{Learn, Create, Innovate};

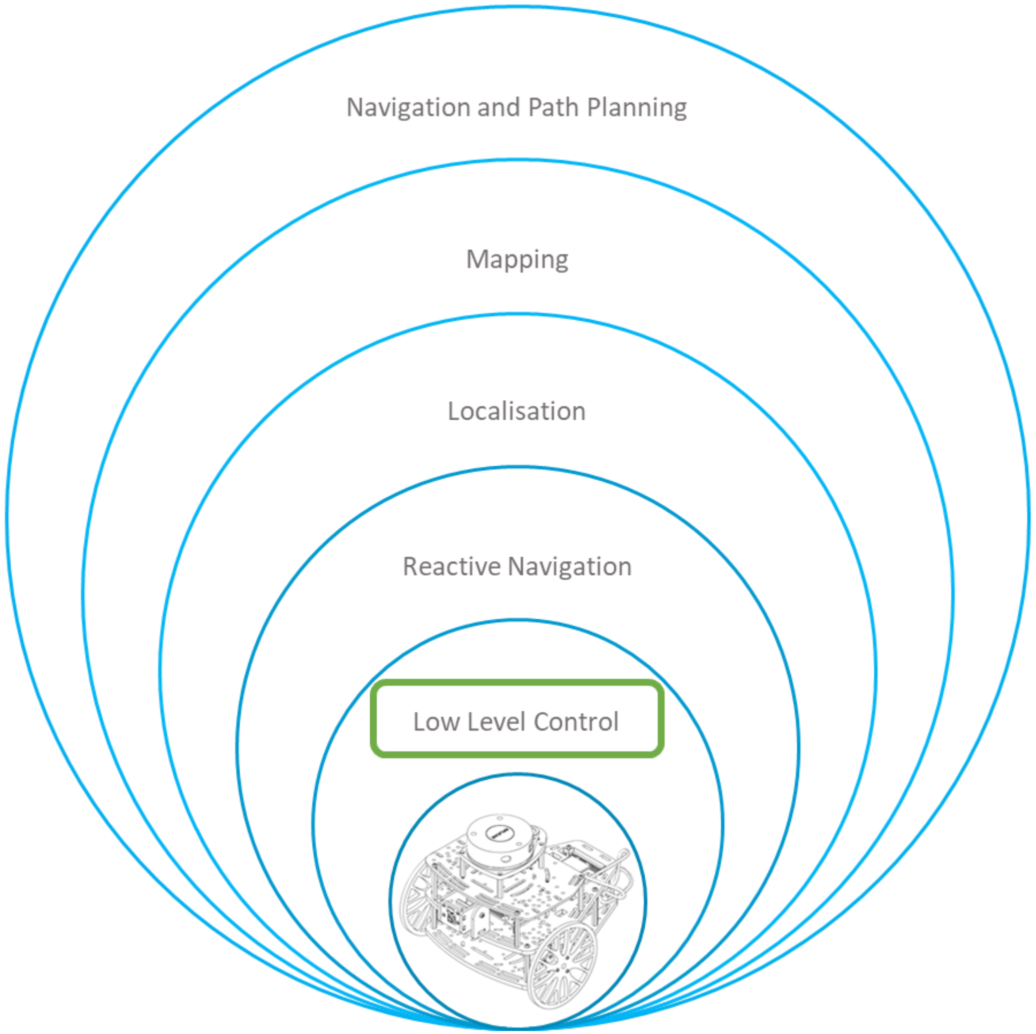
DC Motor Control

Introduction

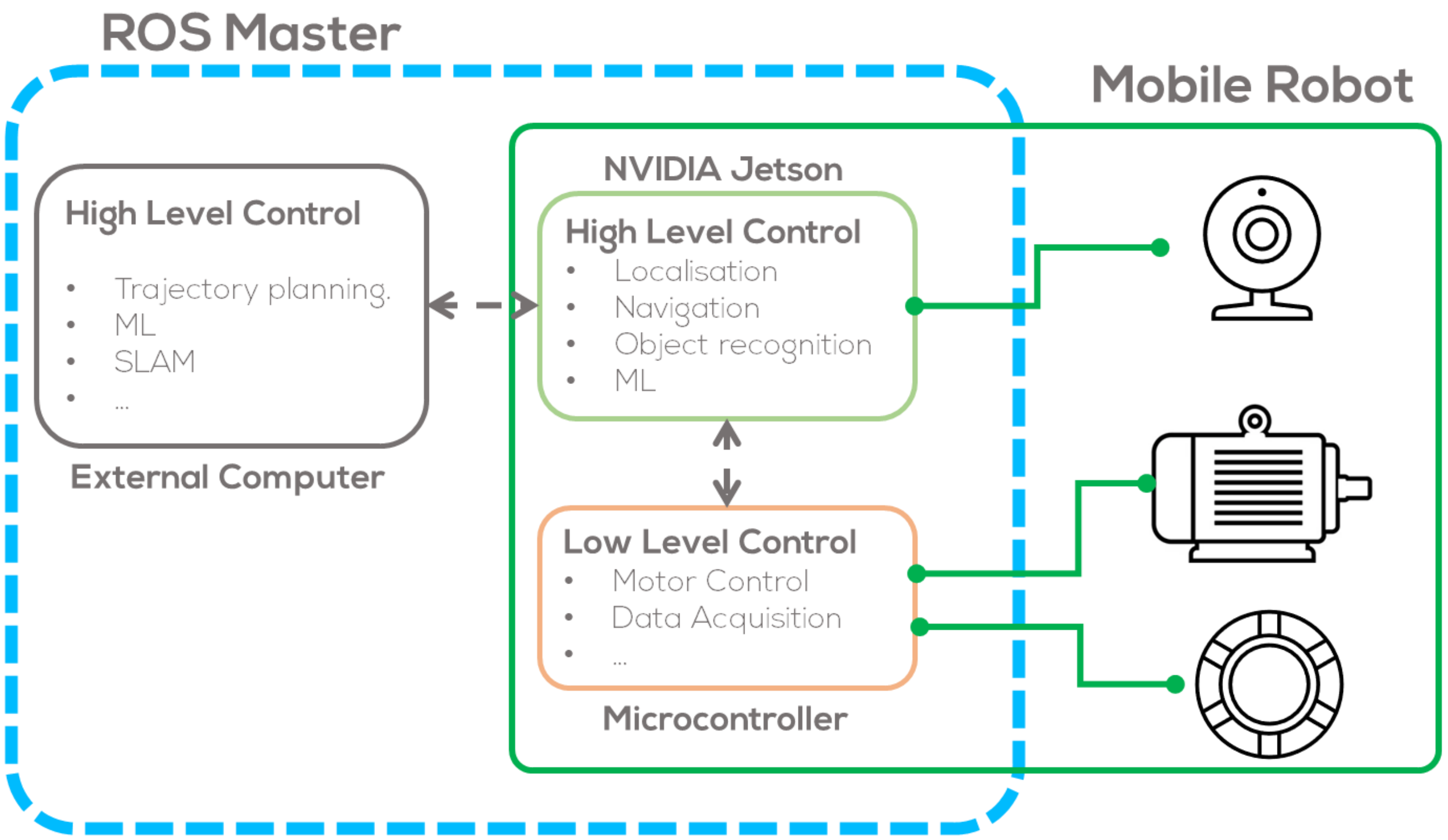
Manchester Robotics

Control Hierarchy

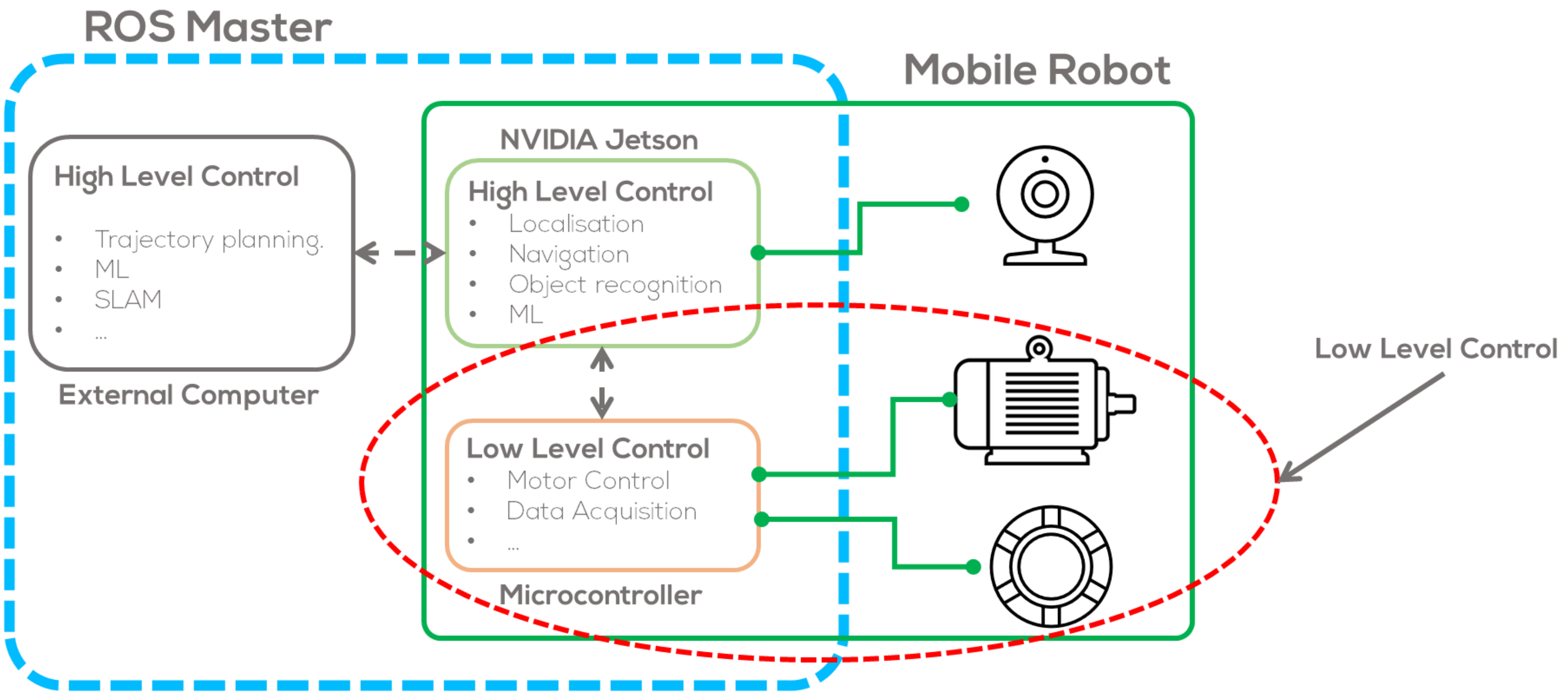
- Robotics systems, are intended to perform complicated tasks in different environments. Such tasks, can be performed autonomously, semi-autonomously or remote controlled.
- The control of such systems can become very complex depending on the task to perform.
- The HCS divides the control into layers, dividing each complex task into subtasks (goals) to be achieved by a dedicated layer.
- This course is dedicated to the Low-Level control layer using ROS.



ROS Control Hierarchy

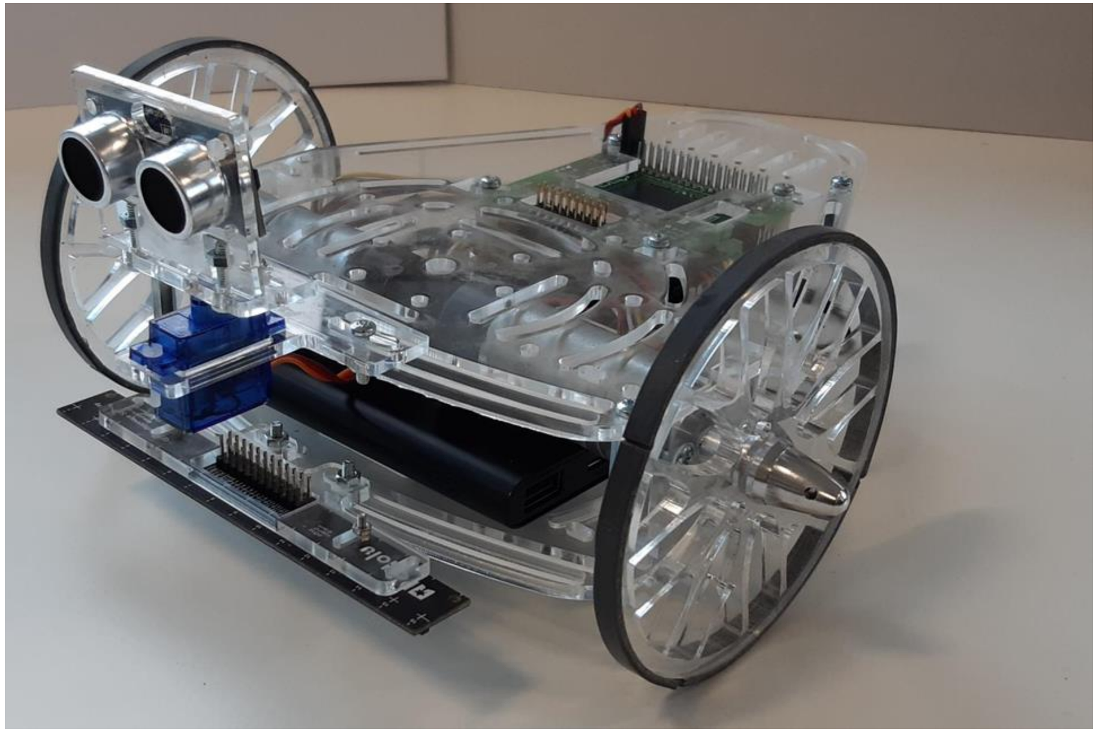


ROS Control Hierarchy



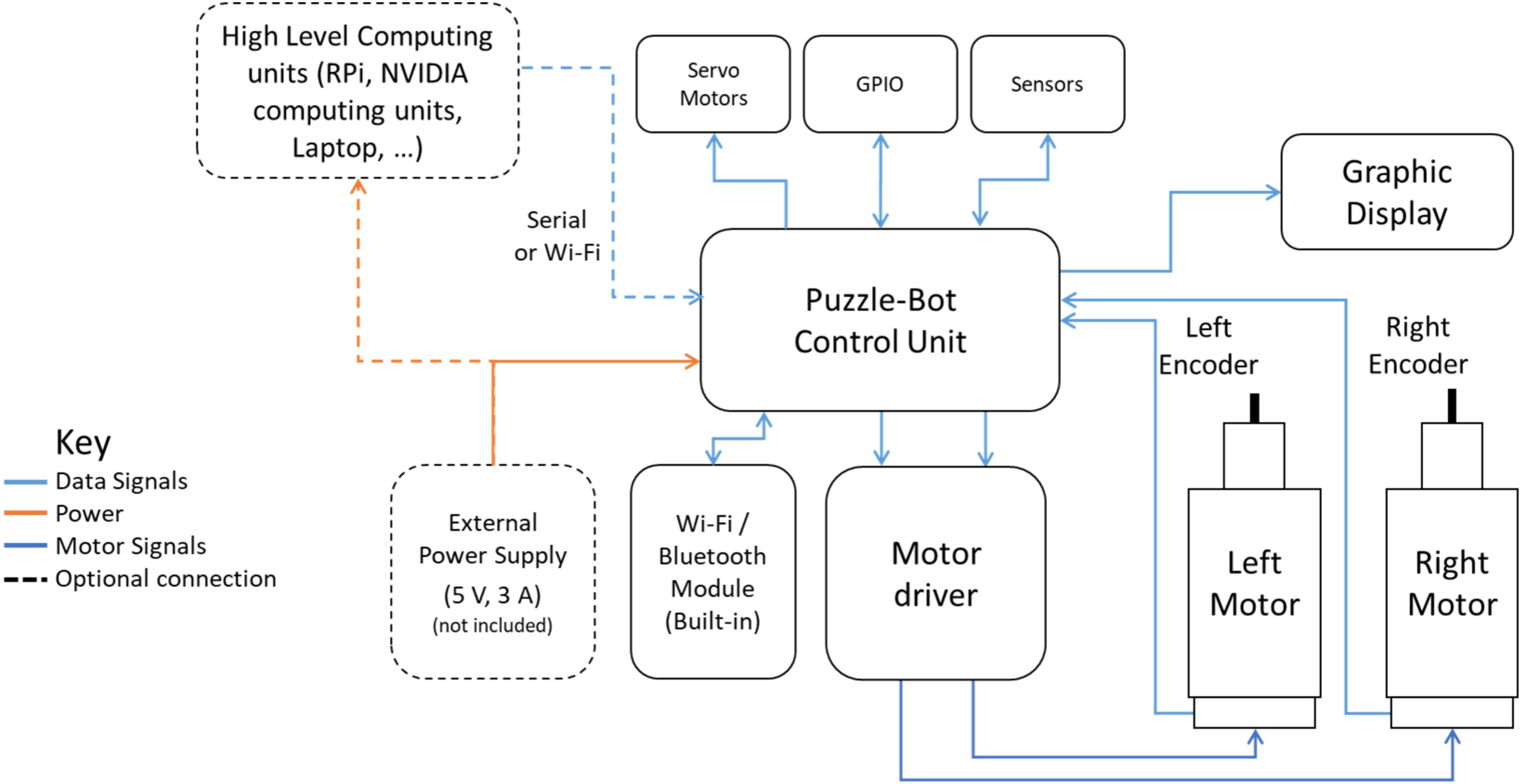
Mobile robots

- Mobile robots, require different sensors and actuators to read information from the environment, and to interact with it.
- One of the most common actuators found in most robotic systems are DC motors.
- DC Motors are found in different robotics applications from wheeled mobile robots, to robotic manipulators and unmanned aerial vehicles.
- One example of a wheeled mobile robot is the Puzzlebot, which uses two DC motors one for each wheel.
- DC motors, are a widely studied in different fields of science from electromechanical systems to control engineering.



Puzzlebot by Manchester Robotics.

Puzzlebot sensors and actuators



DC Motors

Introduction

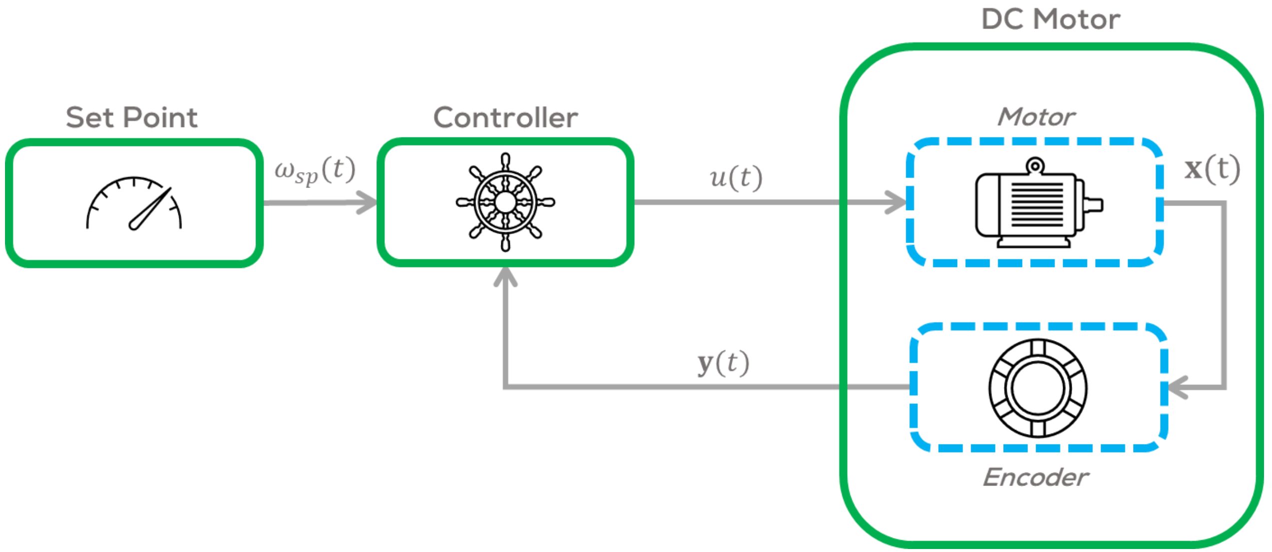
- A direct current (DC) motor is a type of electric machine that converts electrical energy into mechanical energy.
- DC motors take electrical power through direct current and convert this energy into mechanical rotation.
- This is done by using generated magnetic fields from the electrical currents, powering the movement of a rotor fixed within the output shaft.
- The output torque and speed depends upon both the electrical input and the design of the motor.

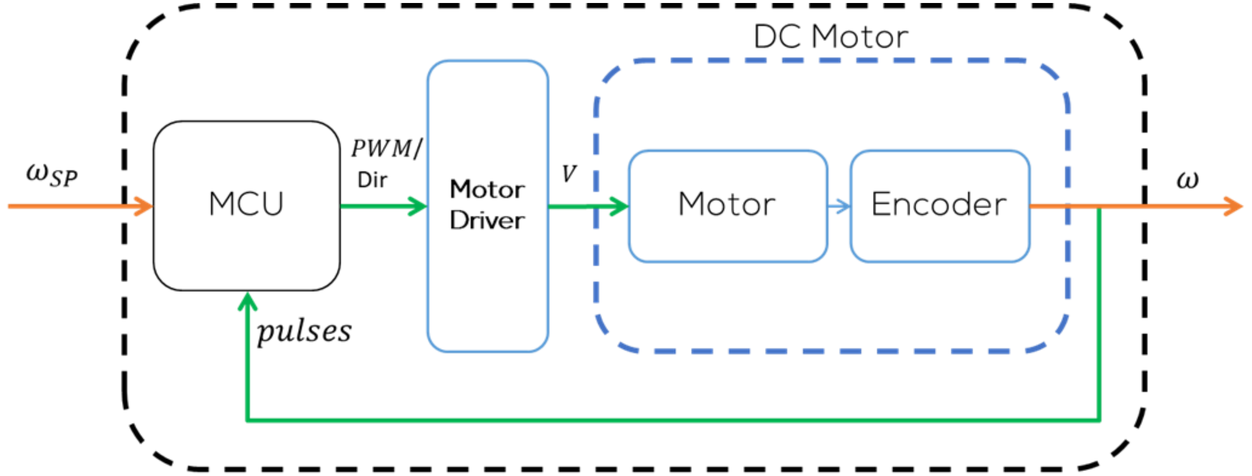
DC Brushed Motor with Encoder.

DC Motor Model Representation.

DC Motors

- In robotics, controllers are used to regulate the rotational speed, angular position or torque, required by the application.
- In robotics this is called low level control.
- For the case of a wheeled mobile robot is a common practice to implement a PID control to regulate the angular speed of the DC motors.
- The regulation of the angular speed or position of a motor, requires different stages.
  - Controller Stage
  - Power Stage (Driver)
  - Plant
  - Sensor Stage



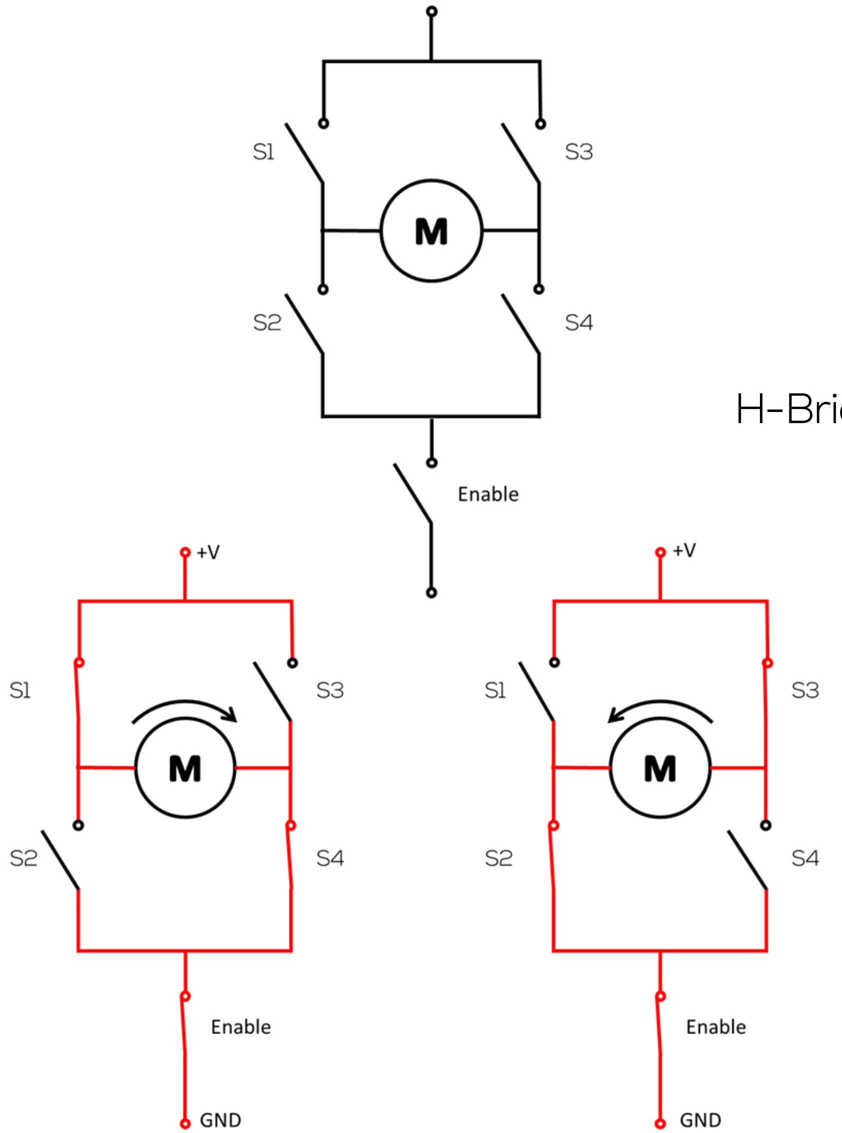


DC Motor Control Diagram*

Motor Driver

- H-bridge is an electronic circuit that switches the polarity of a voltage applied to a load.
- They work using a combination of switching components (mechanical switches, transistors, etc.), as shown in the diagram, to change the polarity to the load.
- H-Bridge Drivers are some of the most common motor drivers used in the control DC motors to run forwards or backwards.

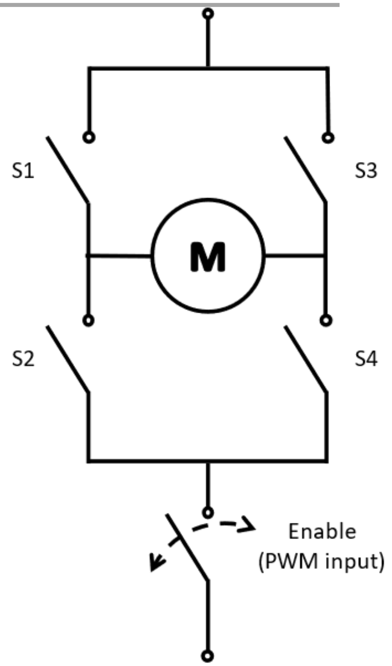
S1	S2	S3	S4	Motor
0	0	0	0	Motor Off
1	0	0	1	Right Turn
0	1	1	0	Left Turn
1	1	0	0	Short Circuit
0	0	1	1	Short Circuit



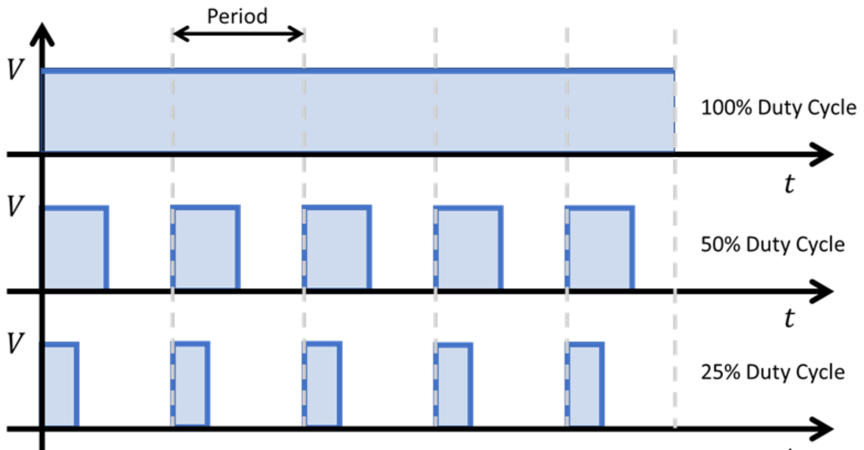
H-Bridge motor Driver Diagram

Motor Driver

- Another capability of the motor driver is to regulate the angular speed of the motor.
- There are many ways to obtain this result, one of the most common one is to send a PWM (Pulse width modulated signal) to the enable pin of the H-Bridge.
- PWM (Pulse Width Modulation): Is a technique used in engineering to control the average power delivered by an electrical signal, by dividing it into discrete parts.
- In practice, this is accomplished by rapidly turning the switch between the load and the source (enable switch), ON and OFF.
- Typically, the PWM (Pulse Width Modulation) signal frequency for controlling DC Motors is ~490-1000 Hz.
- More information can be found [here].



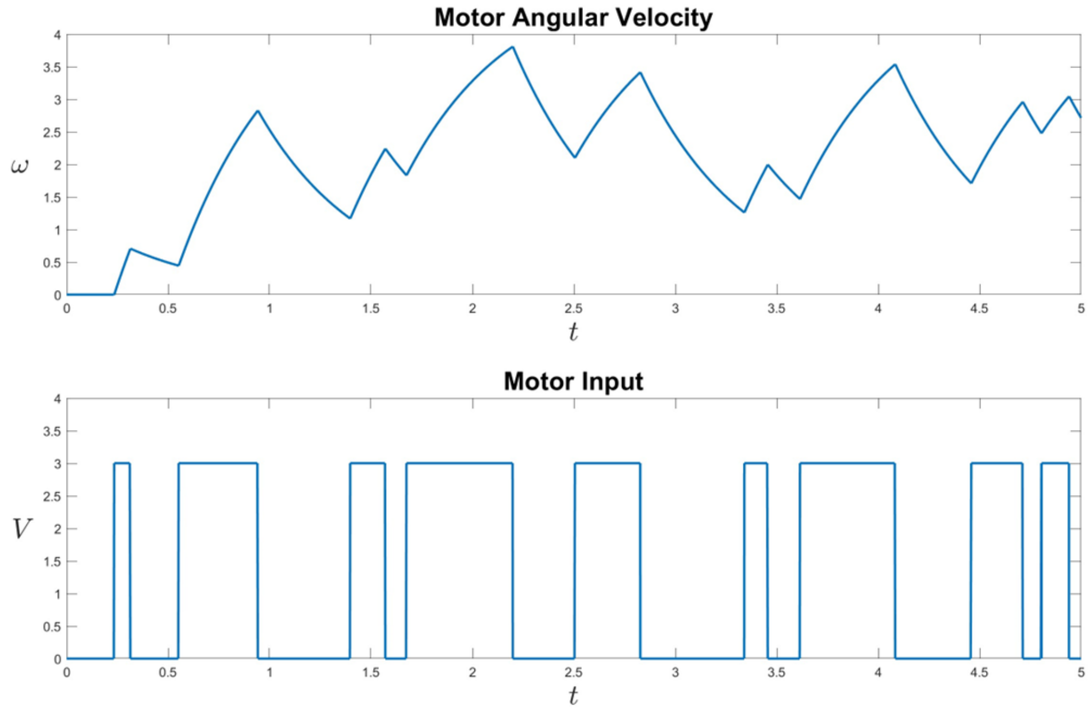
PWM Input



PWM Duty Cycle

Motor Driver

- Given that the motor can be modelled as a second order systems, when applying a PWM voltage as an input, it is possible to observe the output behaviour as in the figure.
- This behaviour, can be used to control the power given to the motor and therefore controlling the motor angular speed.

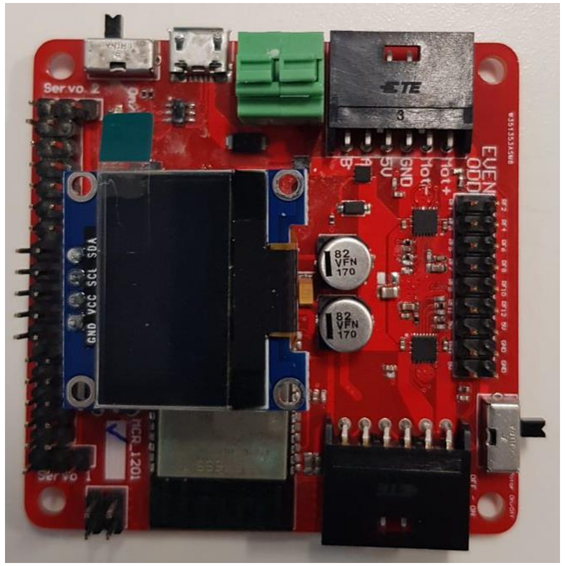


Motor Output/Input

MCU

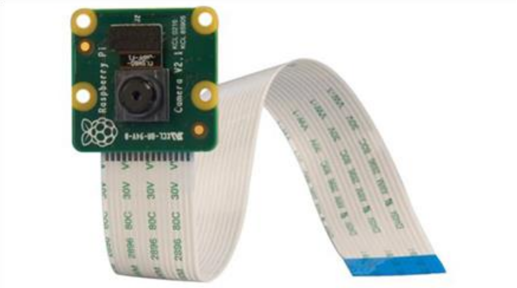
- A microcontroller is a compact integrated circuit. They are made to perform a specific operation in an embedded system.
- In robotics they are usually in charge of the low-level control of the robot, such as motors, and sensors or actuators that require a dedicated and fast controller to work.
- A typical microcontroller includes a processor, memory and input/output (I/O) peripherals on a single chip.

- For the case of the PuzzleBot:
  - ESP32-based Microcontroller
  - Xtensa dual-core 32-bit LX6 microprocessor
  - 520 KB of SRAM
  - WiFi & Bluetooth
  - DC-DC Converter
  - Motor Driver
  - 0.96” I2C LCD Display



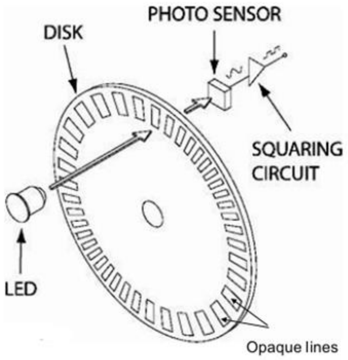
Robotic Sensors

- For the case of the sensors, they can be classified as Exteroceptive and Proprioceptive.
- **Exteroceptive**: Used to measure the environment or the state of the environment, topology of the environment, temperature, etc. Some examples are Sonar, LiDAR, Light sensors, bumper sensors, magnetometers.
- **Proprioceptive**: Used to measure the state of the robot such as wheel position, velocity, acceleration, battery charge, etc. Some examples include, encoders, battery level, gyrometers, accelerometers.

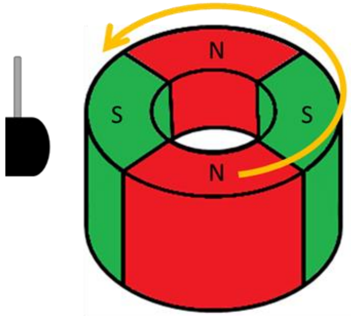


Encoders

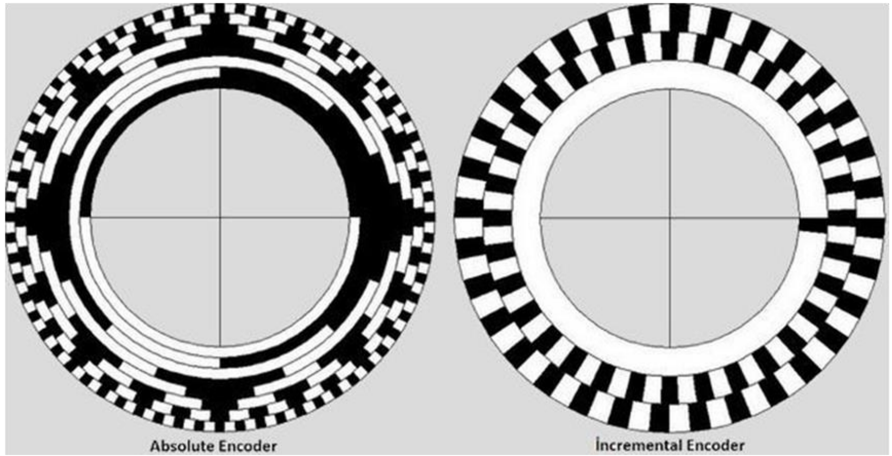
- Device that converts the angular position of a shaft (motor shaft) to an analogue or digital signal.
  - **Absolute:** Indicates the position of the shaft at all times, by producing a unique digital code for each angle (Angle transducers).
  - **Incremental:** Record the changes in position of the motor shaft with no indication or relation to any fixed position of the shaft.
- Encoders in mobile robots are considered proprioceptive sensors because they only acquire information about the robot itself, not the structure of the environment.



Optical Rotary Encoder

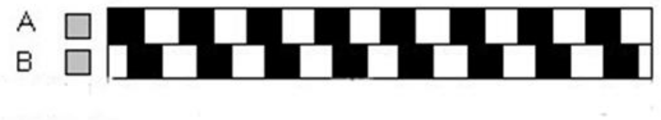


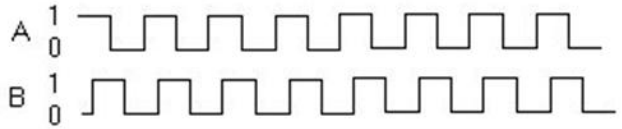
Magnetic Rotary Encoder



Encoders

- Incremental encoders, produce a series of electrical high-low pulses. These pulses, allows to obtain information such as the angular rotation of the shaft or the angular speed of the motor by counting the number of pulses that occur in a certain period (Δt) .
- In robotics, when an encoder is attached to the axle of each wheel in a differential-drive robot, it is possible to convert the number of pulses into useful information, such as the velocity or distance travelled by each wheel.
- With a single set of pulses (single channel / Channel A), it is impossible to know if the motor is rotating clockwise (CW) or counterclockwise (CCW).
- Therefore, a second line (dual channel / channel B) is attached, having its signal shifted by 90 electrical degrees (°e) with respect to channel A.

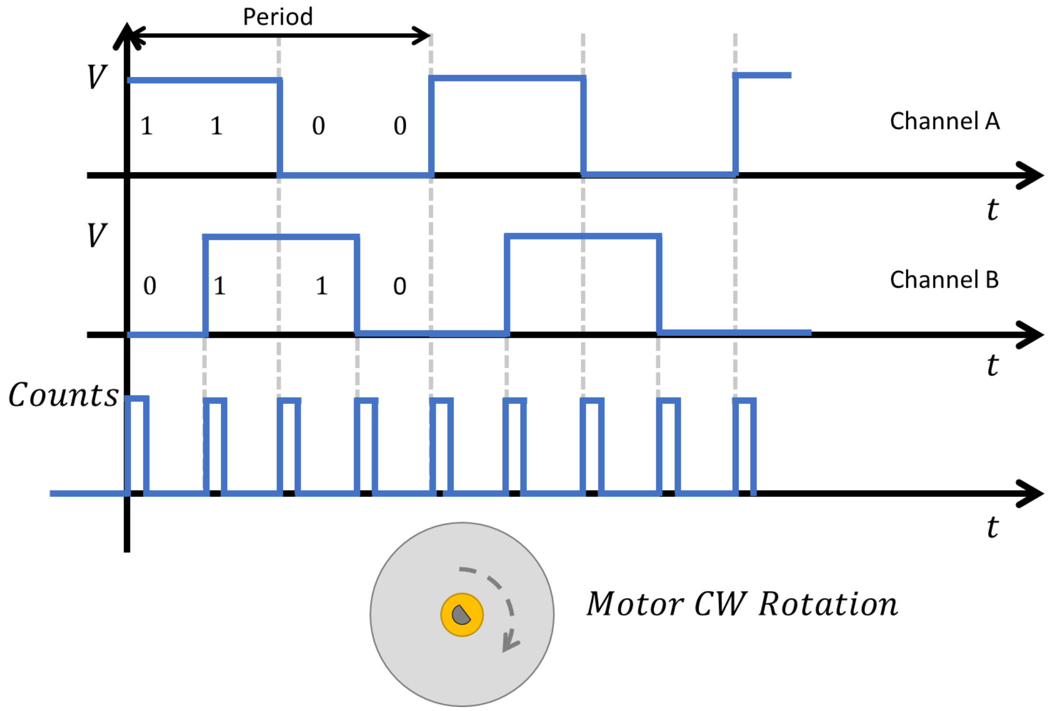




Quadrature Encoder Output

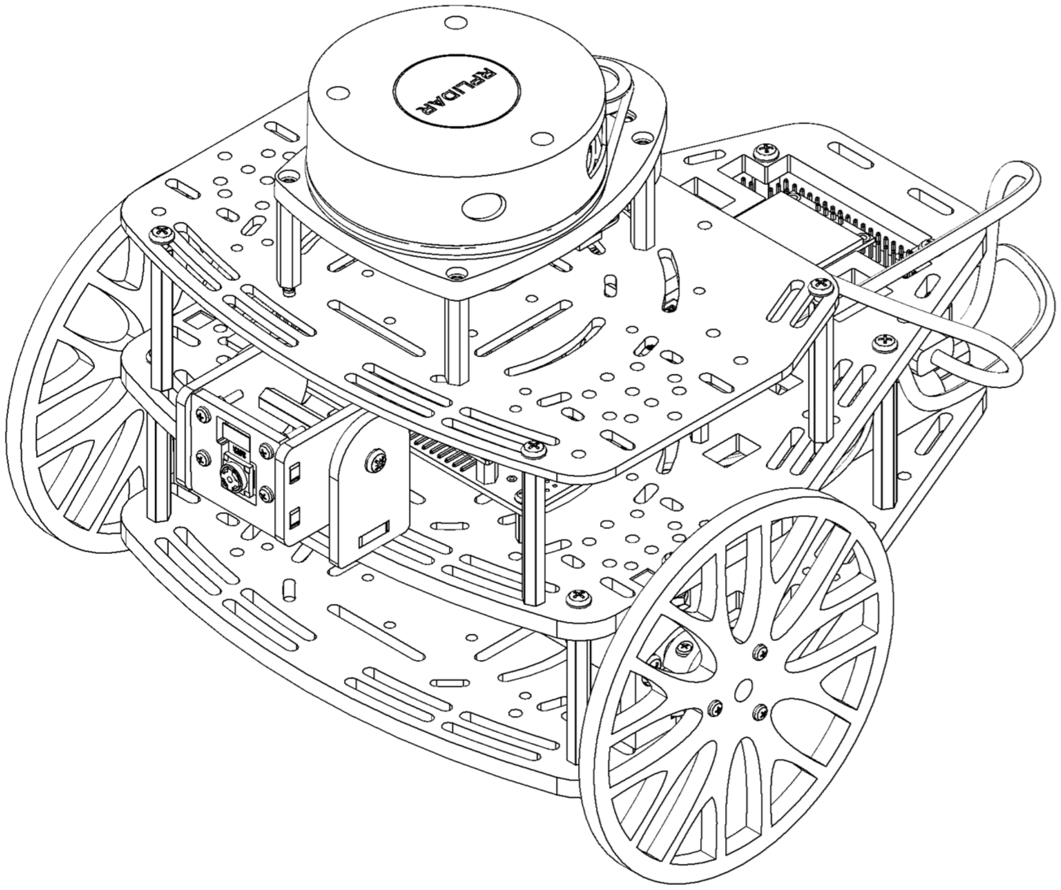
Encoders

- This phase shift allows to determine the direction of rotation.
- Depending on direction of rotation the signal of channel A is preceding channel B or vice versa.
- A simple estimation of the direction would be to verify the previous inputs (Channels A and B) and compare it with the actual inputs, the "code" will change dependant on the rotation direction.
- Counting the pulses of both channels, also leads to a more accurate angular velocity estimation.
- To count the pulses using a MCU, the most common methodology is to use interrupts so that no information is lost.



Encoders

- **Puzzlebot motors use an incremental dual channel, quadrature encoders with 13 pulses per revolution, attached to the motor shafts before the reduction (35:1).**
- The encoder is used to estimate the speed of the motors.
- Since the encoders can have a lot of noise, it is recommended to use a filter (low pass or band pass) to avoid having a noisy signal that affects the controller.



DC Motor: Abstract Model

- Using the previous information, the motor can now be modelled as a single box, where the input is the Voltage (V) and the output are the pulses given by the encoder, as depicted in Figure 1.
- Furthermore, since the pulses allow us to estimate the speed using an MCU, it is possible to abstract this model even more, as shown in Figure 2.
  - Please note that the MCU has enough computational power to perform other things simultaneously, such as control, reading other sensors, etc., NOT only reading encoders.
- As a result, it is possible to describe the motor in terms of its inputs and outputs: the Voltage (V) and the angular velocity (ω) in $\left[\frac{\mathrm{rad}}{\mathrm{s}}\right]$.

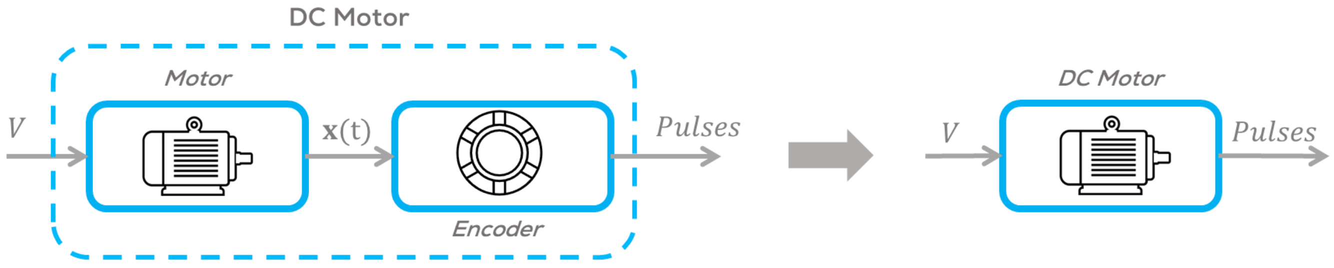


Figure 1

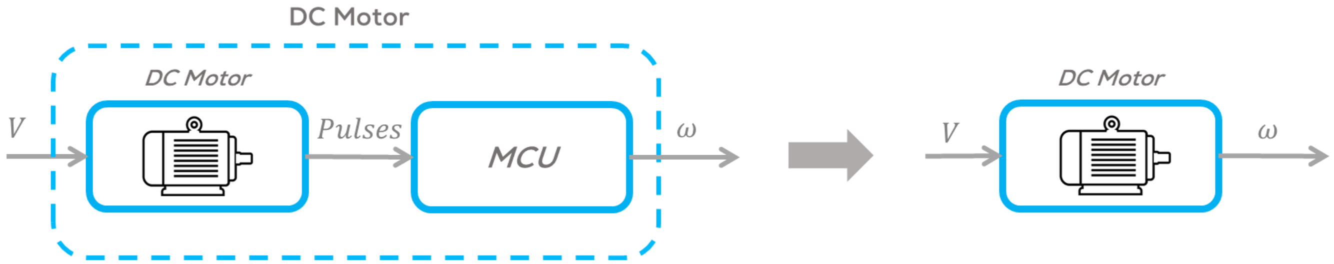
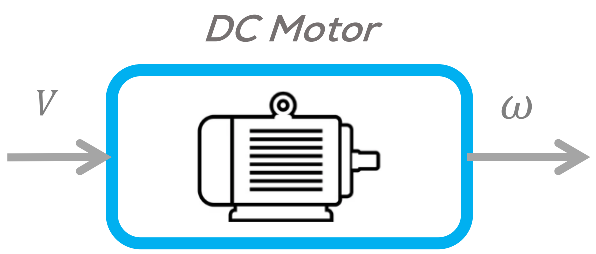
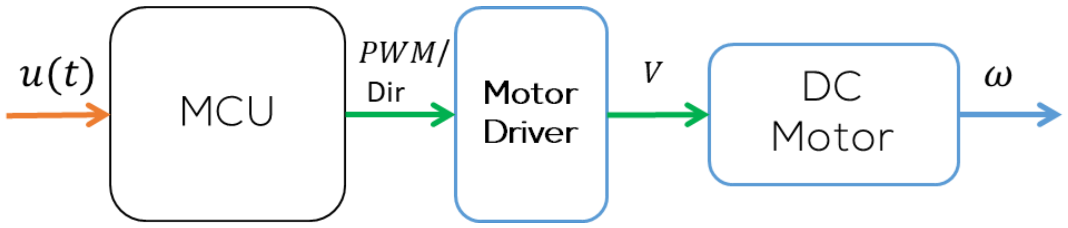


Figure 2

Problem Statement

- Let the system (in this case, the DC motor) be represented as a black box model shown before. The input to the motor is Voltage (V) and the output is the angular speed (ω).
- As stated before, to be controlled by a microcontroller (MCU), a Motor Driver is required to increase the power of the MCU outputs, allowing it to change direction and regulate the speed.
- The motor driver is then controlled by the MCU using digital signals and a PWM signal described previously.
- In this case, it can be said that the MCU acts as a "translator", transforming the input $u(t) \in [0,1]$ to PWM signal.





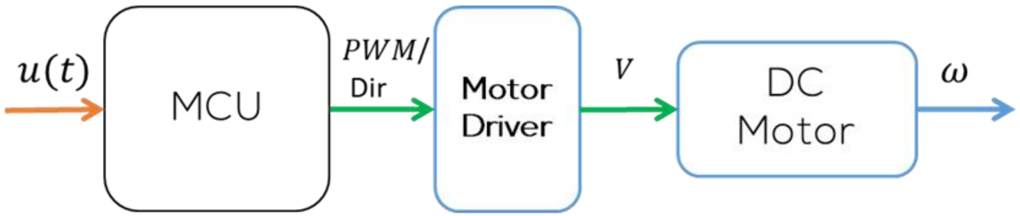
**Please note that the MCU has enough computational power to perform other things simultaneously, such as control, reading other sensors, etc., NOT only reading encoders.

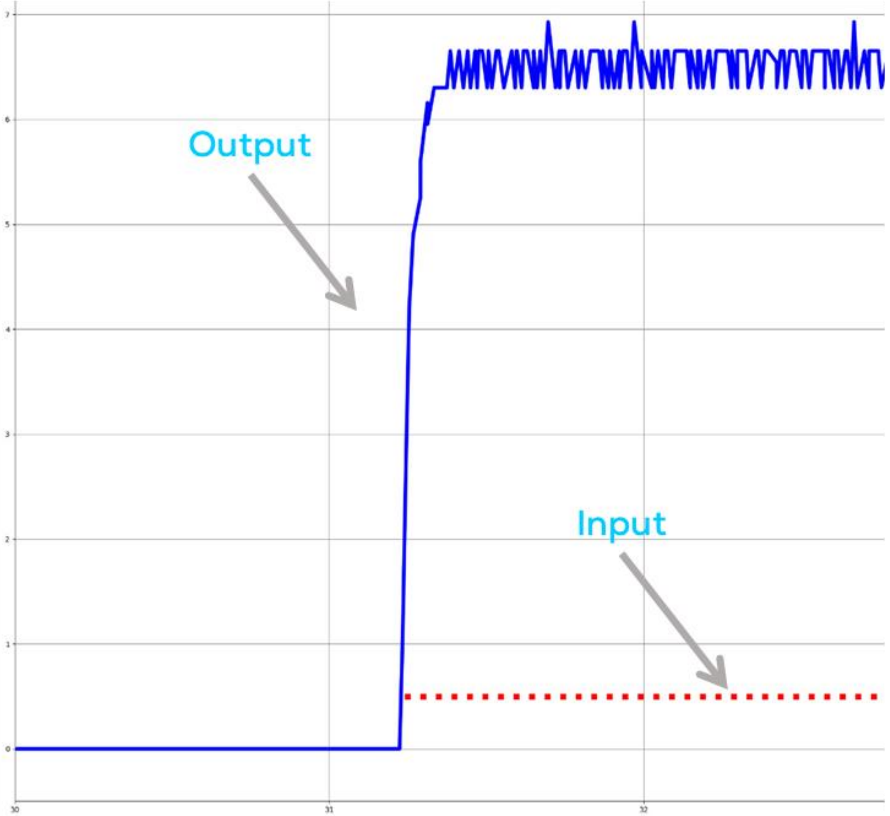
Problem Statement

- Let a value $u(t) = 0.5$ to be the input to the system (step input).
  - Note: This experiment was performed with a MCR2 DC Motor; for other motor types, the response can differ.
- It can be observed that the output of the tends to a steady state and resembles a first/second order system without delay. For simplicity we will use a first order system of the form

$$G(s) = \frac{K}{\tau s + 1}$$

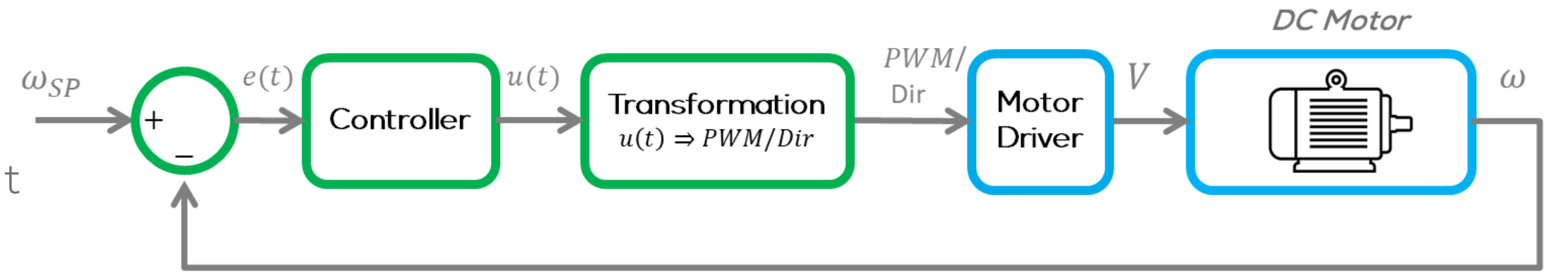
- The figure shows the output velocity to be $6.6\frac{rad}{s}$. Making the gain K of the system to be $K = \frac{\Delta\omega}{\Delta u} = 13.2$.
- The parameter τ (since this is a system without offset and no delay), is obtained by measuring the time when the 63.21% of the total output change $(\Delta\omega)$ is present i.e., $\tau = t_{(0.6321 \cdot \Delta\omega)}$. For this system, the $\tau \approx 0.04$





Problem Statement

- Knowing the system's behaviour, a question arises. Can we use the input $u(t)$, and output ($\omega(t)$) information to drive the motor speed to a desired value (ω_{SP})?
- In other words, would it be possible that by continuously monitor and compare the output (ω) to a desired output value (ω_{SP}), use this information to adjust the system's input ($u(t)$) to drive the motor to the desired speed?
- The answer: YES!

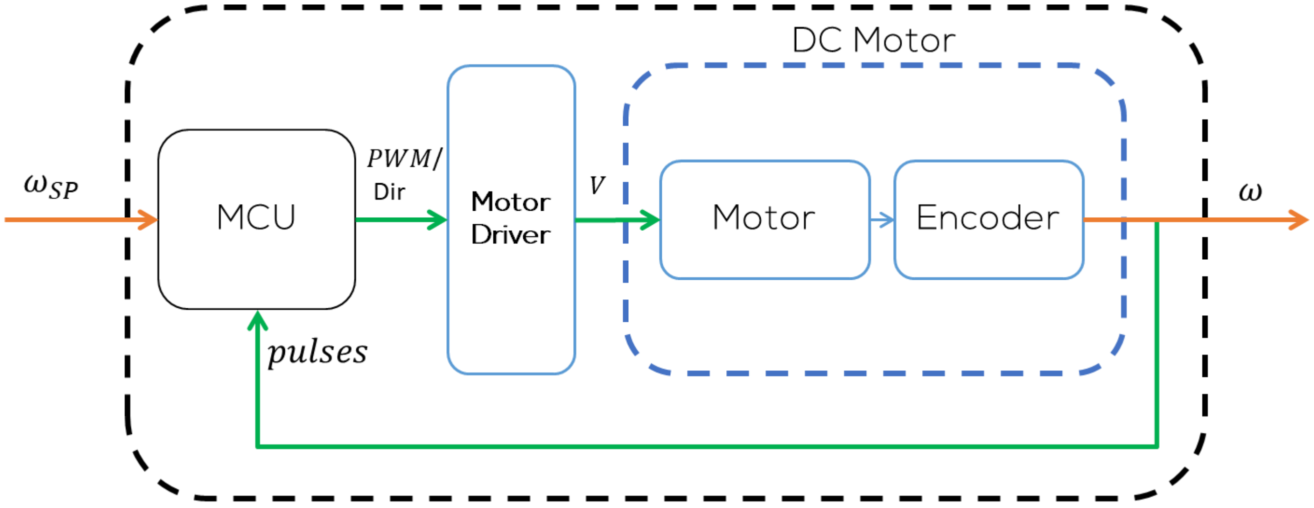


This is called closed loop control.

Closed Loop Control

Introduction

- Closed-loop control systems are widely used in various fields, including engineering, manufacturing, robotics, and automation.
- They offer the advantage of being able to respond to changes and disturbances in the system, allowing for improved accuracy, stability, and adaptability.
- By continuously adjusting the control inputs based on real-time information, closed-loop control systems can maintain desired performance and minimize the effects of external factors that could impact the system's behaviour



DC Motor Control Diagram*

DC Motor Speed Control

Introduction

- In robotics, DC motors are at the heart of countless devices and applications, where precise speed control is crucial.
- In fields like engineering and science, control often refers to the manipulation of variables to maintain stability or desired behaviour in a system.
- Control systems use feedback loops to adjust variables based on input, helping maintain a specific state or achieve a particular goal.
- We'll explore how a simple velocity control system can make a significant difference in the performance and efficiency of these motors.

DC Motor Speed Control

Simple Velocity Control for a DC Motor

- One of the most common controllers used for controlling the speed of a DC motor is known as PID control.
  - Where the "P" stands for proportional, "I" for Integral and "D" for derivative.
  - The most common combinations of this controller are PI (only proportional and integral), PD (Only proportional and derivative) and PID.
- This controller is widely used in industry for controlling Motors and other actuators.
- In mobile robotics, the PID provides a good and accurate velocity control, essential for achieving precise movement in a variety of applications such as moving the wheels of the robot.

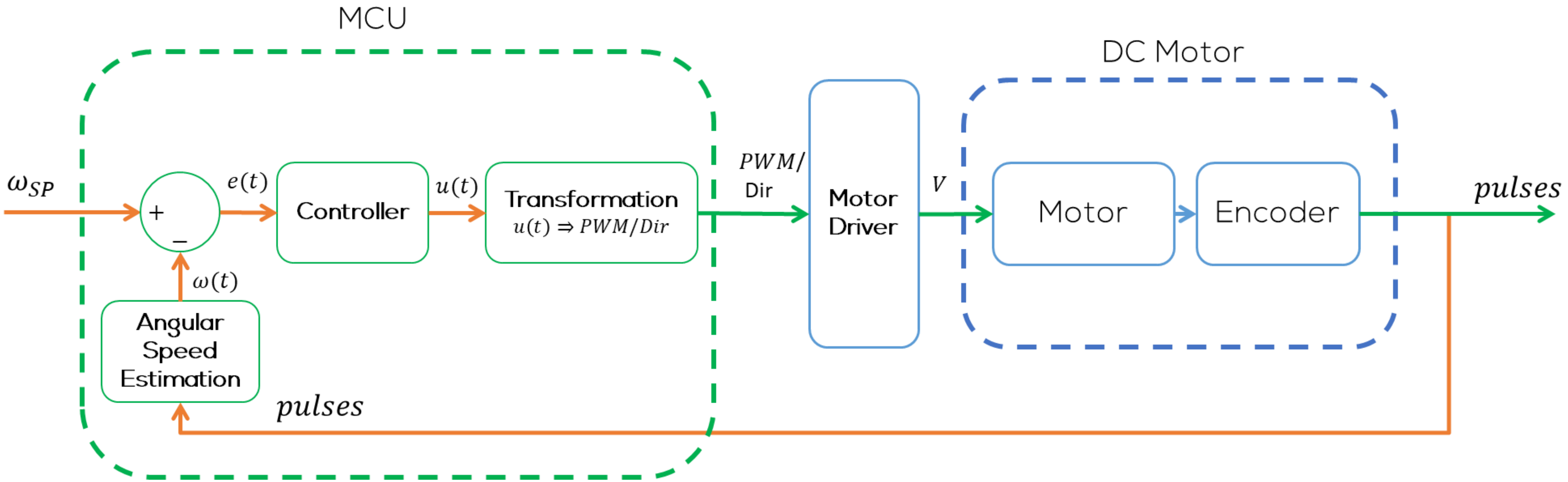
The discrete PID controller is the following:

$$u(k) = K_p e(k) + K_i T_s \sum_{n=0}^{k} e(n) + K_d \frac{e(k) - e(k-1)}{T_s}$$

where $u(k), e(k)$ are the controller output and error at time step k, such that time $t = kT_s$ where T_s is the sampling time. K_p, K_i, K_d are the proportional, integral and derivative gains, respectively. More information here.

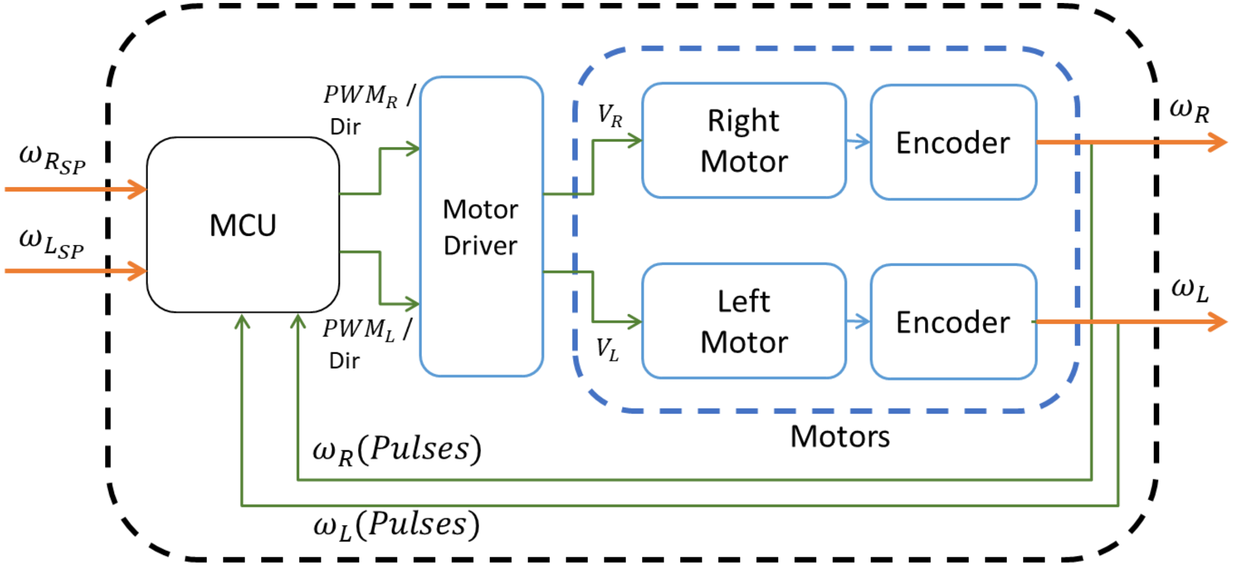
DC Motor Speed Control

Simple Velocity Control for a DC Motor



DC Motor Speed Control

- For the real robot, a closed control loop for each of the motors is required, to move the robot and reach the required velocities, set by the user.
- As stated before, for the case of a wheeled mobile robots is a common practice to implement a PID control.
- It can be observed that the inputs and outputs are the same as the one for a single motor..



Real Robot Diagram*

Q&A

Questions?

{Learn, Create, Innovate};

{Learn, Create, Innovate};

Thank you

MCR2
Manchester Robotics

T&C

Terms and conditions

{Learn, Create, Innovate};

Manchester Robotics

Terms and conditions

- *THE PIECES, IMAGES, VIDEOS, DOCUMENTATION, ETC. SHOWN HERE ARE FOR INFORMATIVE PURPOSES ONLY. THE DESIGN IS PROPRIETARY AND CONFIDENTIAL TO MANCHESTER ROBOTICS LTD. (MCR2). THE INFORMATION, CODE, SIMULATORS, DRAWINGS, VIDEOS PRESENTATIONS ETC. CONTAINED IN THIS PRESENTATION IS THE SOLE PROPERTY OF MANCHESTER ROBOTICS LTD. ANY REPRODUCTION, RESELL, REDISTRIBUTION OR USAGE IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF MANCHESTER ROBOTICS LTD. IS STRICTLY PROHIBITED.*
- *THIS PRESENTATION MAY CONTAIN LINKS TO OTHER WEBSITES OR CONTENT BELONGING TO OR ORIGINATING FROM THIRD PARTIES OR LINKS TO WEBSITES AND FEATURES IN BANNERS OR OTHER ADVERTISING. SUCH EXTERNAL LINKS ARE NOT INVESTIGATED, MONITORED, OR CHECKED FOR ACCURACY, ADEQUACY, VALIDITY, RELIABILITY, AVAILABILITY OR COMPLETENESS BY US.*
- *WE DO NOT WARRANT, ENDORSE, GUARANTEE, OR ASSUME RESPONSIBILITY FOR THE ACCURACY OR RELIABILITY OF ANY INFORMATION OFFERED BY THIRD-PARTY WEBSITES LINKED THROUGH THE SITE OR ANY WEBSITE OR FEATURE LINKED IN ANY BANNER OR OTHER ADVERTISING.*