System integration and communication

Imagine…

...you’re working on a design team to build a “mock” robot to deliver, let’s say, late night cookies to dorms at highly regarded technical institution. You’re on a team of three, and for various reasons, you’ve organized your team around three major sub-systems:

- Drive train
- Delivery mechanism
- Sensor system and control

Each team member is the lead on one sub-system. Due to schedules and other constraints, you often find yourselves working on your respective sus-system when the others are away studying for math exams or playing underwater hockey or whatever life they have outside of RBE.

Scenario 1

- The delivery lead realizes that the four-bar, as sleek and beautiful as it is, needs to be longer so that the robot can deliver cookies to the highly profitable top floor of Messenger. She decides to add a segment to the length.
- How does her decision affect the other two sub-systems?

Scenario B

- The drivetrain lead decides that the robot needs a longer wheel base so that the robot is more stable. He adds some length to the front of the bot, and moves the cross members forward, as well.
- How does his decision affect the other sub-systems?

Scenario 3

- The sensor lead decides that a button would be more reliable than the ultrasonic (which is currently on the robot) for detecting when the robot has reached a dorm.
- How might such a decision affect the other sub-systems?

Scenario D

- Your coding guru, freshly returned from a weekend summit on event-driven programming in Davos, decides to redesign the code to use a fancy state machine.
- How might this decision affect the other sub-systems?

Scenario

- Gearing
- Wheel type
- Number of motors
- Length of wire
- Ultrasonic or vision
- Moves the camera
- More supports to lift mechanism

What can you do?

- Keep a design notebook
  - (probably a little late for this)
- Trello / Slack
- Daily email update
  - “I relocated the camera so it doesn’t interfere with the arm.”
- Even better, email *before* you make big changes
  - (but sometimes people are offline)
- Still better, have a quick meeting/discussion before anyone makes significant changes
  - (but sometimes time doesn’t allow)

Upcoming schedule

- Monday quiz: in-class, but the same policy on open notes
- Tuesday/Wednesday: Advanced mechanisms and other cool stuff
- Thursday/Friday: No class. Good days to build, **but don't put off all your work until then!**

- Monday after: Review
- Tuesday: Final, Part I
- Wednesday: CDR in the morning; OED in the evening
- Thursday: Review. Bragging.
- Friday: Final, Part II