Closed Loop Control

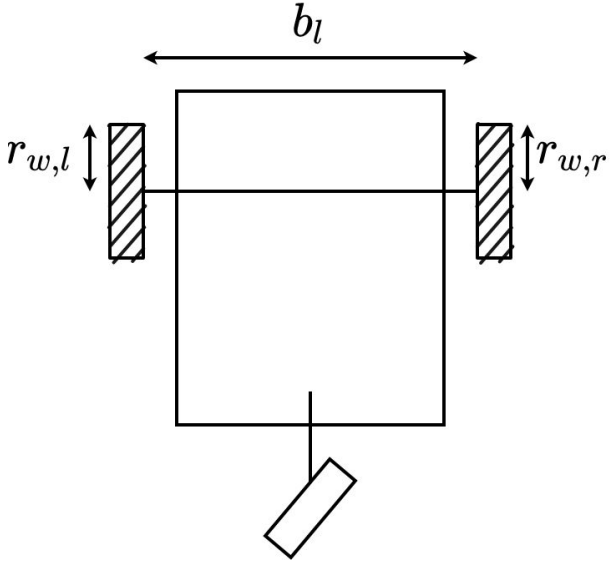
Exercise

SAPIENZA
UNIVERSITÀ DI ROMA

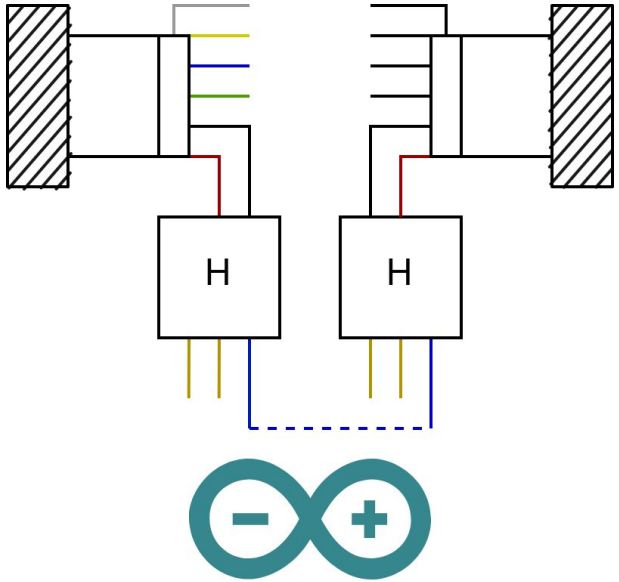
Giorgio Grisetti
Emanuele Giacomini

- Immagine to realize a Differential Drive M.R.

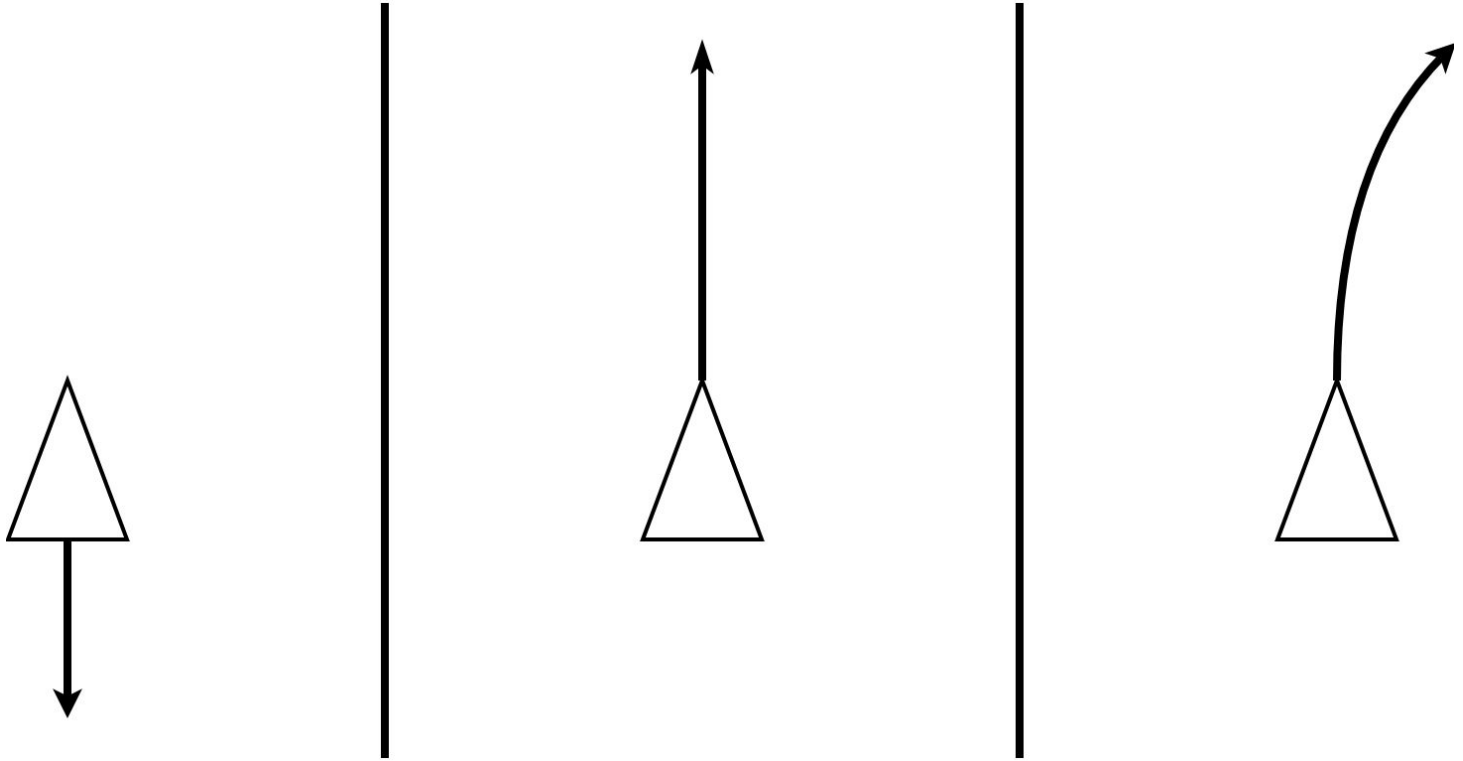
- We need:
  - Platform
  - 2 Motorized wheels [Brushed DC + H Bridge]
  - 1 Caster



- Mount everything and test the platform
- H Bridge configuration:
  - Dual Dir + PWM [3 control pins]
- Apply same PWM signal to both motors



- What happens ?

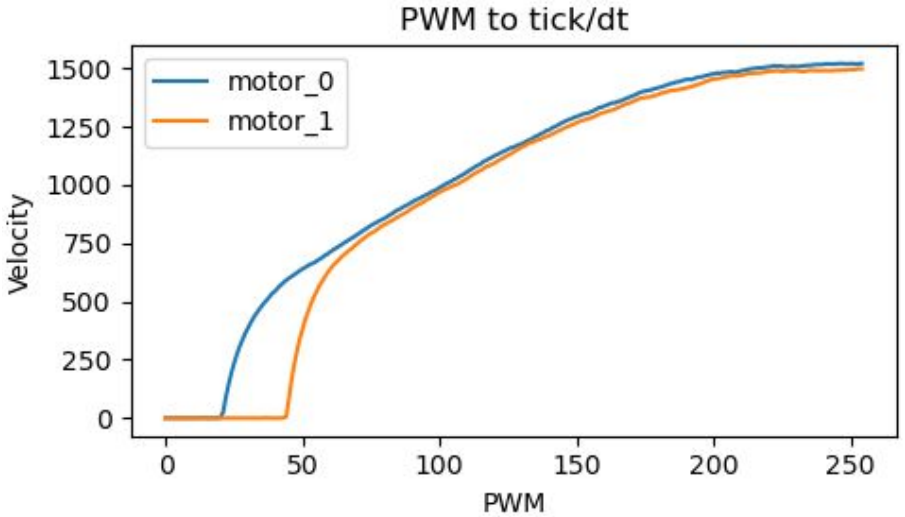


Problem

- Robot is unable to go straight
- The motors are similar but not identical
- How to correct this behavior ?
- Two solutions:
  - Calibrate PWM channels
  - Use Closed Loop Feedback correction

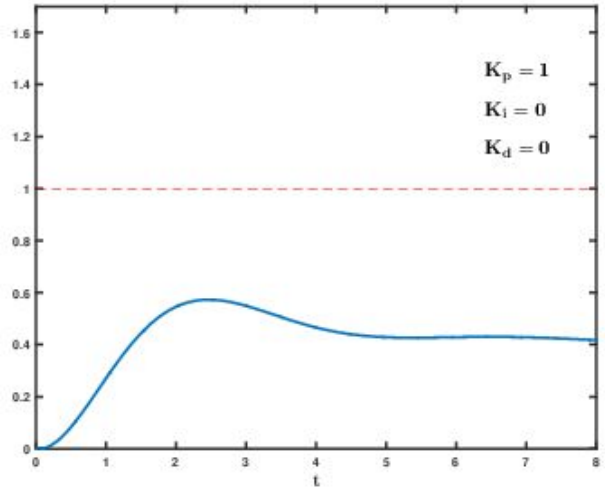
PWM Calibration

- Calibrating PWM channels is painful...
- Velocity differences are not constant
- Requires interpolation and parameter estimation
- Can be done… but why ?



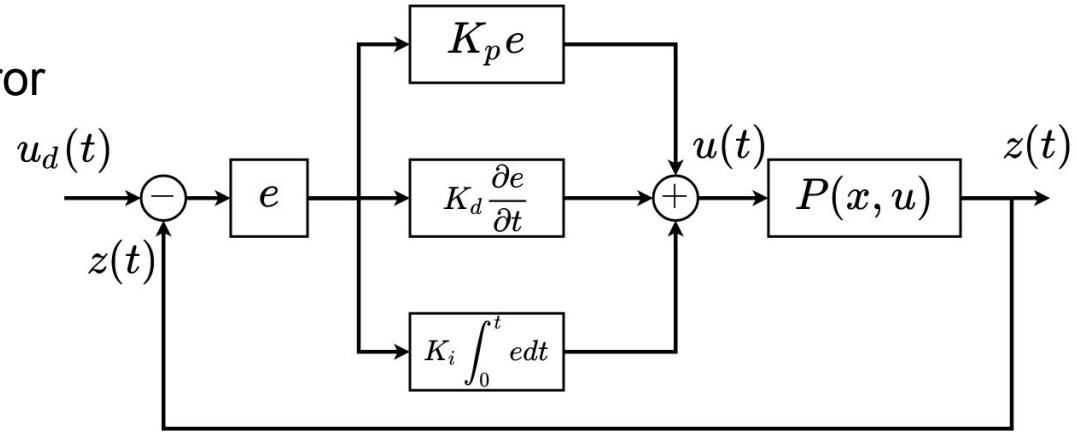
Closed Loop Feedback Control

- Is the shaft's angular velocity measurable ?
  - Yes! encoders
- Compares measurements with input commands
- Feedback design guarantees convergence
- Many Feedback controllers exists
  - We use *Proportional Integral Derivative* [PID]



PID Control

- Compute error between desired and measured speed
- ***P*** acts like a spring
- ***I*** removes the steady-state error
- ***D*** acts like a damper

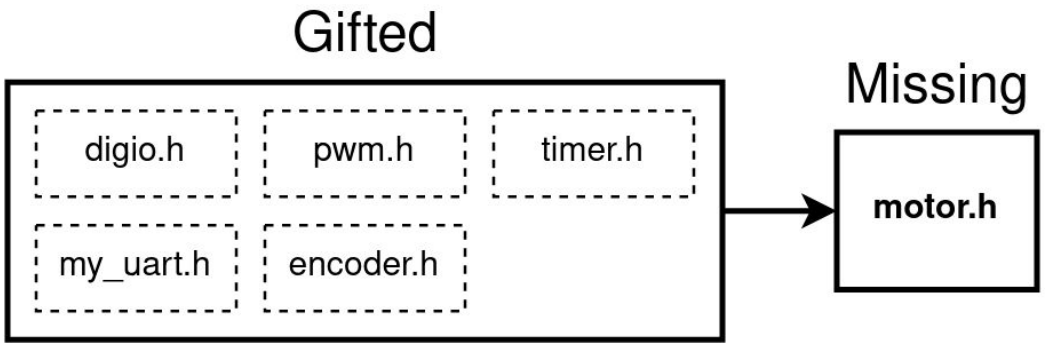


Discrete PID Control

- We need to work with discrete time intervals
  - Proportional
    - $u_P(t) = K_p e(t)$
  - Integral
    - $u_I(t) = \sum_{t_i=0}^{t} K_i e(t_i) \Delta t$
  - Derivative
    - $u_D(t) = K_d \left(\frac{e(t) - e(t-1)}{\Delta t} \right)$
- Useful stability tips:
  - Bound the error
  - Bound the components

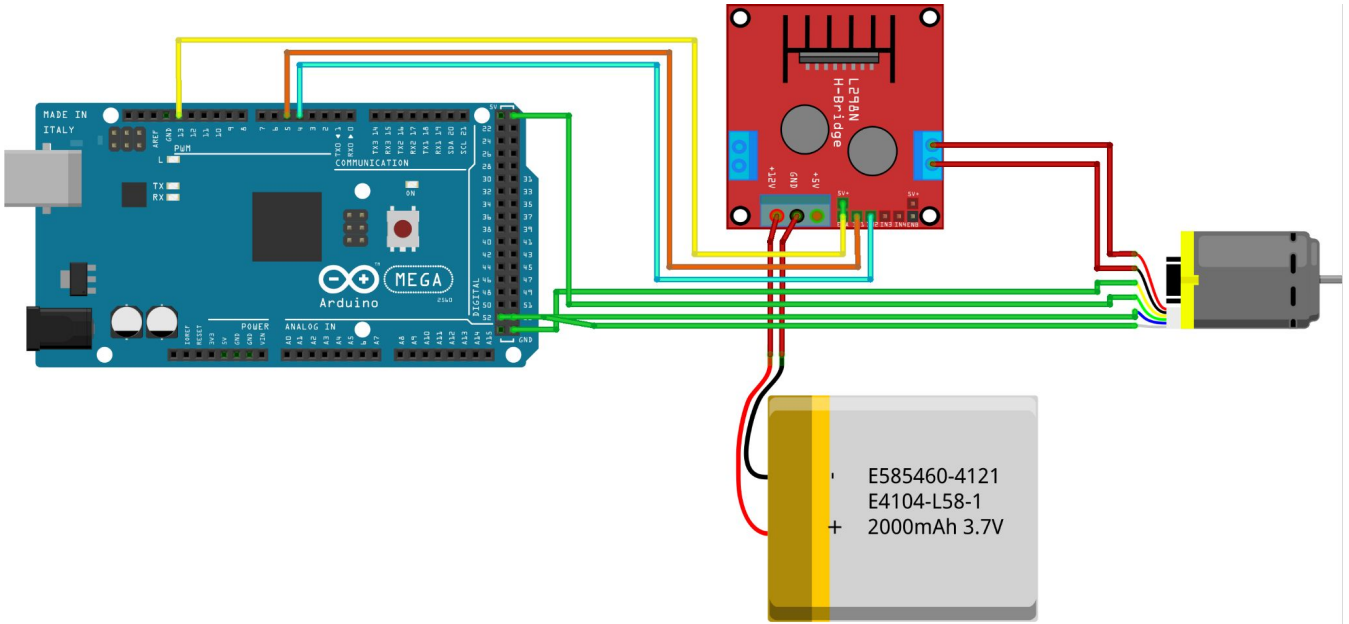
Exercise

- We designed an exercise for you
- Implement a motor class
  - Includes a PID controller
- We gift you all the other modules :)



Our development

- Design problem
  - Comms with PC
  - Motor Control (Direction and Speed)
  - Encoder

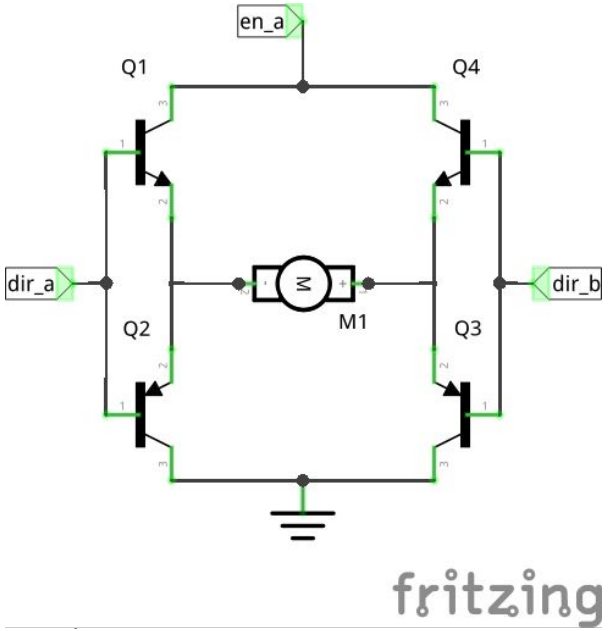


- UART library
  - Initializes Serial communication
    - 19200 bit/sec
    - 8-bit data
  - Allows single/multiple byte transmissions
    - {put/get}Char()
    - {put/get}String(buf)

- Timer library
  - Allows precise loop timings
    - Set to 10ms/loop

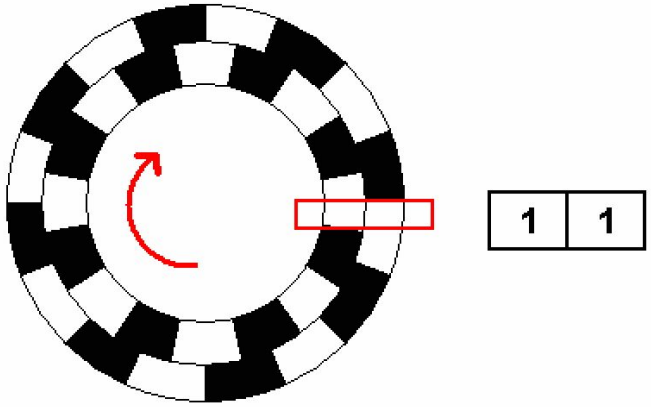
Motor Drive

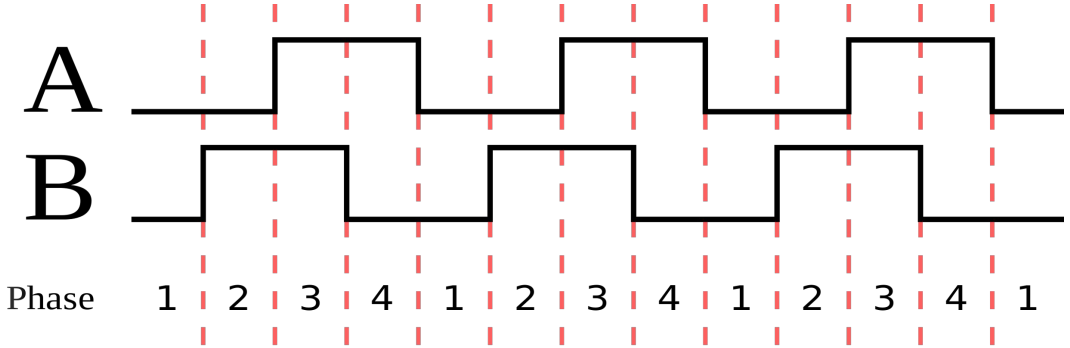
- Brushed DC Motors requires H-Bridge circuit
  - 2 signals for direction
  - 1 signal for speed (Pulse Width Modulation)
- Digital I/O library
  - Allows binary signal generation
    - dir_a and dir_b
- PWM library
  - Allows 2^8 PWM configurations
    - en_a



Closed Feedback

- Brushed DC may include angular velocity sensor
  - Incremental quadrature encoder
- Encoder library
  - Allows 1 encoder port (pin 52, 53)
  - Asynchronous update
  - Refer to previous lectures for details



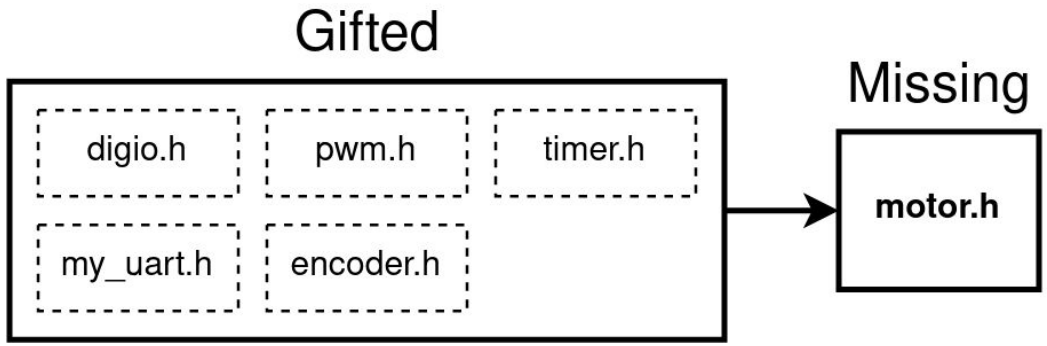


Your turn

- Last missing section

- Abstract the motor interface
  - H-bridge for control
  - Encoder for closed loop feedback

- Design a Motor class that:
  - Allows for multiple controls:
    - PWM Open Loop
    - PID
  - **Computes** PID corrections



Things to know

- For simplicity:
  - DigIO only handles Register E
    - Bit 4 -> pin 2
    - Bit 5 -> pin 3
    - More on the code...
  - PWM only handles pin 13
  - Encoder mapped to pins 52 and 53

- Motor constructor takes:
  - Dir_a, dir_b regE bits
  - PID parameters (Kp, Ki, Kd)