

Introduction to Robotics

Lecture 3: Bringing the Robot to Life

1. 10. 2018

ParaDiSe

Today Goal

- finalizing mechanics of the robot
- making all the components to work (independently)

Next week goal:

- fix imperfections discovered today (via bare metal programming)

Usefull Links From Last Lecture

- [Arduino Uno pinout](#)
- [Ultrasonic sensor datasheet](#)
- [Arduino reference](#)

Few Words On Robot Powering & flashing

The robot can be powered:

- via USB cable (computer, **power bank**)
- by bringing 5V to the 5V pin
- by bringing 7–15V to the input power jack or screw terminal

The current consumption of the robot should be less than 500 mA.

The robot can be flashed:

- via ISP connector and a specialized programmer
- via serial line through USB connector
- via serial line through Bluetooth module (not part of the kit)

Best options: use a long USB cable or get a Bluetooth module.

Task 1 – Turn the Motors

Baby steps; write a simple program that:

- turns the wheels to the both directions
- controls the speed of the wheels (`analogWrite` might help you)

Final Task: Write a program that allows you to control the robot using the WASD keys on your computer.

Task 1 – Observations

Observation: The motor starts to rotate at 75

- `analogWrite` produces PWM @ 480 Hz
- different motors react differently to frequencies
- our motors work best at low frequencies (100 Hz and less)

Task: Try it yourself, implement low frequency PWM in software (can be done easily).

Solution: We will show next week how to implement HW PWM at given frequency

Task 2 – The Ultrasonic Sensor

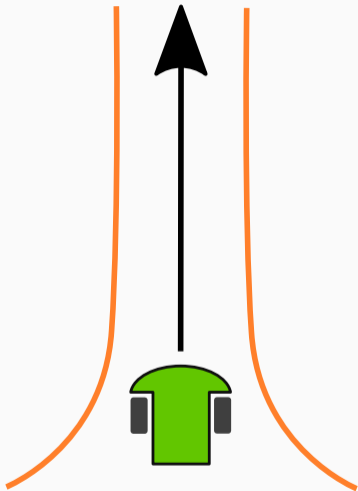
Write a simple program that repeatedly prints to the serial terminal distance in centimeters from the sensor.

- read the datasheet (link previously in the slides) to see the measurement procedure
- `pulseIn` might help you
- speed of sound in the air at 25 C is $34 \frac{\text{cm}}{\text{ms}}$

Then test the sensor

- does it work with small objects?
- how does it behave with soft/fluffy stuff (e.g. a soft toy)
- how does it behave in corners?
- how does it behave when two sensors look at each other?

Task 3



Program the robot such that:

- given a long corridor the robot can go through the corridor not hitting the walls
- it is not hard-coded to a fixed corridor width
- feel free to start with a cloggy motion, then try to achieve smooth motion
- hint: you have an ultrasonic sensor on rotational mount