Technical Review

The *Technical Review* is a presentation of current technical problems you are encountering on your project to an audience of both your peers and more experienced engineers. It is your chance to talk through and get advice about a problem you are working on in real-time.

Educational Goals:

By delivering a *Technical Review*, you will:

1. Develop and practice skills in communicating current technical problems within your project in a clear and realistic way that elicits constructive audience feedback.
2. Practice candidly presenting technical problems encountered on your project to peers and other experts.
3. Continue to learn to listen to and incorporate feedback by demonstrating openness to feedback, and flexibility to change.
4. Gain experience planning for and running a type of design review that is commonly used in many design and engineering fields.

Description:

The main goal of the technical review is to gather a group of knowledgeable colleagues to provide technical feedback to your team on a focused issue or problem that you are working on. The format of the presentation is up to the team. You should provide relevant technical details and choose something that you are working on in the moment. Decide a few days before (or even the day of) what you will present. A design? A set of designs? Something you are struggling with? What are the failure modes? Use this as an opportunity to share qualitative and/or quantitative analysis of an aspect of your project.

The goal of a *Technical Review* is to have a meaningful exchange with the other folks in the review. This means you need to be candid about what you are having trouble with and understand that the barrage of questions you get from the reviewers are meant to help you. Some scenarios are presented below if you are struggling.

Advice:

- Remember first and foremost that an animated questioning by reviewers and ensuing discussion is not seen as weakness on the part of your team, it is seen as a successful session.
- DO make sure your audience understands the problem you are trying to solve and why. This is an excellent place to start.
- Any visual aids you use (handouts, prototypes, Powerpoint, etc.) should help your audience better understand your problem and your approach.

- While clear communication is a must, DON'T spend a huge amount of time making a glossy presentation.
- If you haven't met with relevant experts already, work with your faculty advisor to get them to your technical review.

Scenarios for Technical Reviews

There should be no point in time when you can't sit down and tell people what you are working on. To help you get a handle on this we've come up with some scenarios and ideas for what to talk about. If you aren't sure what to talk about, ask your faculty advisor or Alisha and we'll help you figure something out.

1. It's the first design review of the year and we've only been doing background research.
   - *What is the current state of the field? Where does your problem fall in that context?*
   - *Where is this going?*
   - *What is your understanding of the problem? Is it primarily an engineering challenge, a design challenge, a research question?*

2. Our team has developed a set of hardware designs and we're ready to order materials.
   - *What requirements does hardware need to meet? Does your current design meet these requirements?*
   - *Explain materials choices in detail.*
   - *What analysis have you done to support your choices?*

3. We're in the middle of gathering data or doing experimental work. We don't need help, just more time. What do we talk about?
   - *What questions are you trying to answer?*
   - *How do you know the experiments you have planned will answer these questions?*
   - *What is the range of expected values?*
   - *Are there published results on a similar problem or experiment?*
   - *Are there standard protocols for the experiments you're proposing/running?*
   - *How will you validate your results?*

4. We're writing code and it doesn't run yet so we have nothing to discuss.
   - *What are the units of functionality the code is intended to provide?*
   - *What does work? Can you demonstrate those pieces? If not, why are you not breaking things down into manageable modules?*
   - *Do a code review – either high level (structural) or low level (line by line).*
   - *How did you get into this position that you are stuck here and the only option is to get this working – take time to reflect technically on what you really want to do.*

5. We've been ideating for 2 weeks but aren't ready to talk to people yet
   - *Yes it's nebulous, and sometimes you loop around, but you have to know where you are.*
   - *Get a design person there to talk through the problem with you.*
   - *Users? Values? Who are you talking to? Can you synthesize what you've learned?*
   - *If it's a more technical/less people problem – still about background, what have you learned? What is your process?*