# Tutorial 5: Midterm and Project Info

# Midterm

# Midterm

- July 3, 5-7pm in SW309
- Covering everything up to/including Lecture 5 (Sequential Circuit Design/FSMs)
- No lecture on day of midterm
  - I'll hold extra office hours during that time instead
- General tips:
  - Be careful
  - Practice tutorial material
  - Be careful
  - Bring a ruler
  - Be careful

# Project

# Hardware project

- 3 Week to complete project
  - Final week for presentations
- Build whatever you want
  - Using FPGAs + Verilog (not software!)
  - We have additional hardware available
  - Look at existing projects on Projects page
  - Can build off existing work (with appropriate credit given, and clearly explain what you add!)
- Teams of 2-4

# Available Hardware

- DE2 Boards (obviously)
- INPUTS
  - Buttons
  - Keyboards
  - Joysticks
  - Sensors (light, touch, vibration, rotation, temperature, Infrared, heat, sound, and lots more)
- OUTPUTS
  - VGA monitors (you'll use these in lab 6)
  - LEDs
  - Buzzers
  - Motors

# Pairing

- Choose groups of 2-4.
  - Bigger group are expected to have more ambitious projects.
- You can stay with your current lab partner.
- You don't have to stay with your current lab partner.
- You all need to work together every week in the same practical (inform TA ahead of time if you will be changing practical sessions)
- Changes to group and PRA will not be granted.

# Reports

- Proposal
  - Initial plan due July 5
    - (Don't worry, we won't be holding you to this plan… much)
  - Project file on website
    - Must complete team member info, proposal and motivations
- Weekly Updates
  - Each week you will update your file with what happened that week and how you plan to proceed. Submit to Quercus (updated file due every week)

# Final presentation + report

- Present your final project to your TA during your last practical (final week)
- This means you won't have time to work during that week
- You will record a presentation and create a short video.
- Final report will include with links to videos, code location, etc. Due August 2.
- With your permission, we will put select projects online for future students.
  - If you don't want it online, just let us know

# Marking

- Project is worth 20% of your final grade
- You will be marked on effort and achievement:
- Effort:
  - Documentation, report file, proposal, comments
  - Video / final presentation
  - Working hard during lab time, and planning
- Achievement:
  - Ambition of project
  - Execution of final product
- Roughly speaking, 50% effort, 50% achievement.

# Tips

- Design will be KEY
  - Good hierarchy
  - Modularity
  - Testing each component
  - See what's available from sources of old projects
- Plan and use your limited lab time wisely
  - Prepare beforehand
  - Know your goals before you arrive
  - Most of your in-lab time should be debugging
  - Backup regularly or use source control

# Tips

- Teamwork
  - Divide tasks
  - Check partners code (its amazing the bugs that can be found by a fresh set of eyes)
- Planning/Scale
  - Your mark will be based on course-related content
  - Hardware will be fun to play with, but don't get so bogged down you forget about core principles
  - Have back-up plans. Build iteratively
  - You don't have to use fancy hardware. There has been impressive projects with just DE2 and VGA.

# Perspective

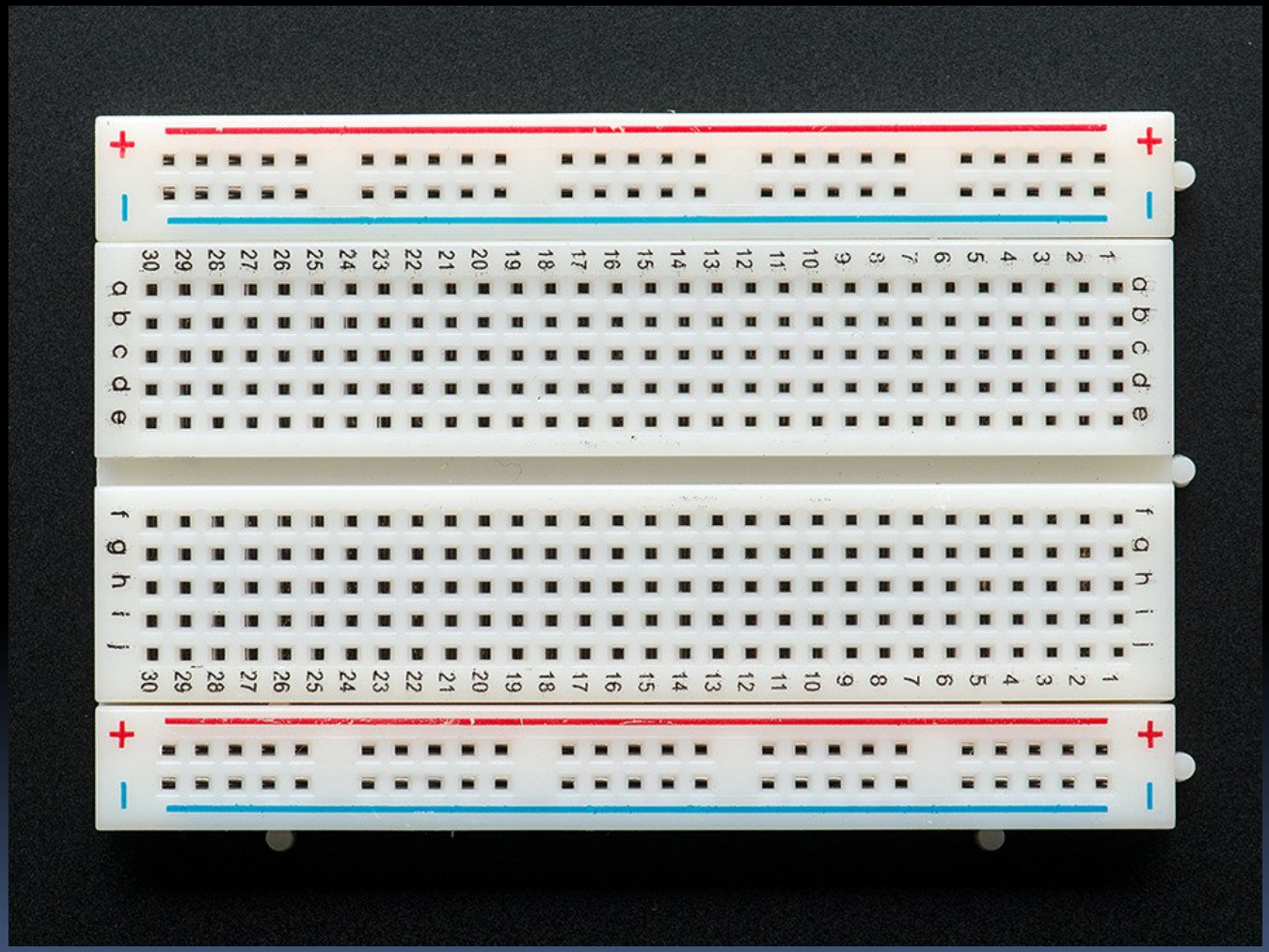
- Maintain perspective
- This project is worth 20% of your grade.
- It is a lot of work.
- Don't get so absorbed that you forget to prepare for exams or submit other assignments.
- Backup regularly or use source control

# Past projects

- Some links on course website
- Searching CSC258 on YouTube will show you other projects
- Remember: You don't have to start from scratch, but you MUST give credit for any code you use, and clarify the novelty (what you added on top of existing work).

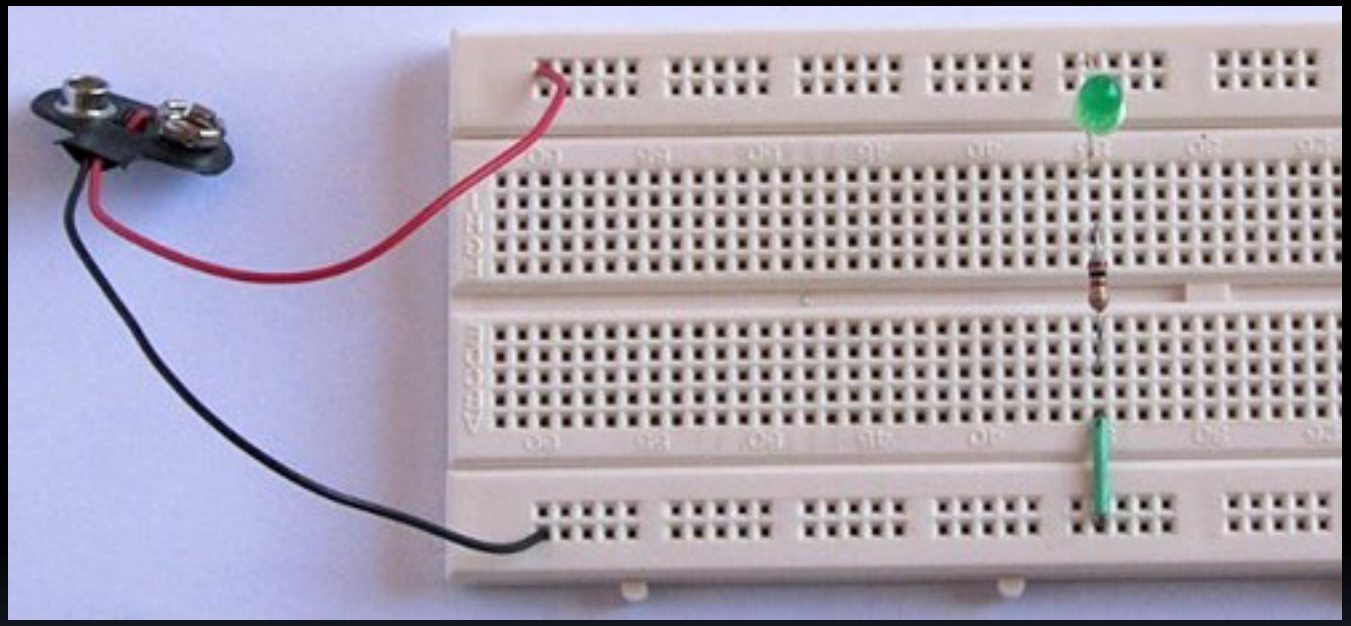
# (Very) Brief Intro to Hardware

# Breadboard



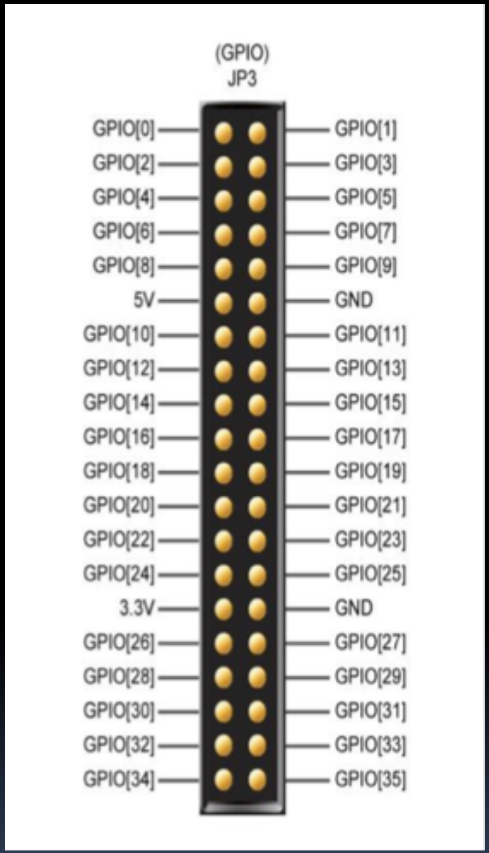
# Breadboard connections

# Creating a circuit



- Instead of a battery, you could hook up to the 5v and GND inputs on the FPGA
- You need a resistor (or hardware that resists)

# Controlling a circuit




- Instead of a constant 1 or 0 (5v or GND), connect to a GPIO port
- (General Purpose Input Output)
- Can turn the port on/off to control the circuit (output)
- Can read from the port to see if circuit is complete (input)

# Good luck