Admin

- Quiz due Friday at 11:59pm
  - One hour time limit (so you need to start by 11:00pm!)
  - Short answer, fill in the blank, code snippets
  - Open book; open notes. Closed compiler.
  - If you're given a score, it is only preliminary
  - Quiz covers:
    - Coding/programming
    - Intro material on sensors, robots, what is a robot, etc.
    - (No "DoF")
- Quiz review, 6-8pm, Higgins 116
  - The intention is to have two, one-hour reviews.

Admin

- HW
  - Solutions to 1.* are on canvas
    - Note that we're only grading select problems
  - 2.0 due tomorrow at the start of class;
  - 2.1, 2.2 due Monday at the start of class
- Read Chp. 4 & 5 for tomorrow. They give a great overview of actuation.
- Friday, 3-5: another "ESP32 Workshop"
  - (take a moment to find a partner...)

rotateTo vs. rotateFor

Functionality

Defining Functionality

- Actions of the system
- Solution independent
- For illustrative purposes, we often define them as verb:noun pairs
  - (but “be <something>” is typically *not* a good functional definition)

DigSafe Robot

You work for a large electrical company, which commits a lot of resources to marking out electrical lines so that when contractors are digging, they don't dig up your wires. You are aiming to make the process more efficient, and one of your brilliant ideas is to make a robot that can survey an area for buried cables and mark them, much as is done by a person now. You have GIS information on where lines *should* be, but they aren't always accurate.

Define the functionality of such a robot.

How you define functionality affects your solution space

- Consider the difference between:

Less restrictive	**More restrictive**
Move around workzone	**Drive around workzone**
Store data	**Transmit data**
Measure soil temperature	**Insert probe**

Sit with your team!

- Take a few minutes to compile all of your individual functions into one list for your team. Write them on a fresh piece of paper (so you can turn in the individual ones).
- Make the list as comprehensive as you can.

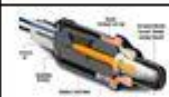
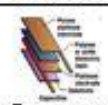
Evaluation time

- As a team, evaluate the functional specifications for the submission you've been given.
  - Put a check mark by ones you agree with
  - If you disagree with one, explain why you disagree with it
  - Write the team number and names of the evaluators at the bottom

Collective list of functions

Morphological chart

P16414: Toilet Use Sensor – Updated on March 2st, 2016

Function	Importance	Possible Solutions						
Trigger	9	Physical Volume Calculation	 Motion Sensor	 Proximity Sensor	 Heat/Infared Sensor	 Optical Trip wire	 Pressure Sensor	 Noise Sensor
Microcontroller	9	 Arduino	 Rasberry Pi	 MSP 430				
Temperature	9	 Thermocouple	 RTD	 Thermistor	Silicon based	 Thermometer	 Pressure Spring Thermometer	
Oxygen Levels	3	Dissolved Oxygen Sensor	Colorimetric	 Titration	 Heated	Unheated	 FLO	 Planar
pH	9	Electrode Methods	 Color Indicator Comparison	Ion Selective Field Effect Transistors	Semi-Conductor Methods			
Humidity	9	 Capacitive	 Resistive	Thermal Conductivity	 Psychrometer	 Hygrometer	 Dew Check	
Data Extraction	9	 Book	Wirelessly	 USB	 Bluetooth	 Disk	 SD Card	
Protection	9	 Waterproof Case	 Waterproof Bag	Magnetic Force Field				

Assignment

- Generate a comprehensive set of ideas for each of the functions of your robot
  - Concentrate on the functions where you have lots of options
  - It's OK to have just one "idea" if that solution has been provided for you (e.g., the VEX radio will be used for communication)
- Create a morphological chart for your ideas
- For each function, select 1 – 3 ideas that you are most likely to pursue
  - Why do you think each solution is promising?
  - What would you still need to figure out to implement it?

Functional advice

- Organize your project by functionality (including report!)
- Generate lots of ideas for each function
- Evaluate your options by:
  - Constraints & requirements
  - Performance
- Use physical prototypes and analytical tools wisely
- Be conscious about trade-offs
  - External: e.g., cost vs. performance
  - Internal: e.g., performance vs. complexity