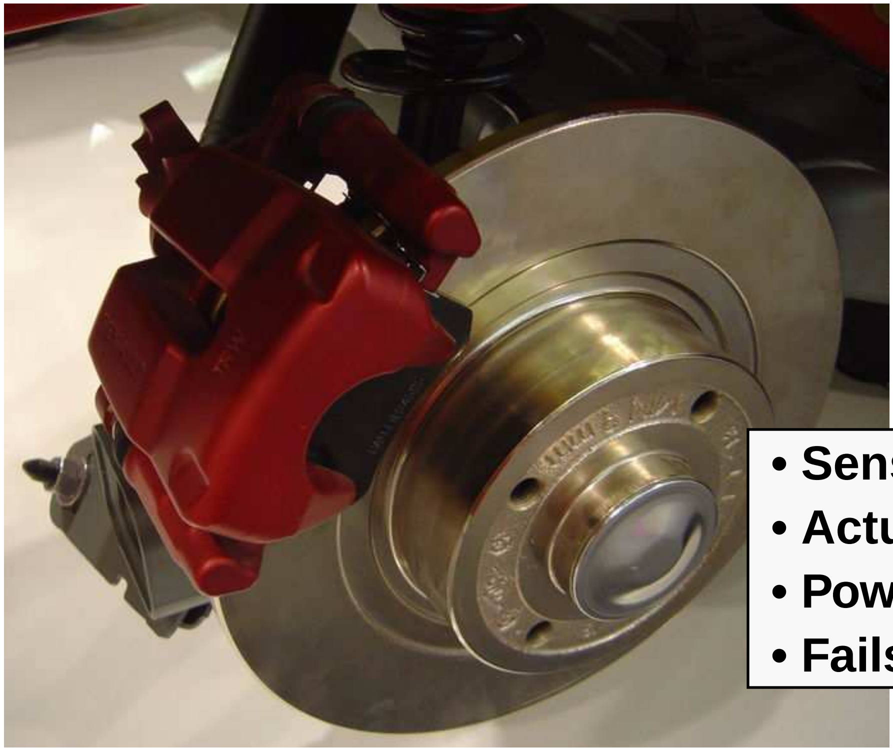
The Toilet



- Sensor: float
- Actuator: valve
- Power: water level
- Failsafe: overflow tube

- 2000 year old control system
- System not used for present purpose until 19th century (cholera epidemics)

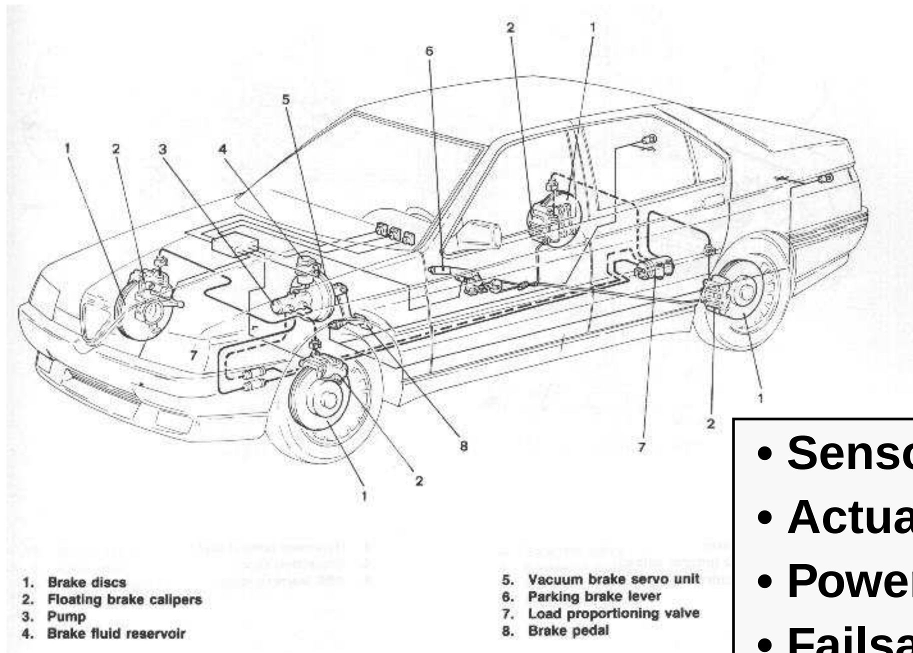
Power Brakes (e.g. disk)



- Sensor: foot pedal
- Actuator: brake calipers
- Power: hydraulic
- Failsafe: dual system

- **Manual activation**
- **Separate hydraulic networks (per brake or per opposite pair)**
- **Additional failsafe (optional): power needed to hold brake open (fails closed)**

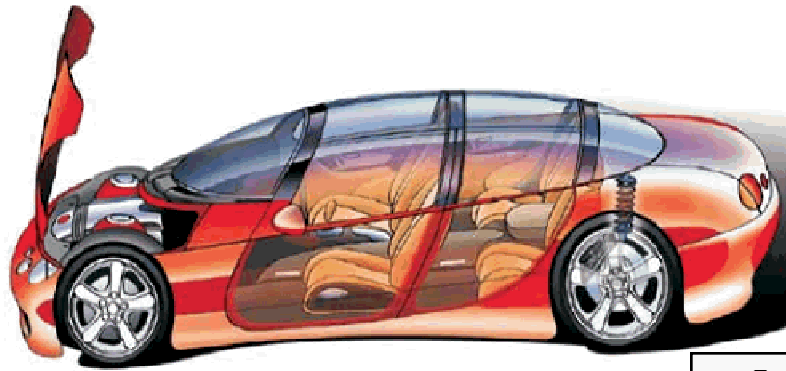
Power Brakes (e.g. disk)



- **Sensor: foot pedal**
- **Actuator: brake calipers**
- **Power: hydraulic**
- **Failsafe: dual system**

- **Manual activation**
- **Separate hydraulic networks (per brake or per opposite pair)**
- **Additional failsafe (optional): power needed to hold brake open (fails closed)**

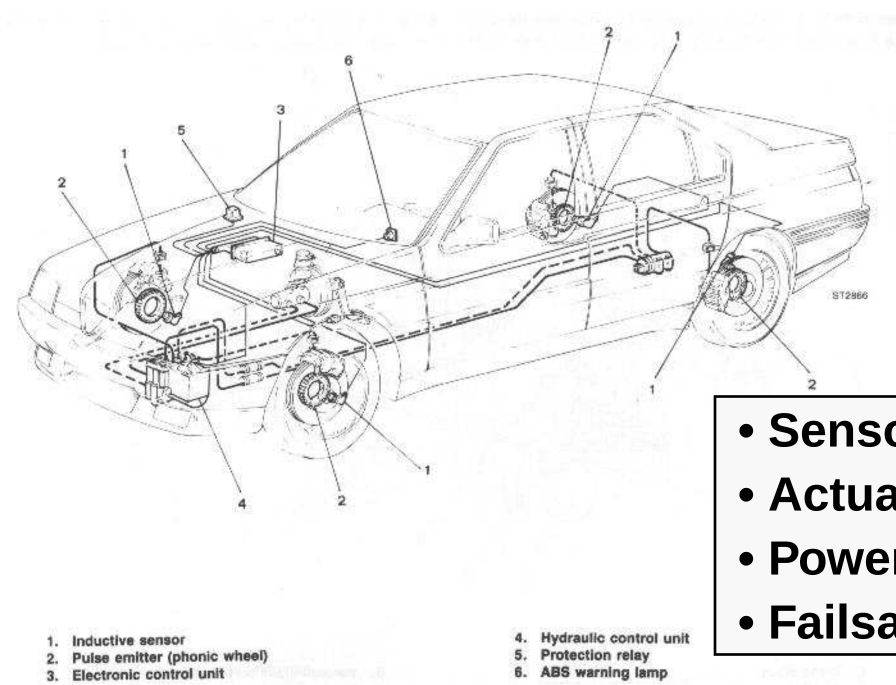
Antilock Brakes



- Sensor: wheel speed
- Actuator: pulse emitter
- Power: hydraulic
- Failsafe: manual, sensors

- **Each wheel monitored separately for significant deviation in wheel speed**
- **Each wheel controlled/pulsed separately**
- **Problem: contaminated sensors**
- **Add'l sensors: wheel angle & gyroscope**

Antilock Brakes



- Sensor: wheel speed
- Actuator: pulse emitter
- Power: hydraulic
- Failsafe: manual, sensors

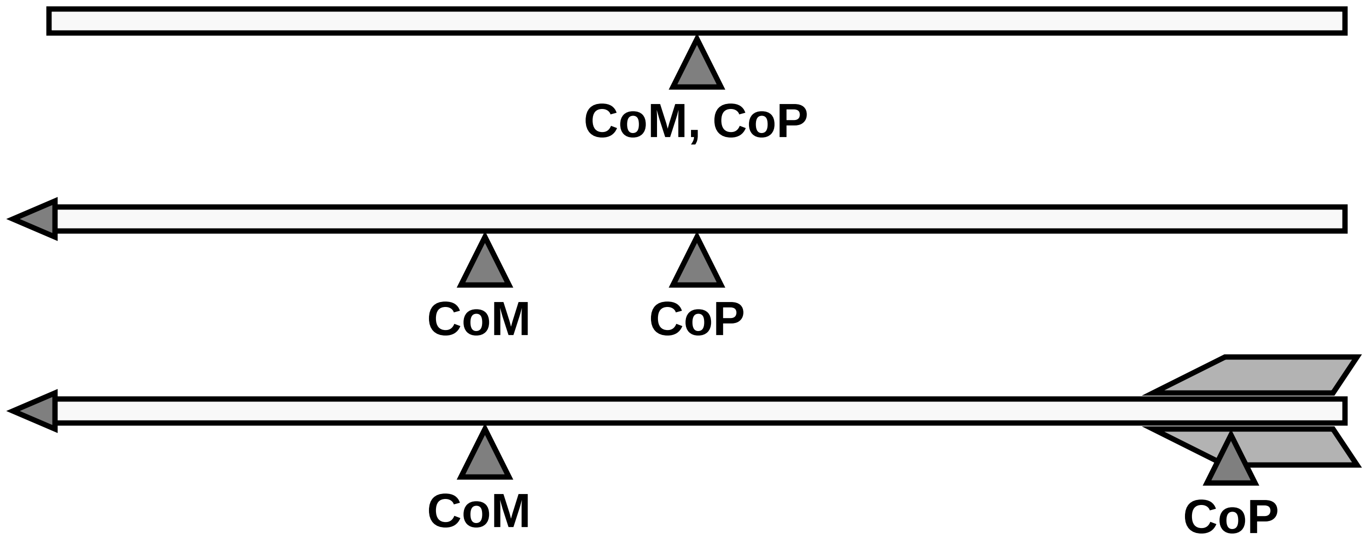
- **Each wheel monitored separately for significant deviation in wheel speed**
- **Each wheel controlled/pulsed separately**
- **Problem: contaminated sensors**
- **Add'l sensors: wheel angle & gyroscope**

Fletched Arrow

- Sensor: fletching
- Actuator: fletching
- Power: pressure
- Failsafe: n/a

- **Bare shaft: completely unstable**
- **Weighted tip: slightly more stable**
- **Fletching acts as control mechanism (correction proportional to deviation)**

Fletched Arrow



- Sensor: fletching
- Actuator: fletching
- Power: pressure
- Failsafe: n/a

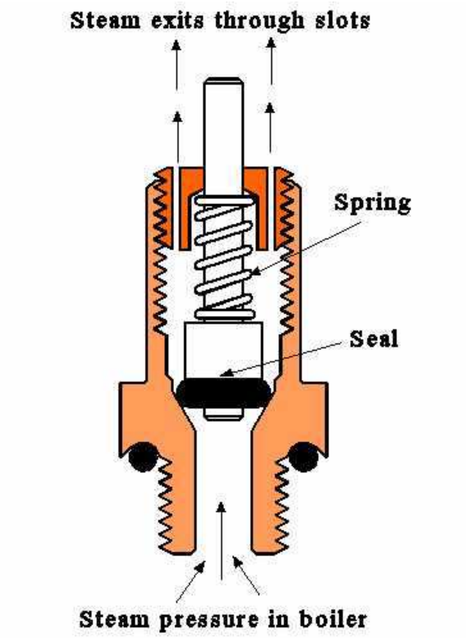
- **Bare shaft: completely unstable**
- **Weighted tip: slightly more stable**
- **Fletching acts as control mechanism (correction proportional to deviation)**

Steam Valve

- Sensor: spring-loaded piston
- Actuator: valve
- Power: (steam) pressure
- Failsafe: backup/none

- **Plug/spring acts as control mechanism (correction proportional to deviation: higher pressure => valve opens more)**

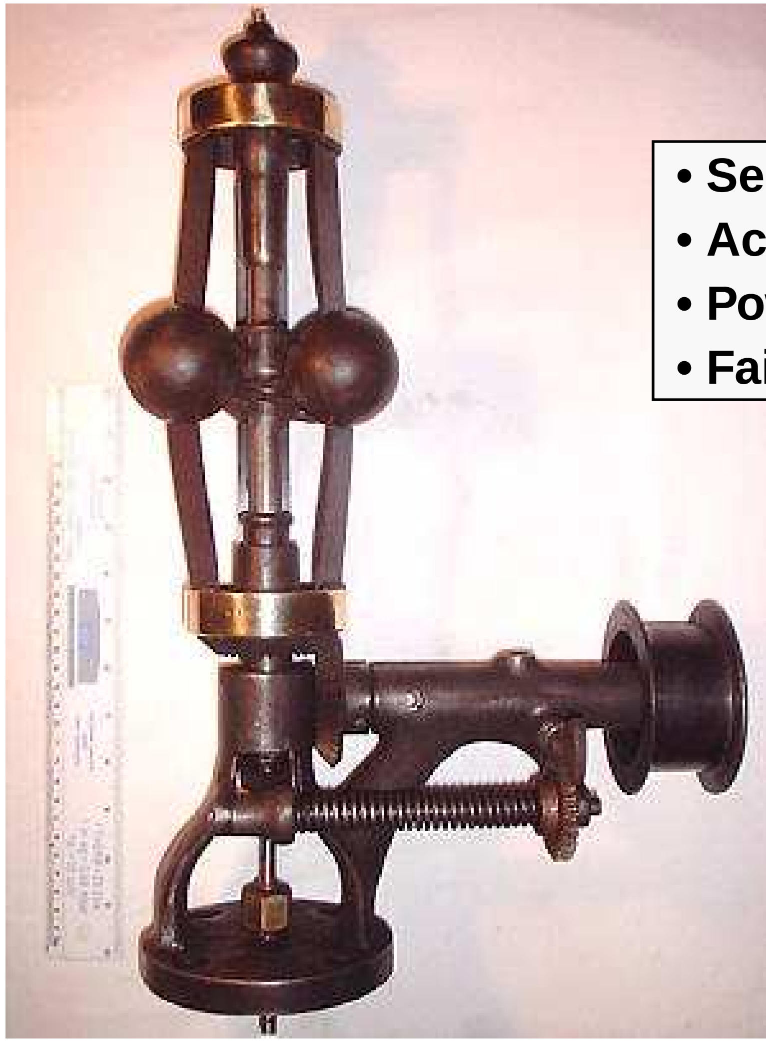
Steam Valve



- Sensor: spring-loaded piston
- Actuator: valve
- Power: (steam) pressure
- Failsafe: backup/none

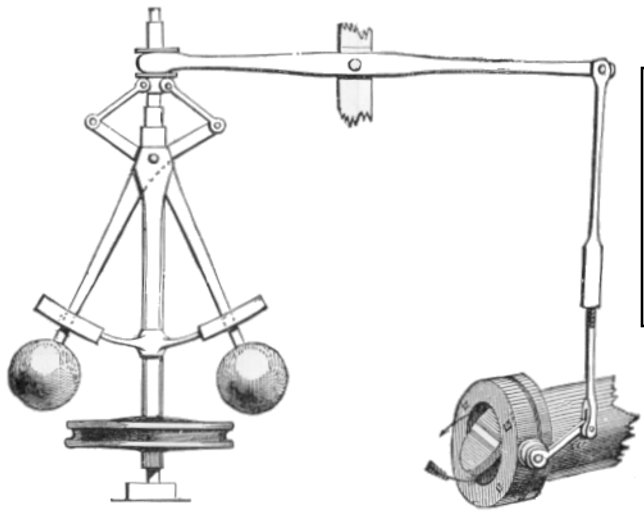
- **Plug/spring acts as control mechanism (correction proportional to deviation: higher pressure => valve opens more)**

Centrifugal Governor



- Sensor: centrifugal pendulum
- Actuator: valve
- Power: torque on shaft
- Failsafe: backup/none

Centrifugal Governor

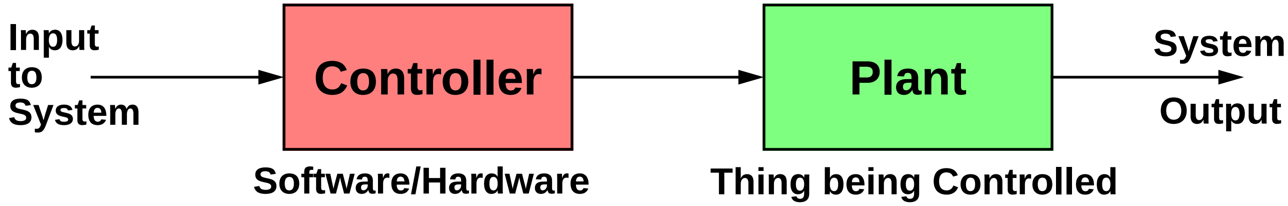


- Sensor: centrifugal pendulum
- Actuator: valve
- Power: torque on shaft
- Failsafe: backup/none

- **Also called the "flyball" governor**
- **Proportional control: the faster the rotation, the more the valve closes**
- **On nearly every steam engine made**

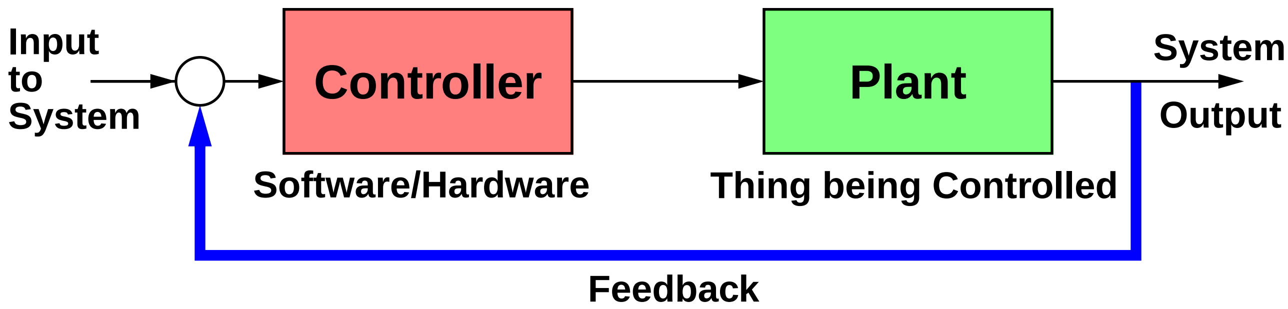
Feedback Control

OPEN LOOP



- Power Brakes
- Power-Assist Steering
- Manual Throttle

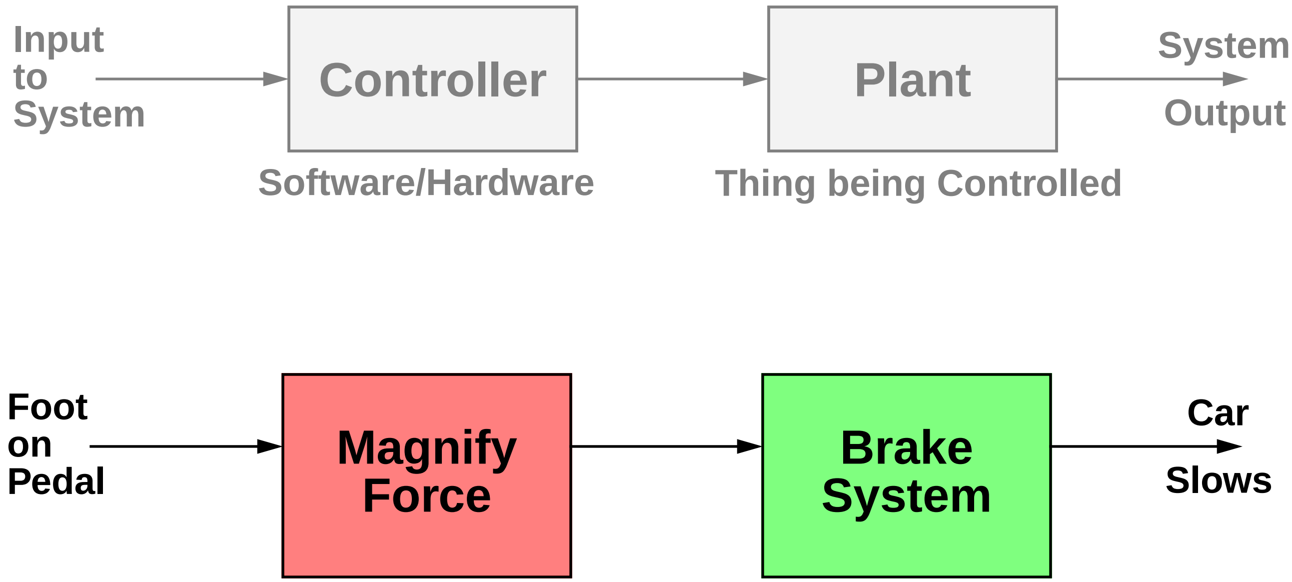
CLOSED LOOP



- Anti-Lock Brakes
- Compass-Assisted Steering
- Governor-Controlled Throttle

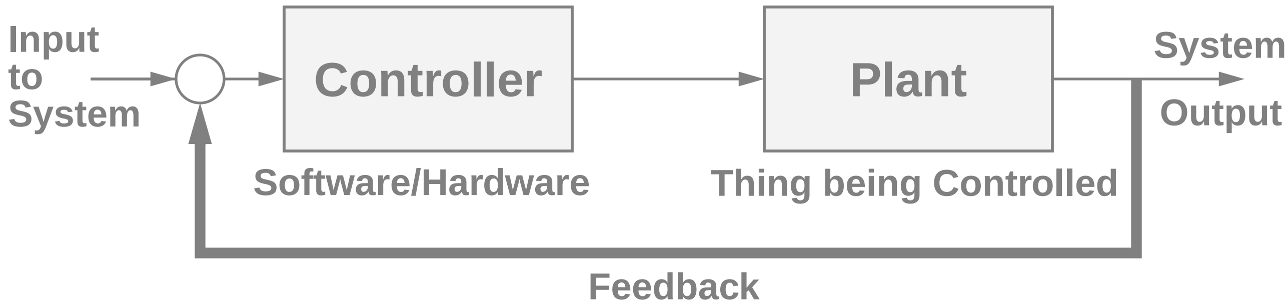
Power Brakes

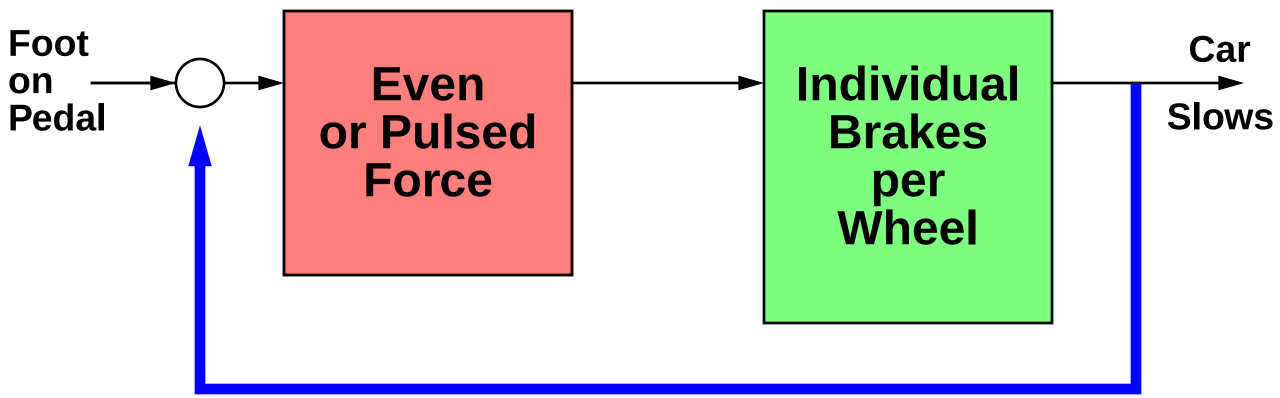
OPEN LOOP



Anti-Lock Brakes

CLOSED LOOP



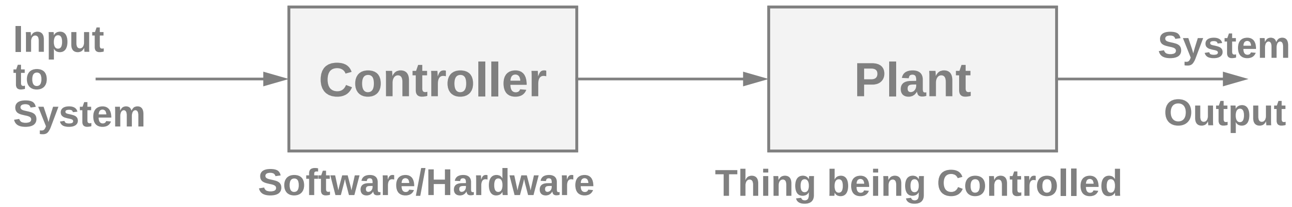


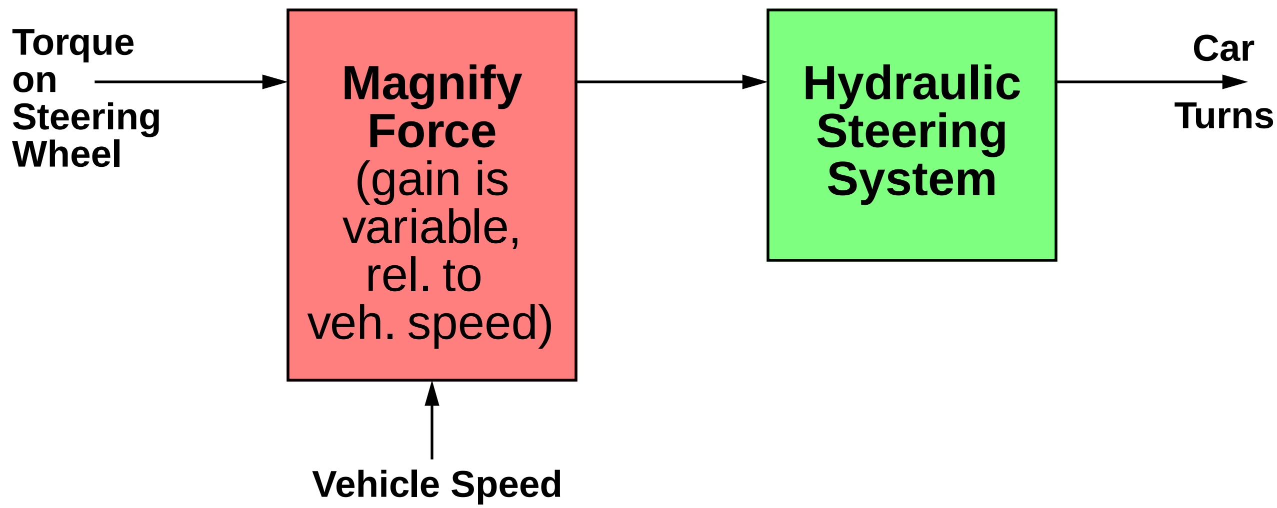
Feedback: Wheel rotation speed

Are any wheels slipping?
Are any wheels rotating much more slowly than the others?
If so, slipping brakes are pulsed to try to recover traction.

Power-Assist Steering

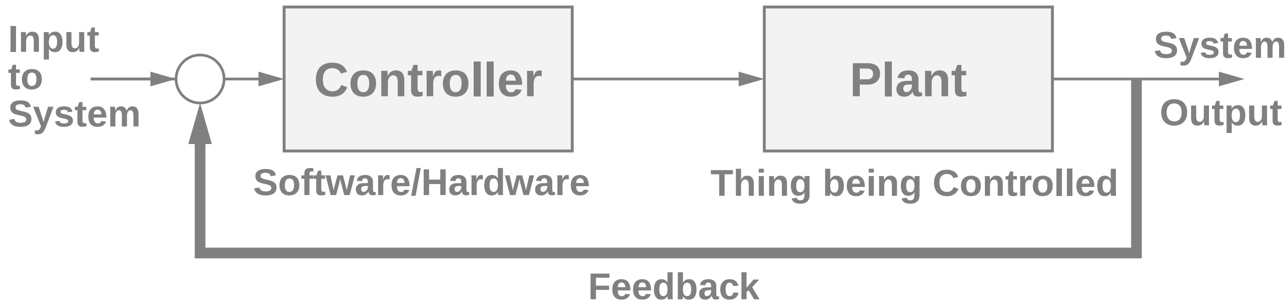
OPEN LOOP

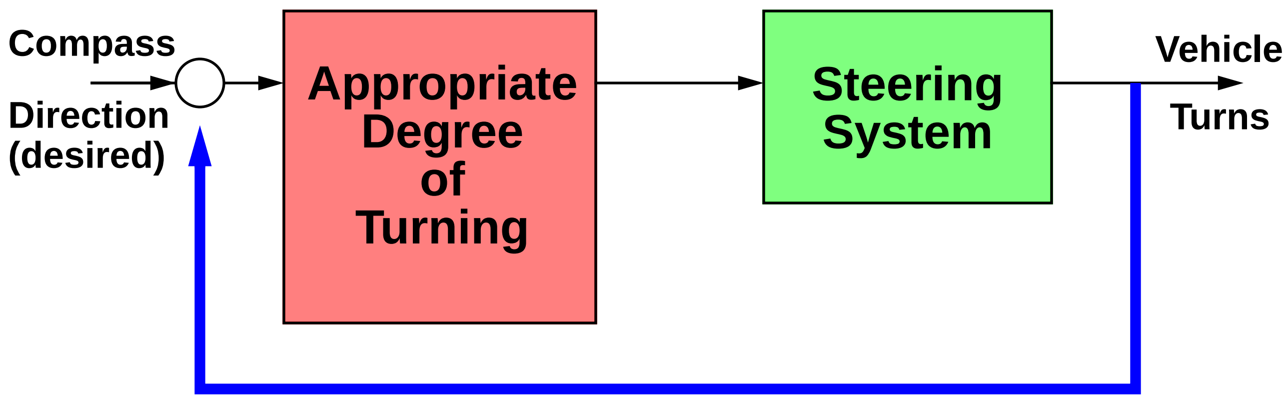




Compass-Assisted Steering

CLOSED LOOP





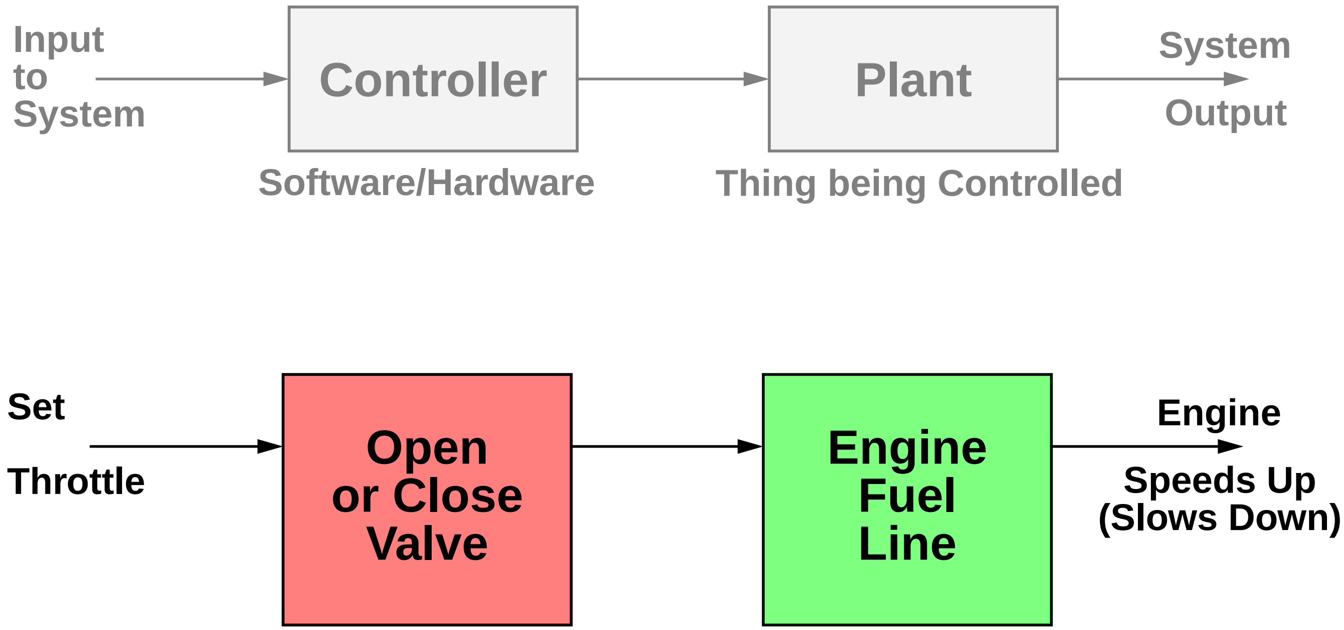
Feedback: Current vehicle direction of travel

If direction of vehicle is not equal to the desired compass point, control system adjusts steering appropriately

(note: vehicle can point one way and go another)

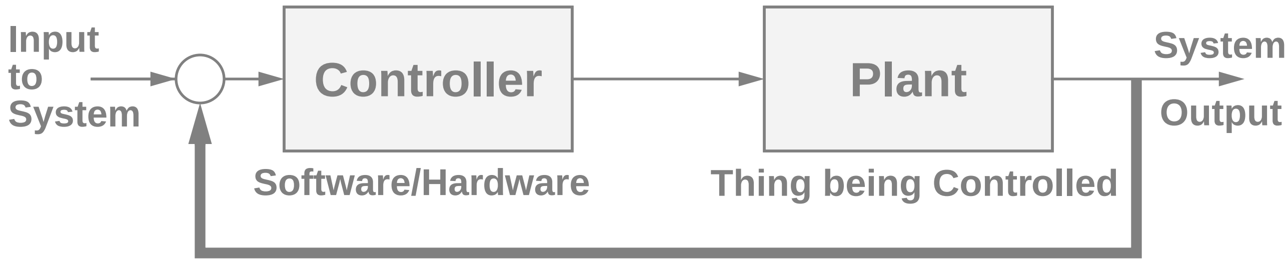
Manual Throttle

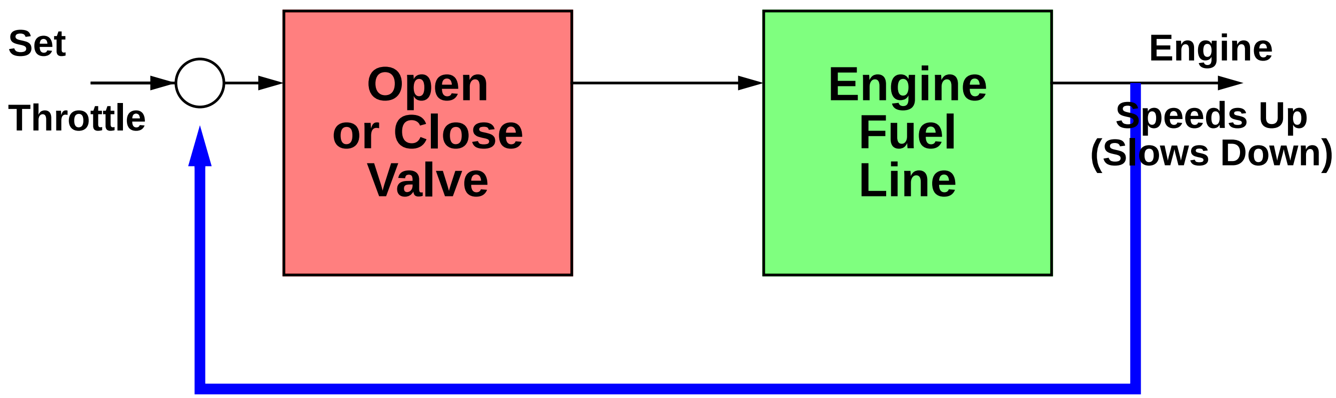
OPEN LOOP



Governor-Controlled Throttle

CLOSED LOOP



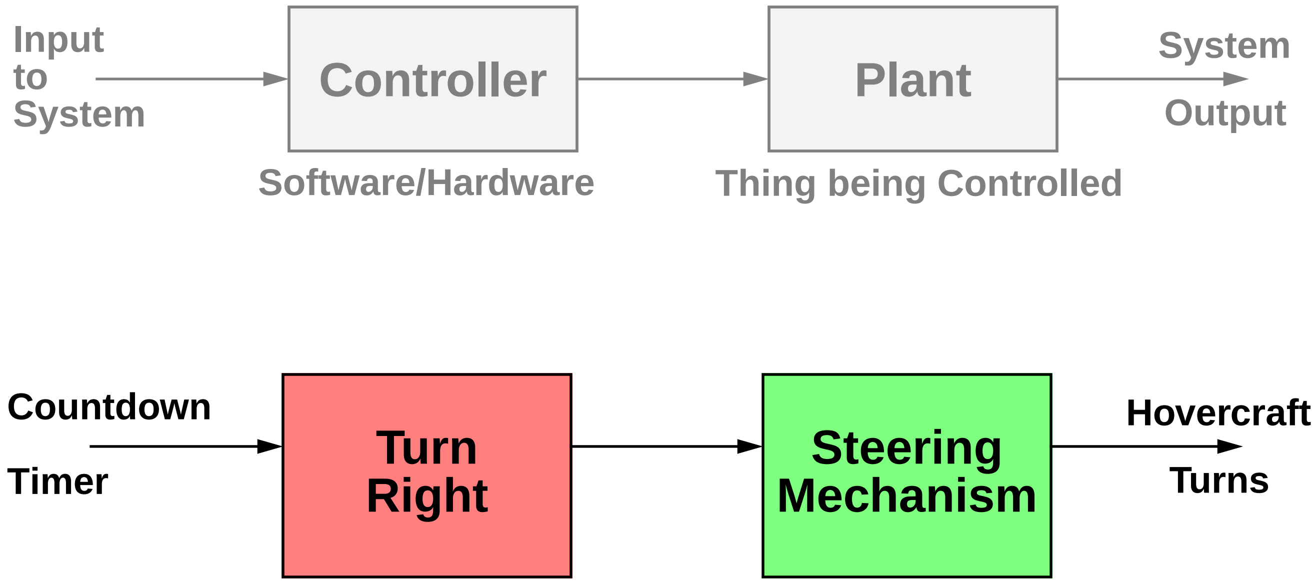


Feedback: Engine shaft rotation speed

If load on engine increases, the rotation slows, causing the governor to open the throttle

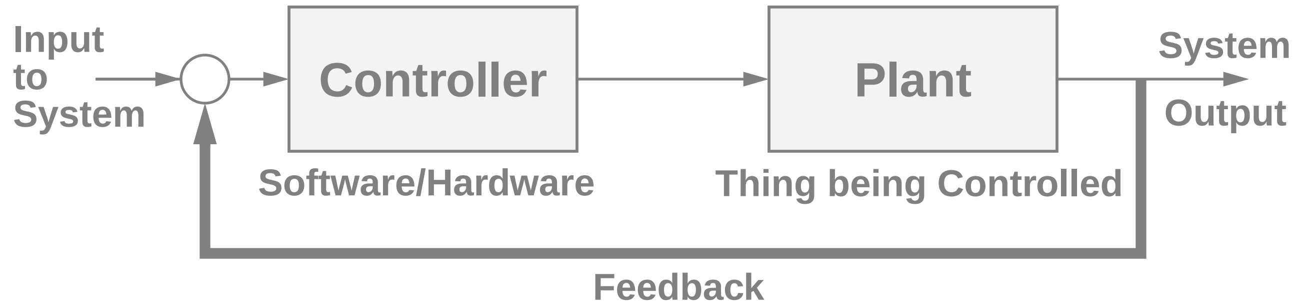
Hovercraft A

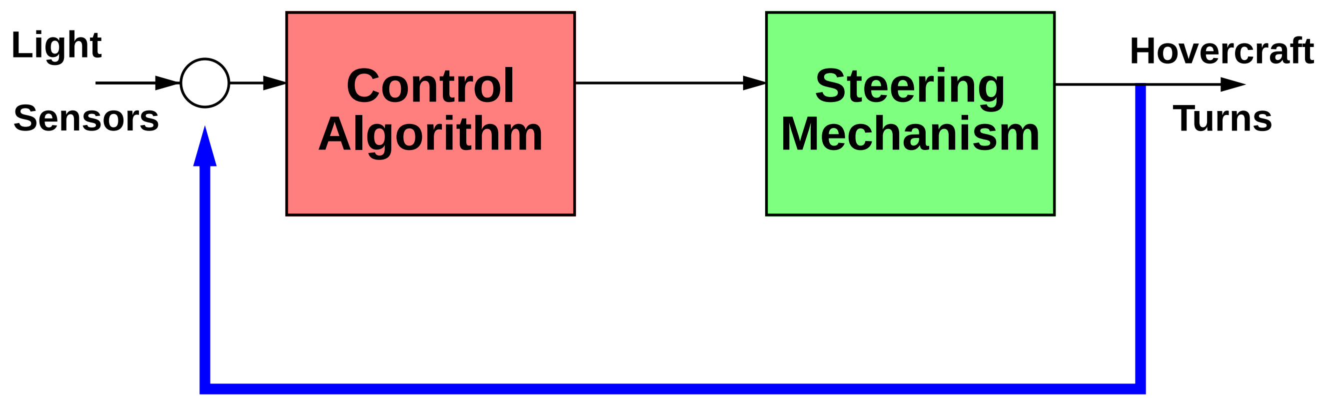
OPEN LOOP



Hovercraft B

CLOSED LOOP



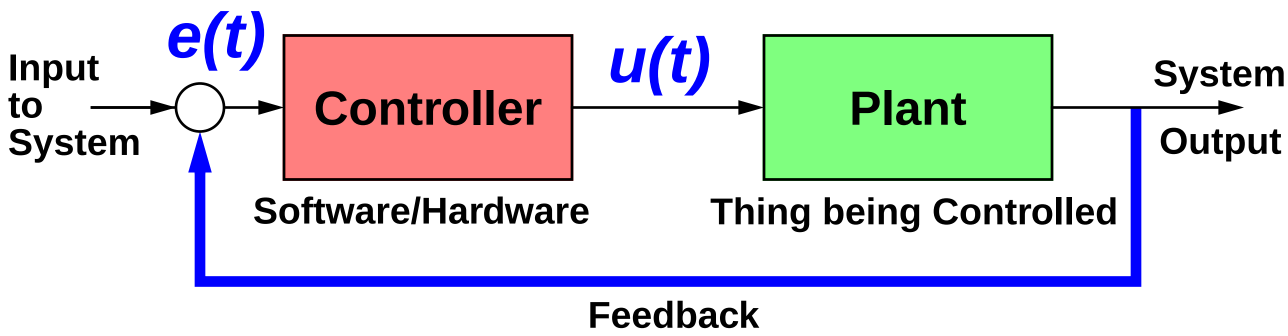


Feedback: Orientation changes → course changes

As hovercraft changes orientation with respect to tape, light sensor readings change, causing course corrections

PID Controller

Proportional, Integral, Derivative

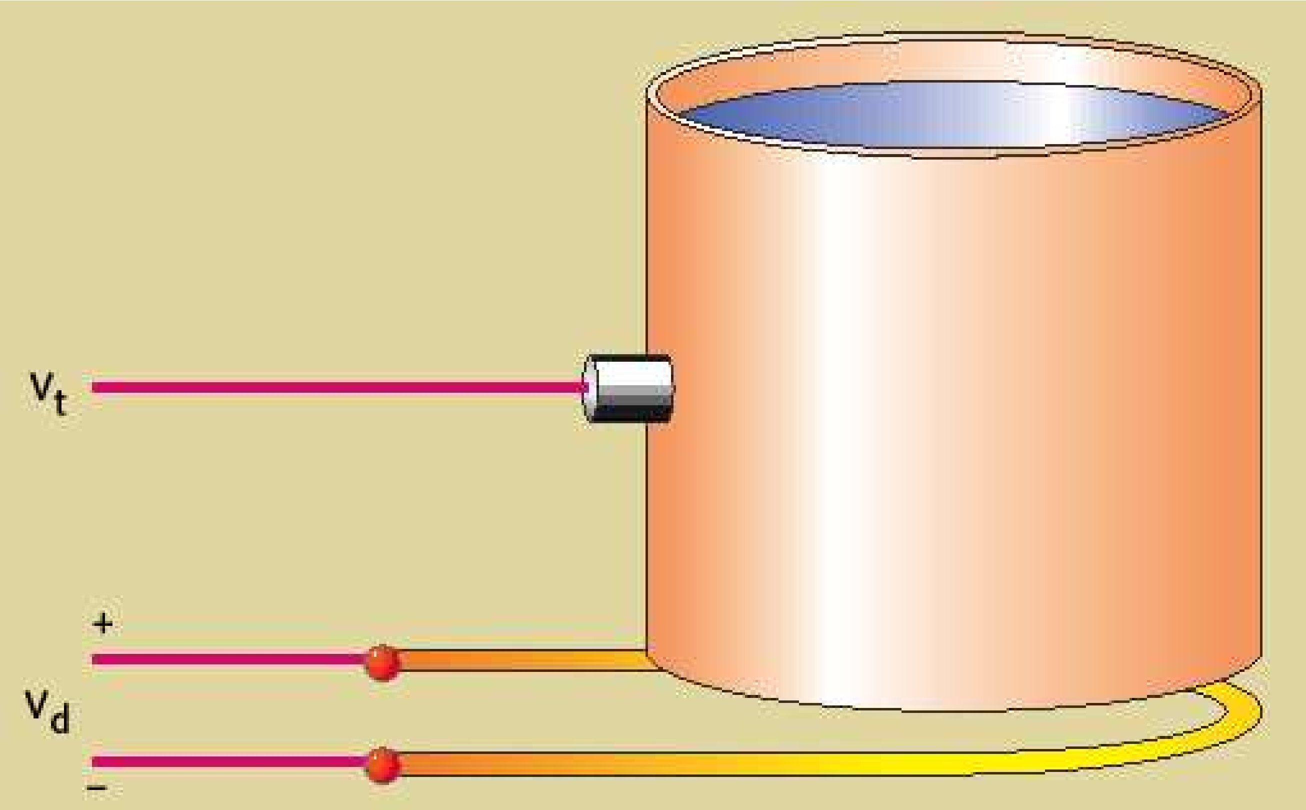


$$u(t) = K_P e(t) + K_I \int_0^t e(t)dt + K_D \frac{de}{dt}$$

- ***Proportional*** **term ensures the system reacts as soon as there is a change in the system: change in new output follows the error.**
- ***Integral*** **term provides hysteresis, tracks effectiveness of control system: measures delta between output and input to date.**
- ***Derivative*** **term anticipates future behavior: reacts to quick changes in plant output vs. input.**

Example System

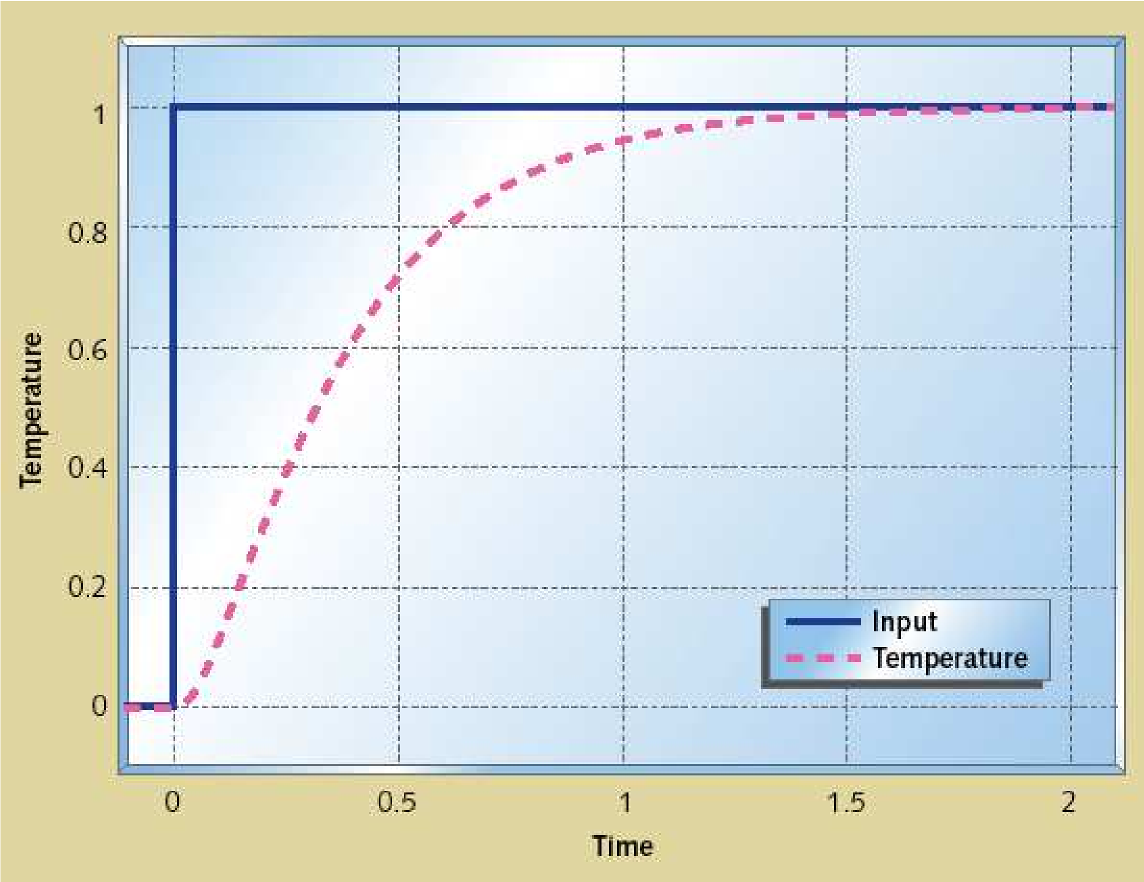
Thermostat — A Popular Controls Example



- **Water heater: controlled by voltage**
- **Sensor: temperature (V representing T)**

Example System

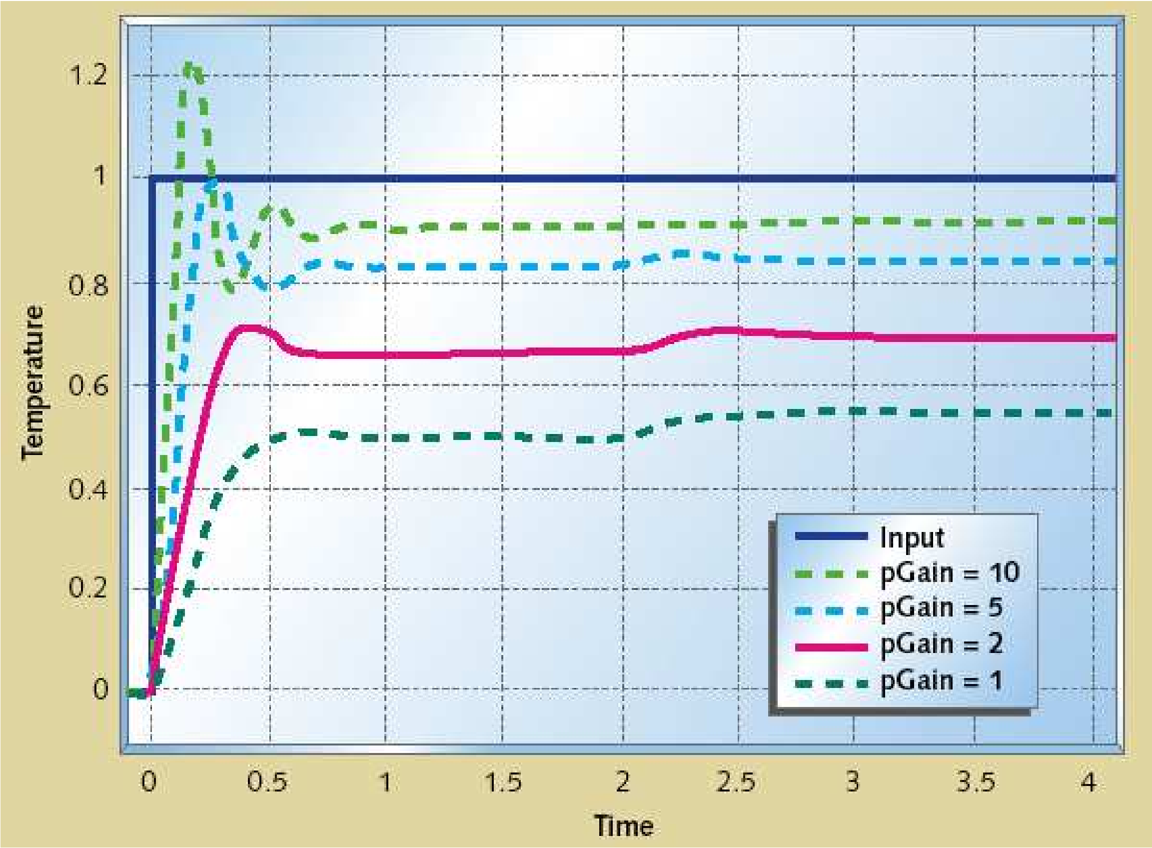
Thermostat — A Popular Controls Example



- **Response of system to step input**

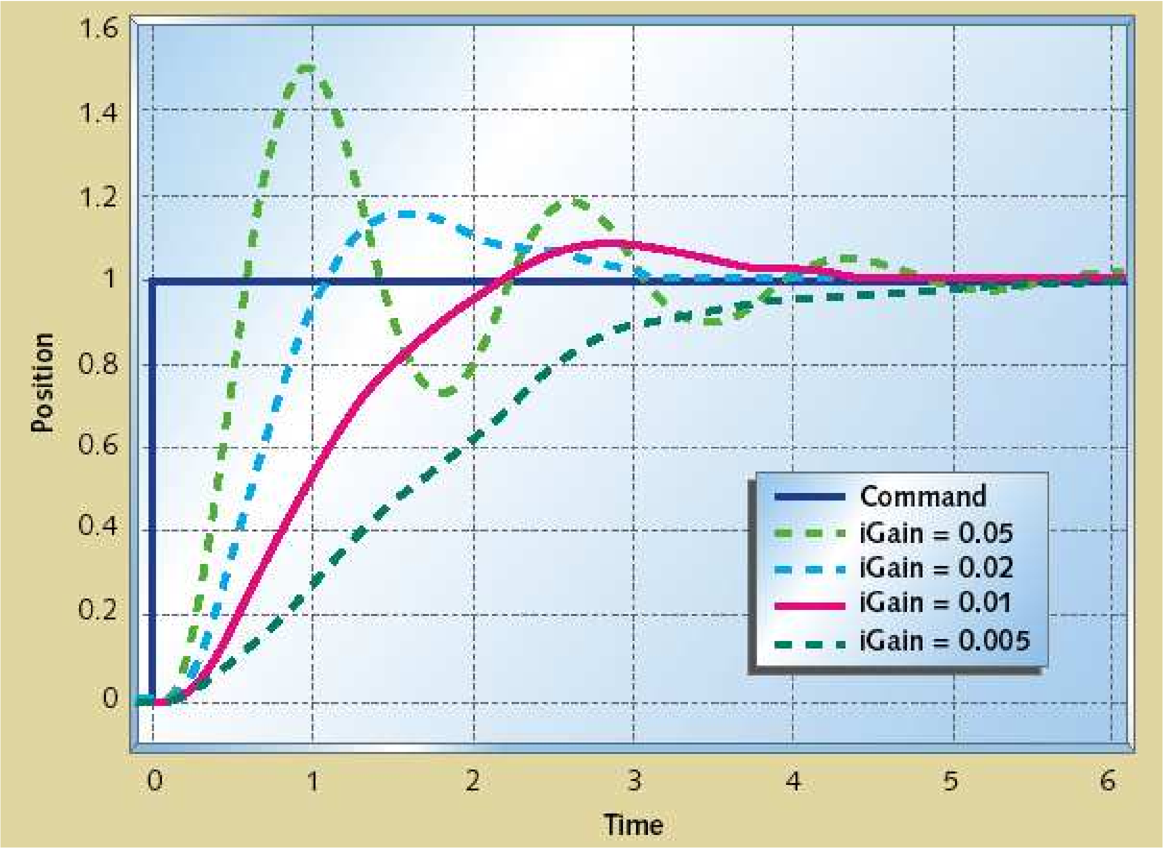
Proportional Controller

```
while (1)
    error = desired() – reading();
    increase_temp( error * pGain );
```



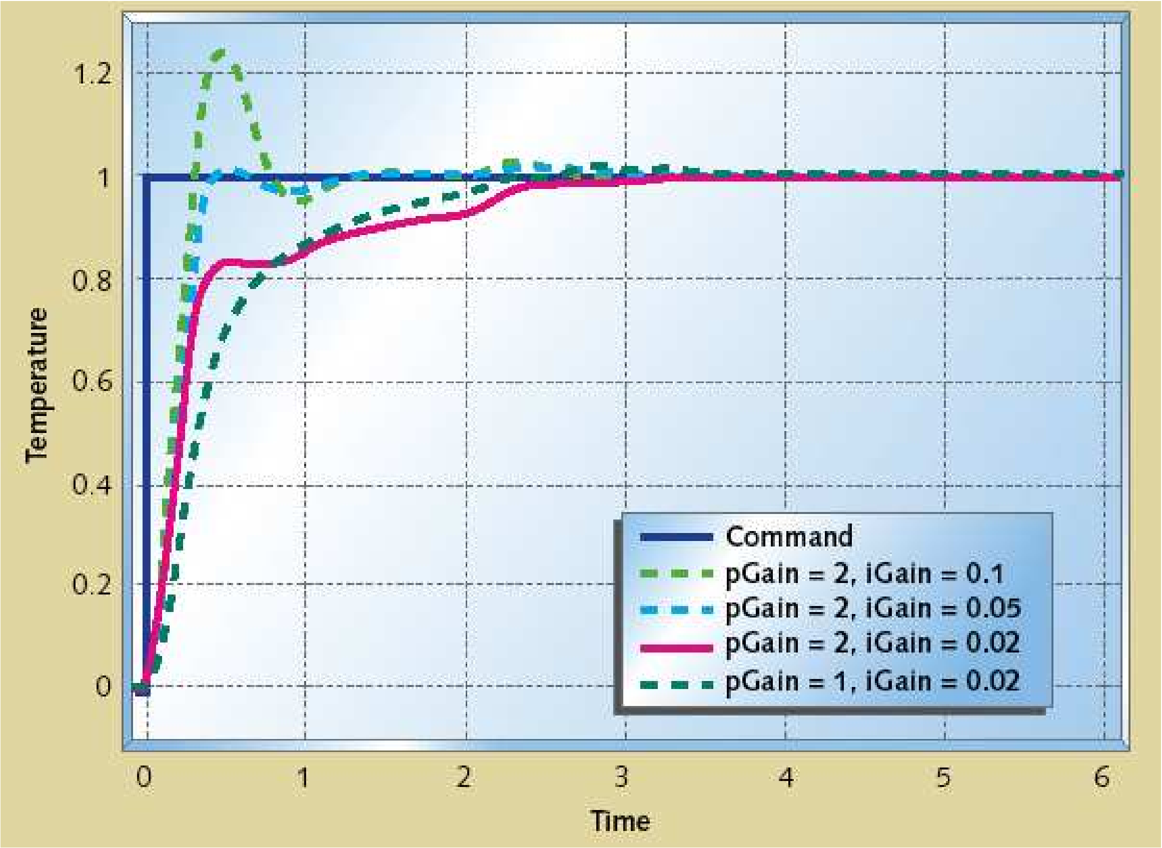
Integral Controller

```
while (1)
    cum_E += [desired() – reading()];
    cum_E = bound_cumulative_error( cum_E );
    increase_temp( cum_E * iGain );
```



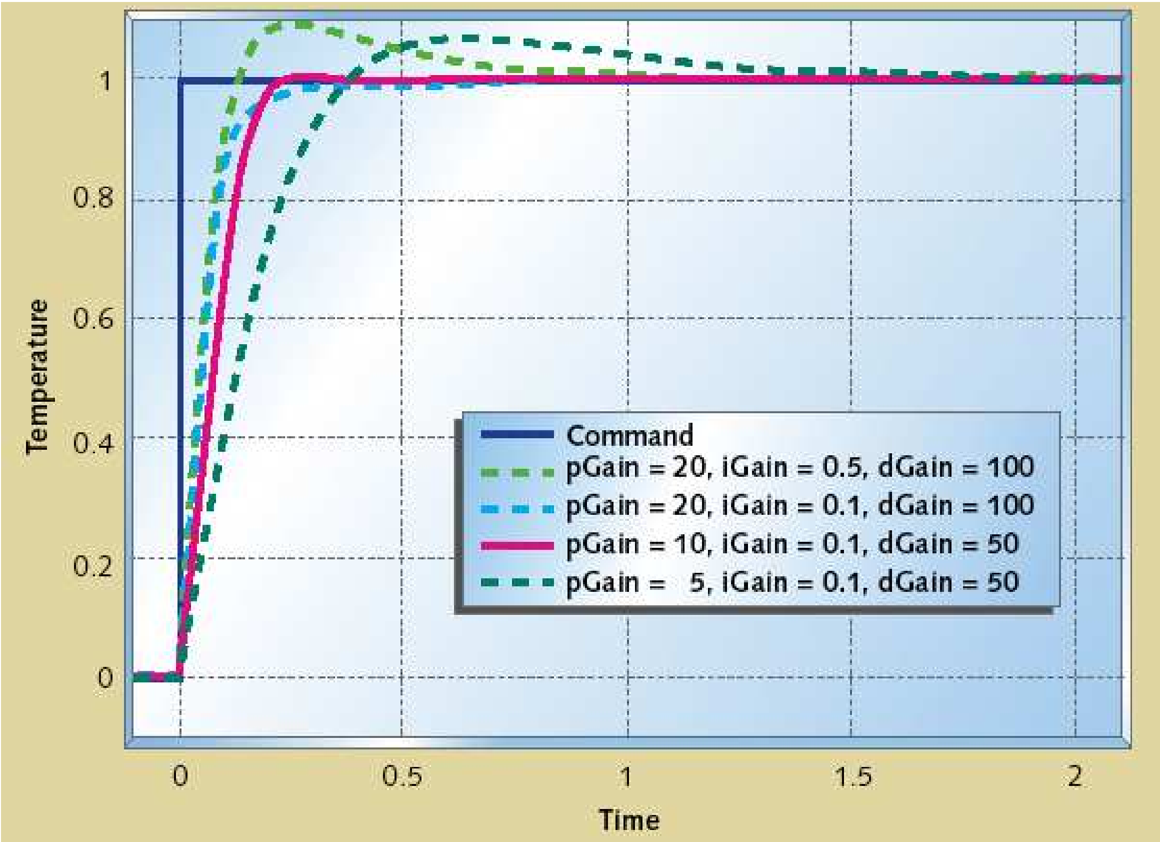
PI Controller

```
while (1)
    error = desired() – reading();
    cum_E += error;     // and then bound it
    increase_temp(   error * pGain +
                     cum_E * iGain );
```

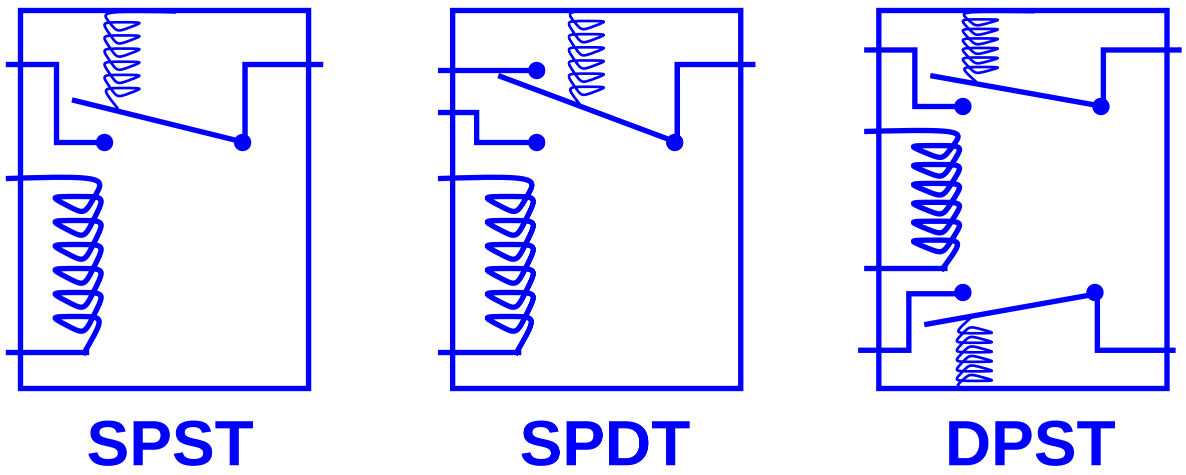


PID Controller (predictive)

```
error = desired() – reading();
cum_E += error;      // and then bound it
delta = prev_reading – this_reading;
increase_temp(   error * pGain +
                 cum_E * iGain +
                 delta * dGain );
```



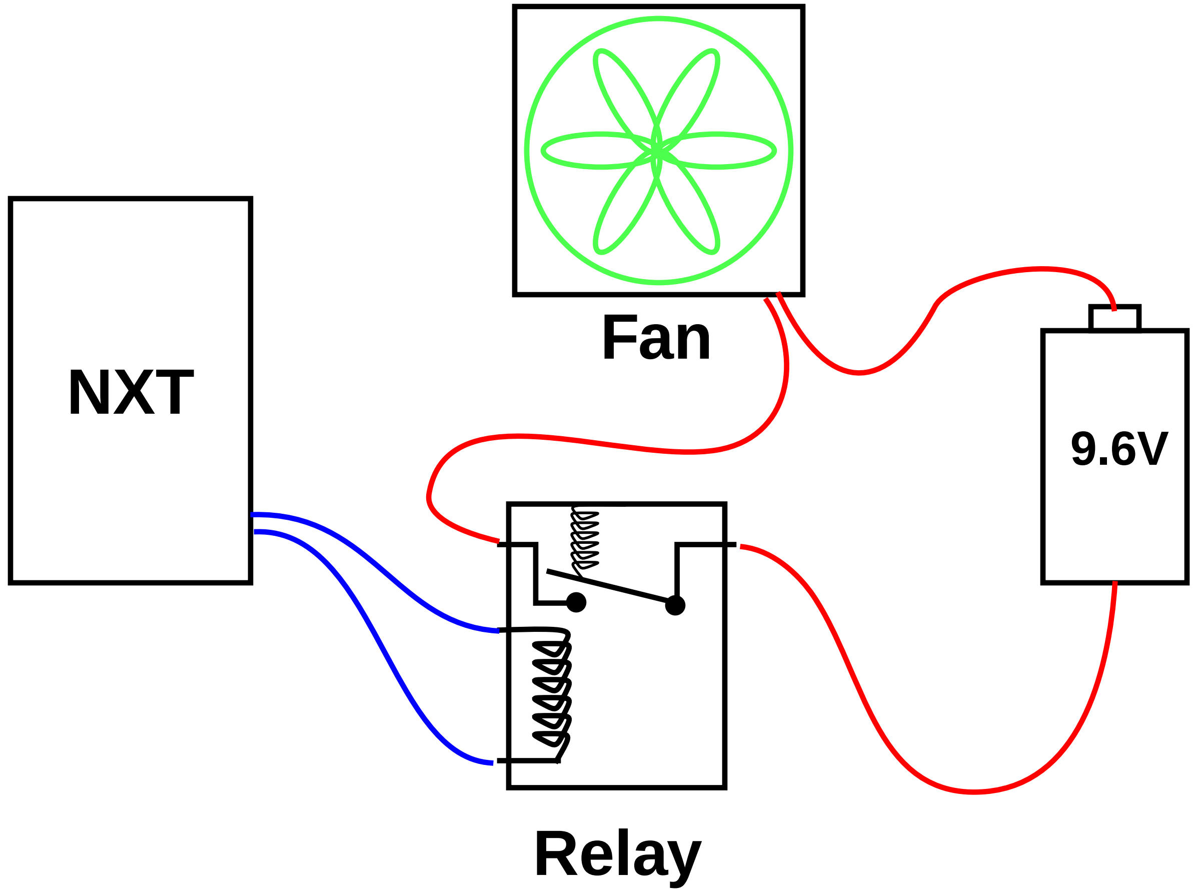
Circuits: Relays



Two VERY DIFFERENT things:

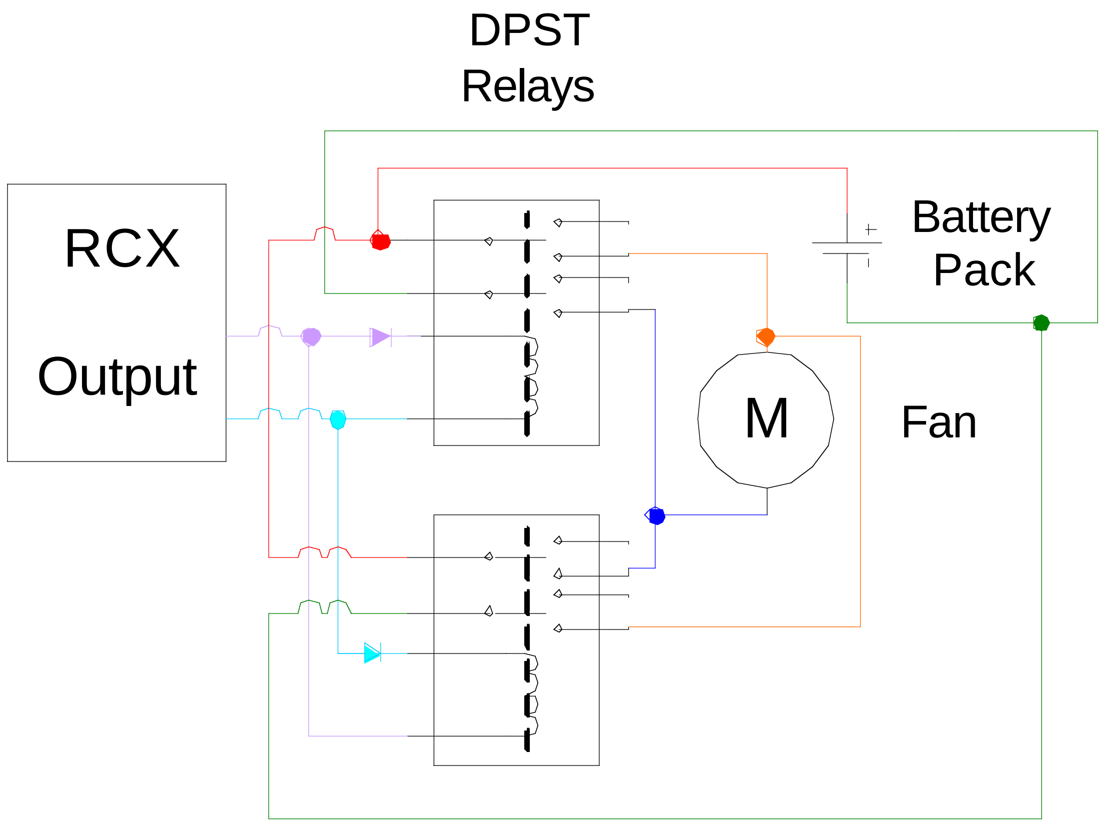
- **trip voltage (to power electromagnet)**
- **max current (through switch)**

Circuits: Simple Fan Control



Downside: only on/off control

Circuits: Reversing Fans



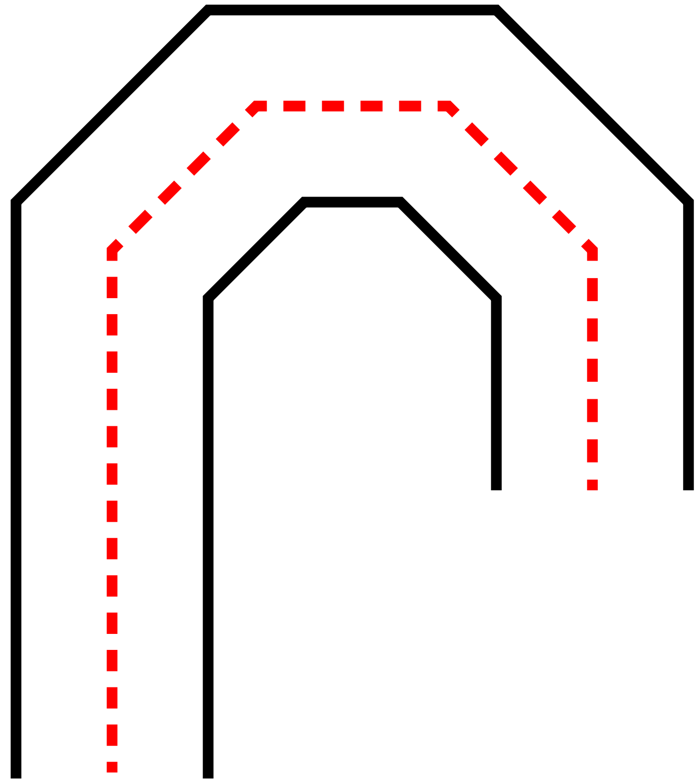
*** thanks to Prof. Wes Lawson**

Hovercraft Control Issues

Issues you will have to address:

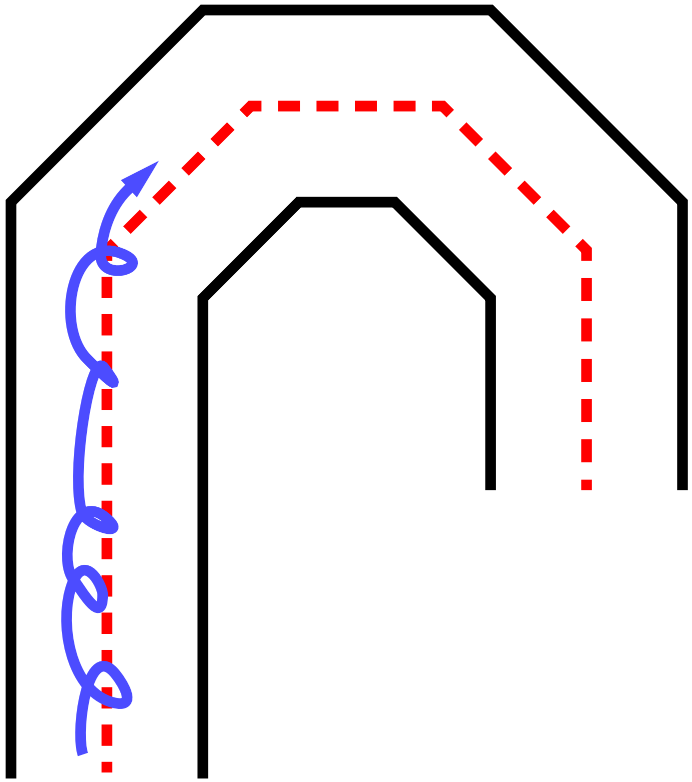
- **Sensing location**
- **Sensing speed/direction**
- **Changing location/speed/direction**
- **Making informed decisions**

Sensing Location



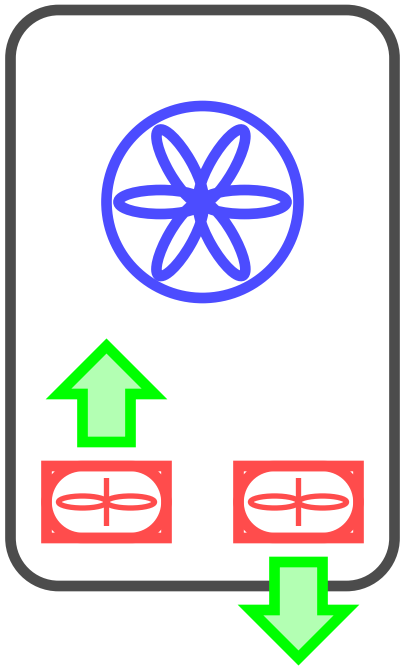
- **Echolocation (distance from walls)?**
- **Dark/light sensor (following tape)?**
- **Magnetic sensor (following tape)?**
- **GPS (absolute coordinates)?**

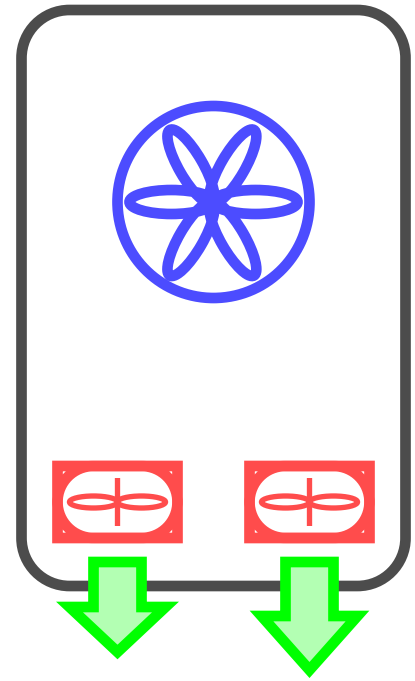
Sensing Speed/Direction



- **Is following tape/walls sufficient?**
- **What about angular momentum?**

Changing Orientation





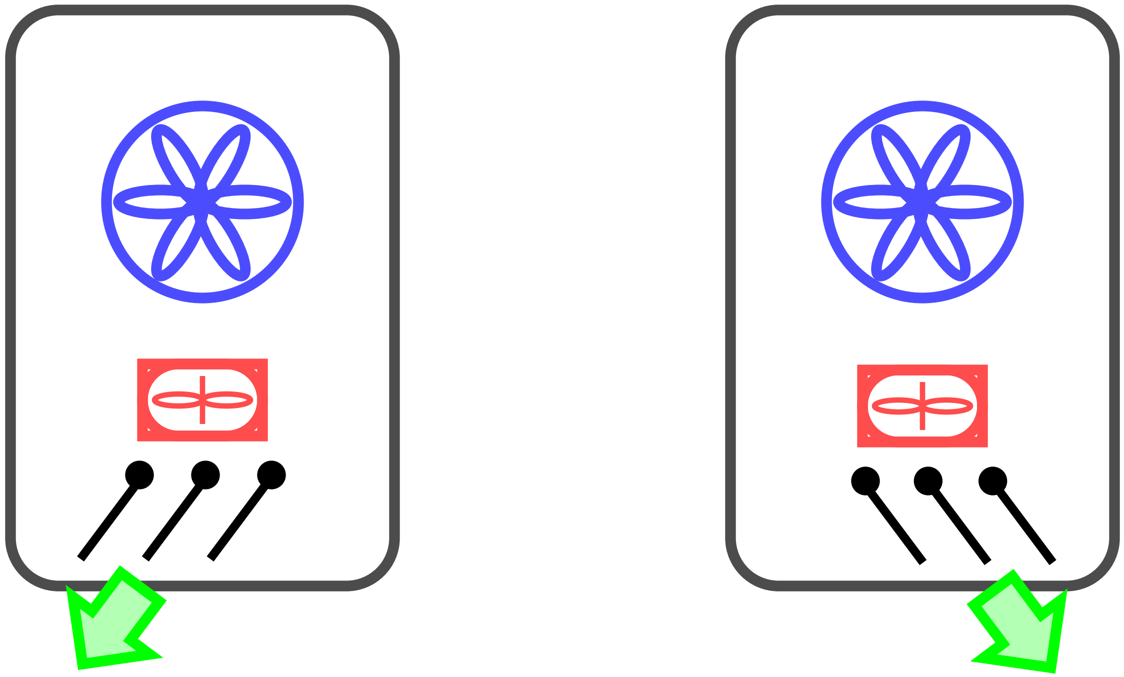
Turning is obvious
… or is it?

How do you *stop* turning?

Forward thrust is obvious
… or is it?

Are your fans perfect?

NXT vs. RCX: servos

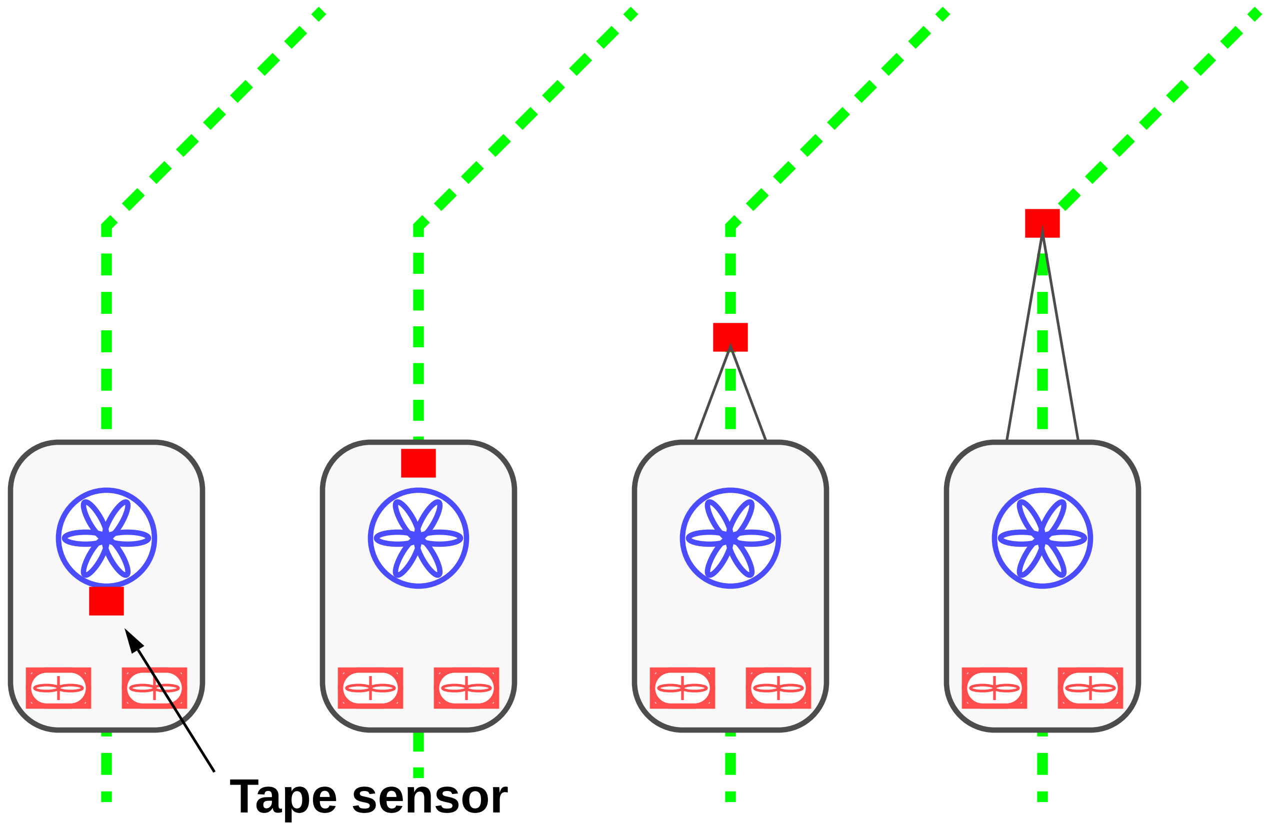


Does not necessarily *simplify* control …

BUT — reduces number of outputs by 1
AND gives a finer degree of control

Some Things to Think About

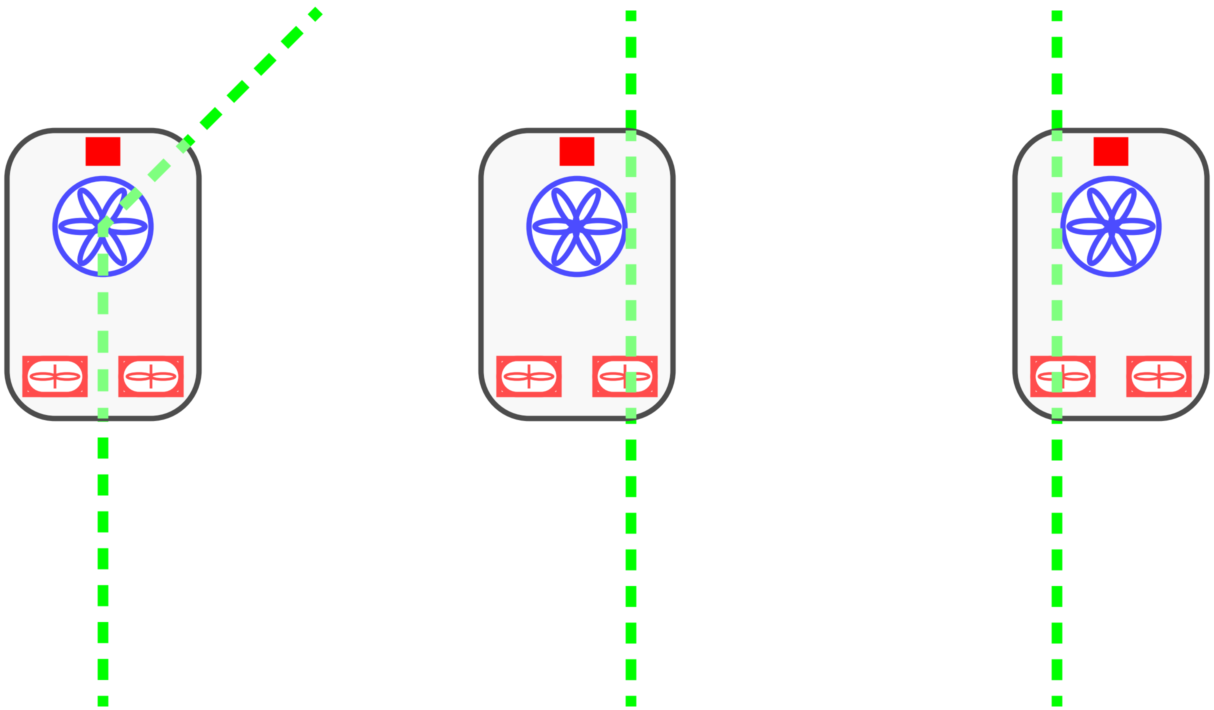
Which is likely to be easiest?



When you drive, do you look ahead to turn?

Some Things to Think About

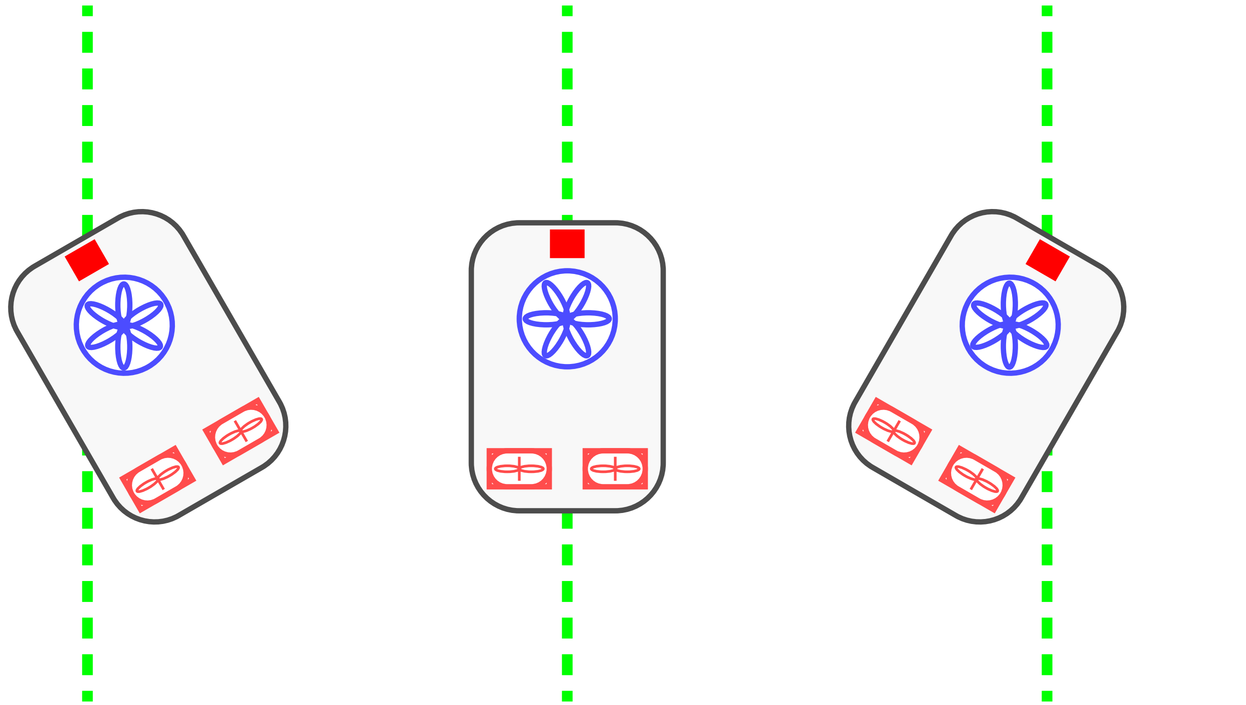
How do you tell the difference?



And does it matter?

Some Things to Think About

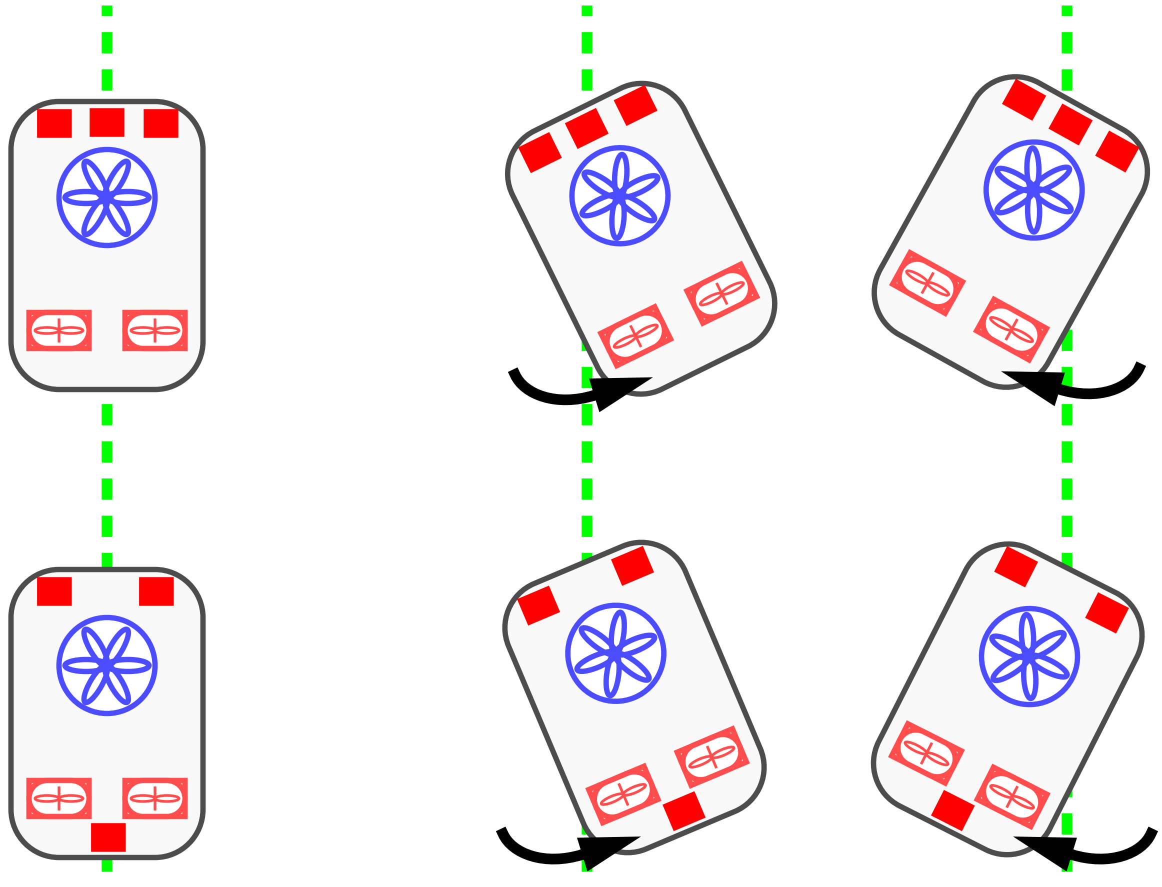
How do you tell the difference?



And does it matter?

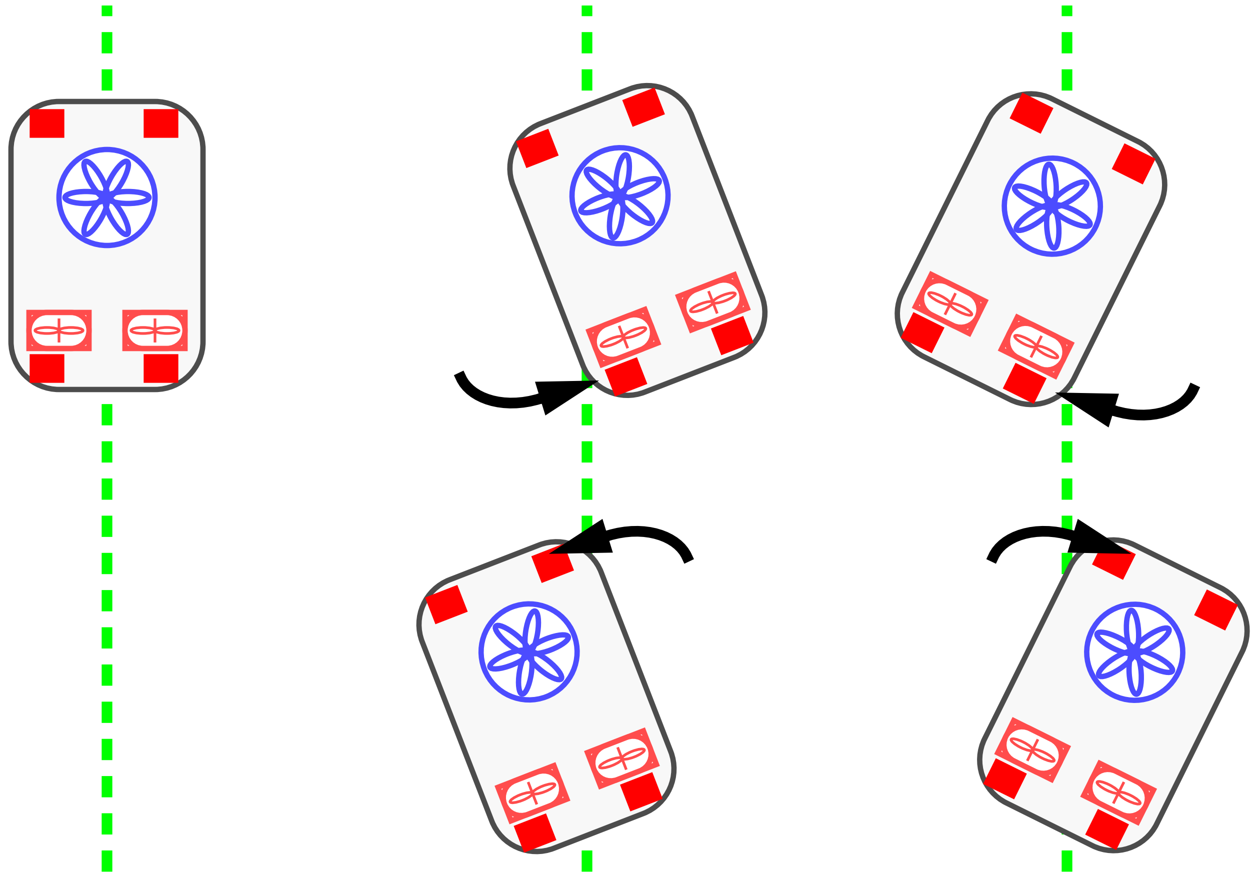
NXT vs. RCX: sensor inputs

4 inputs vs. 3 — RCX has 3 inputs:



NXT vs. RCX: sensor inputs

4 inputs vs. 3 — NXT has 4:



Bottom Line

The control problem will be your biggest headache when designing your hovercraft.

Give it a lot of thought.