

Intro to Systems Engineering aka Design Modularity

ECE 445



ECE ILLINOIS

 ILLINOIS

What is Modular Design?

- A Design that can be easily broken apart
 - Blocks, Modules, Sub-Systems
- Transform a problem into a system
 - Problem -> Set of sub-problems
 - Sub-problems -> Modules

Why Use Modular Design?

- Make Complexity Manageable
- Enables Parallel Work
- Accommodates future uncertainty

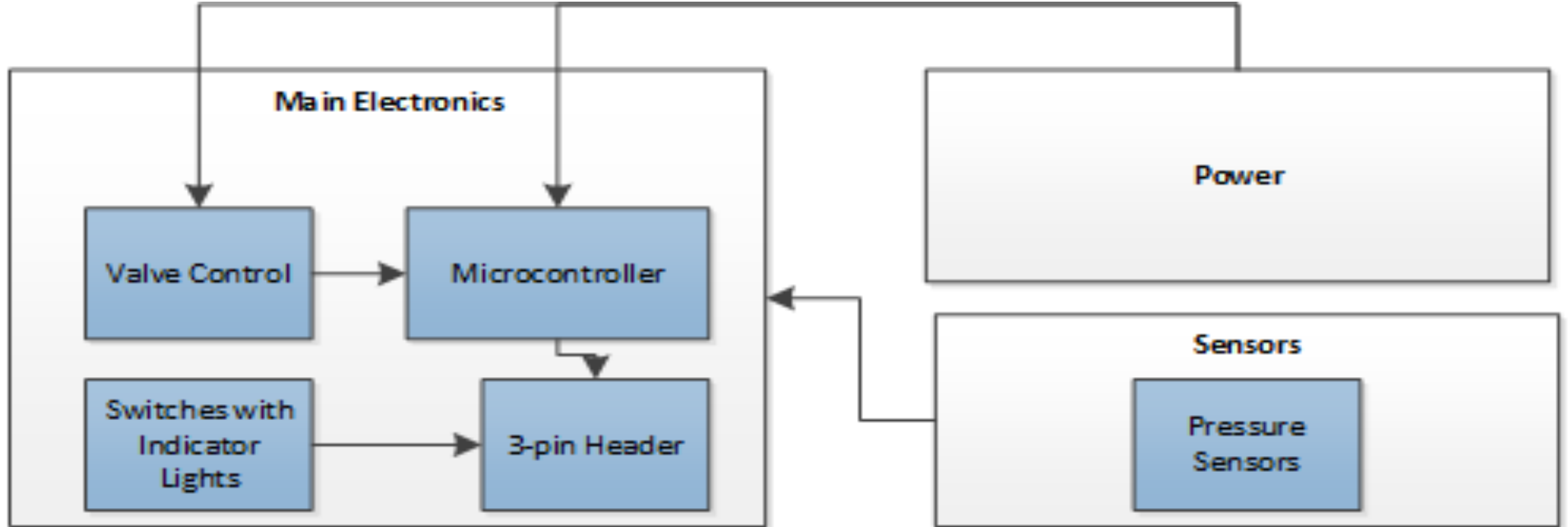
Conceptual Overview

1. Think high level
2. Divide into “modules” or “blocks”
 - a. Allows reader to see what your design does
 - b. Inputs and outputs are per module
(make sure numbers and units match)
 - c. Functionality is important not physical placement!
3. Will formalize into a block diagram for the proposal

Modular Design Example

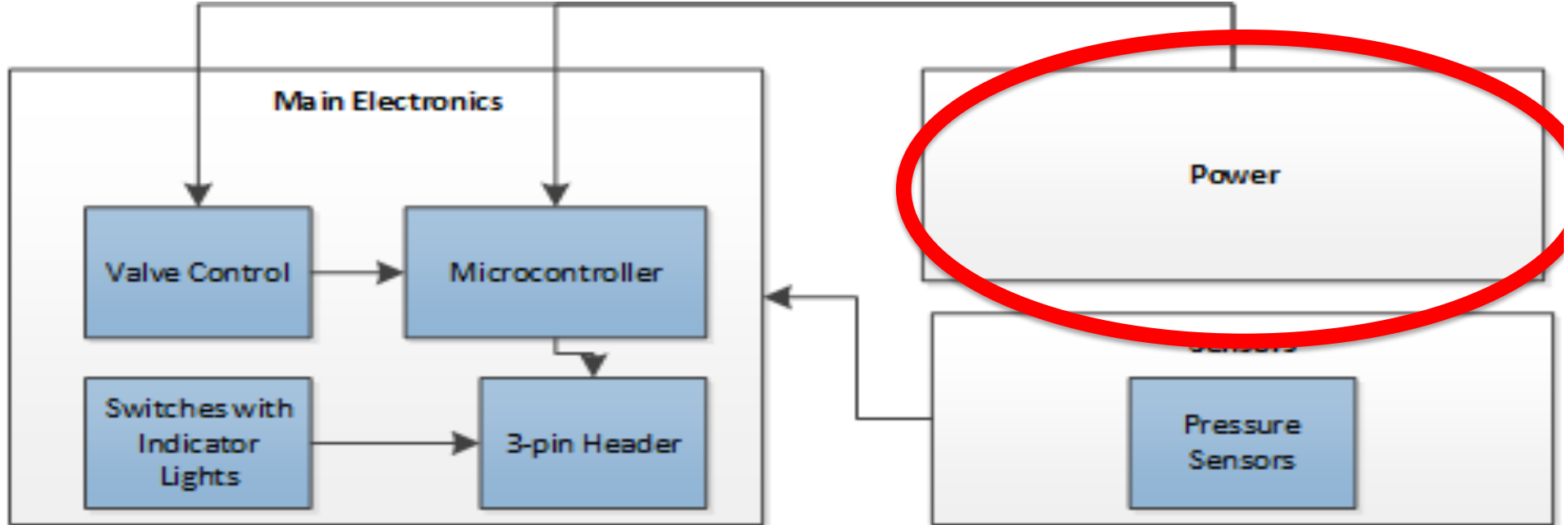
- As an example, let us consider the design of an irrigation controller.
- We want to provide a user with a set of pressure sensitive buttons to open and close irrigation valves.
- What would a modular block diagram for this system look like?

Bad Modular Design



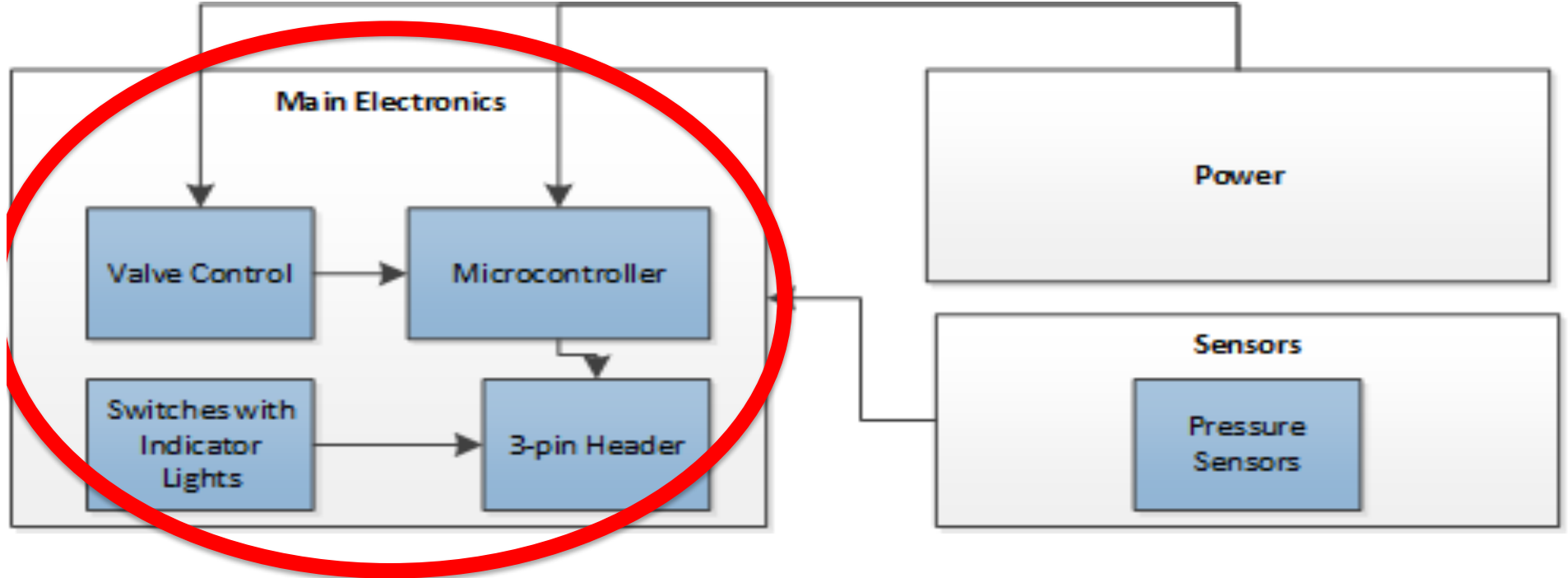
Bad Modular Design

How exactly does the power system interact with the individual blocks? Is power provided to the sensor systems?



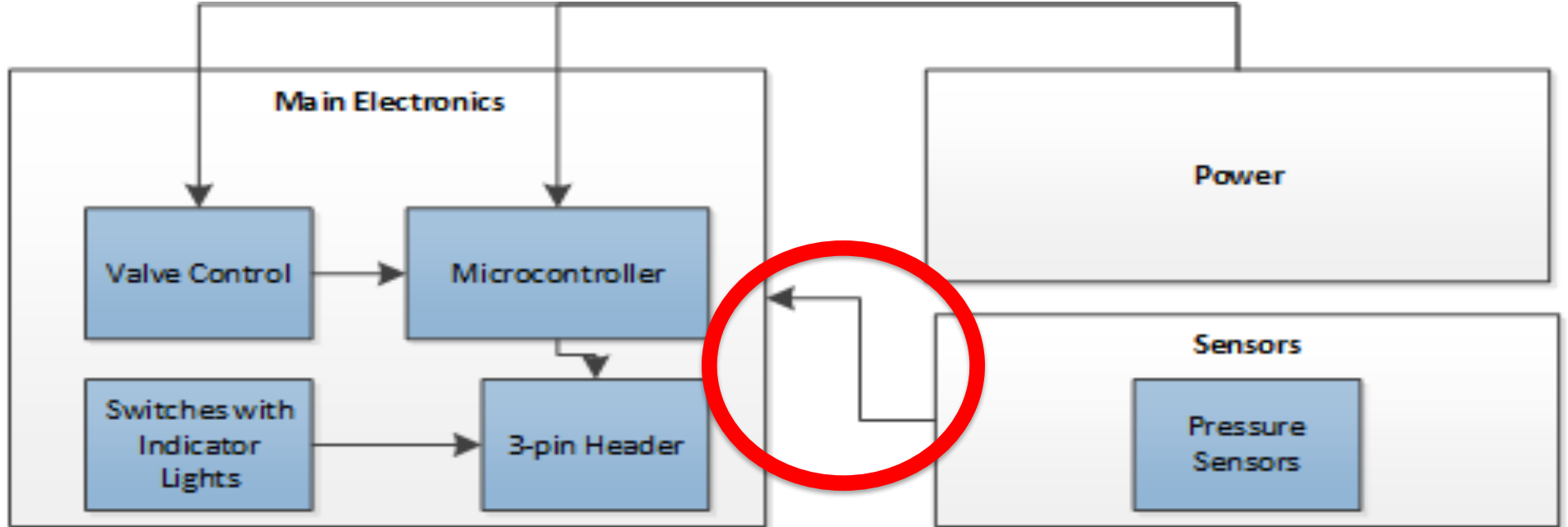
Bad Modular Design

Is 'Main Electronics' broken down into the appropriate subsystems?

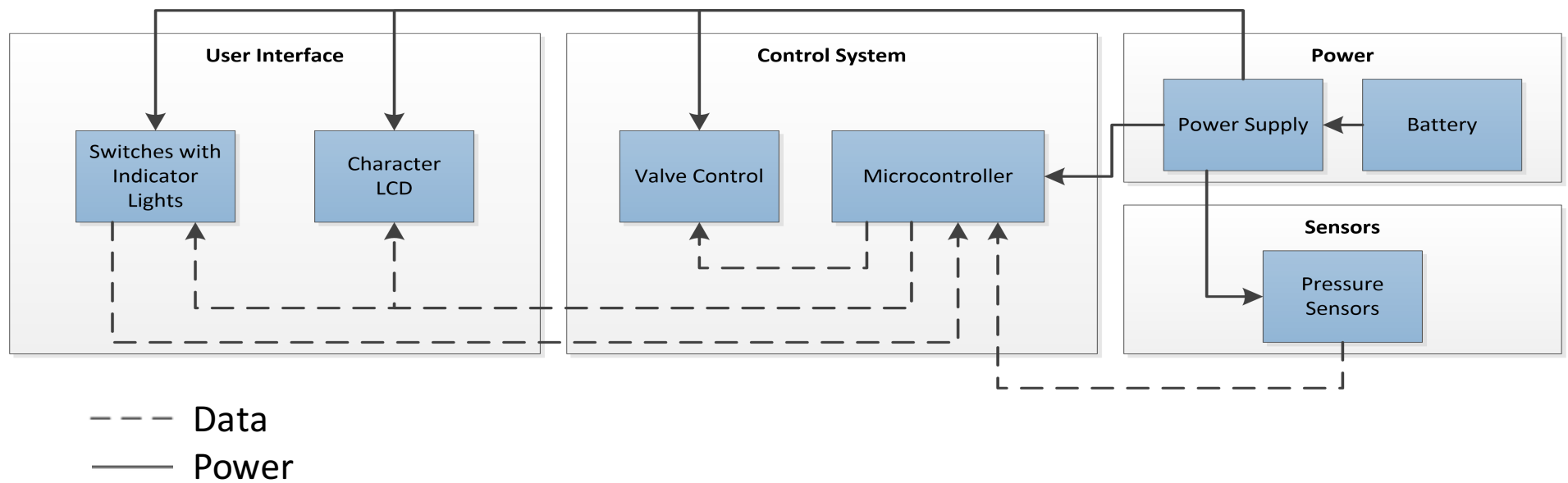


Bad Modular Design

What individual blocks do the pressure sensors actually connect to?

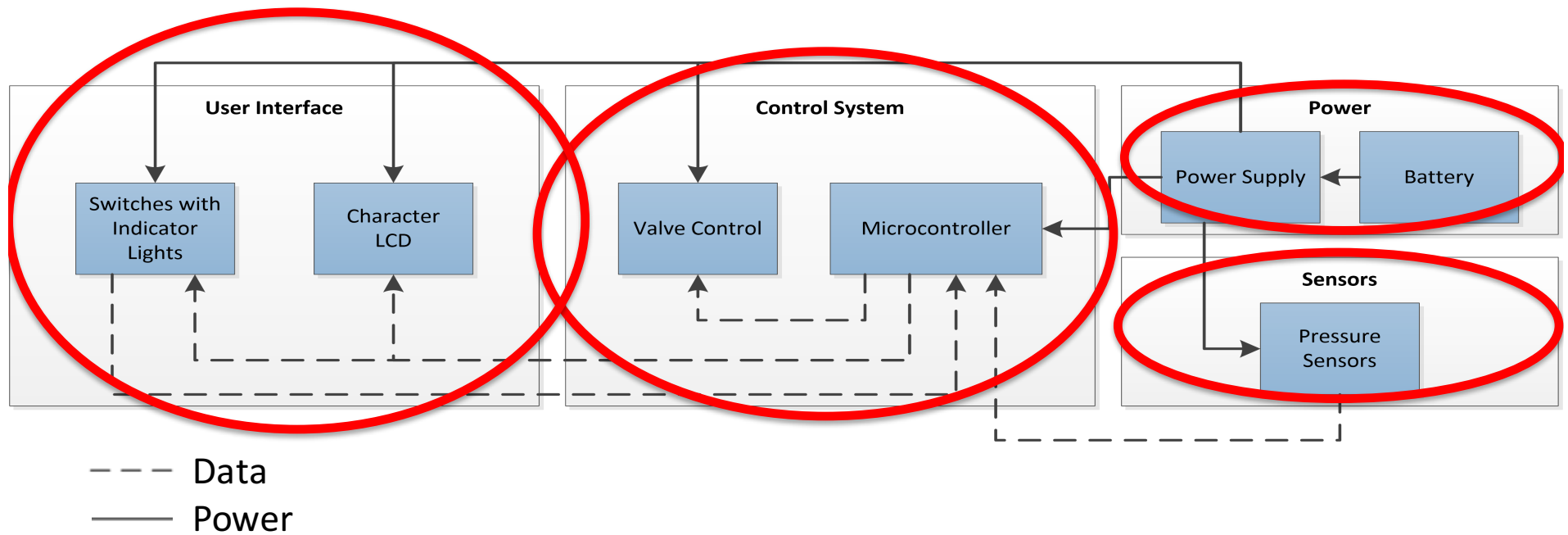


Better Modular Design



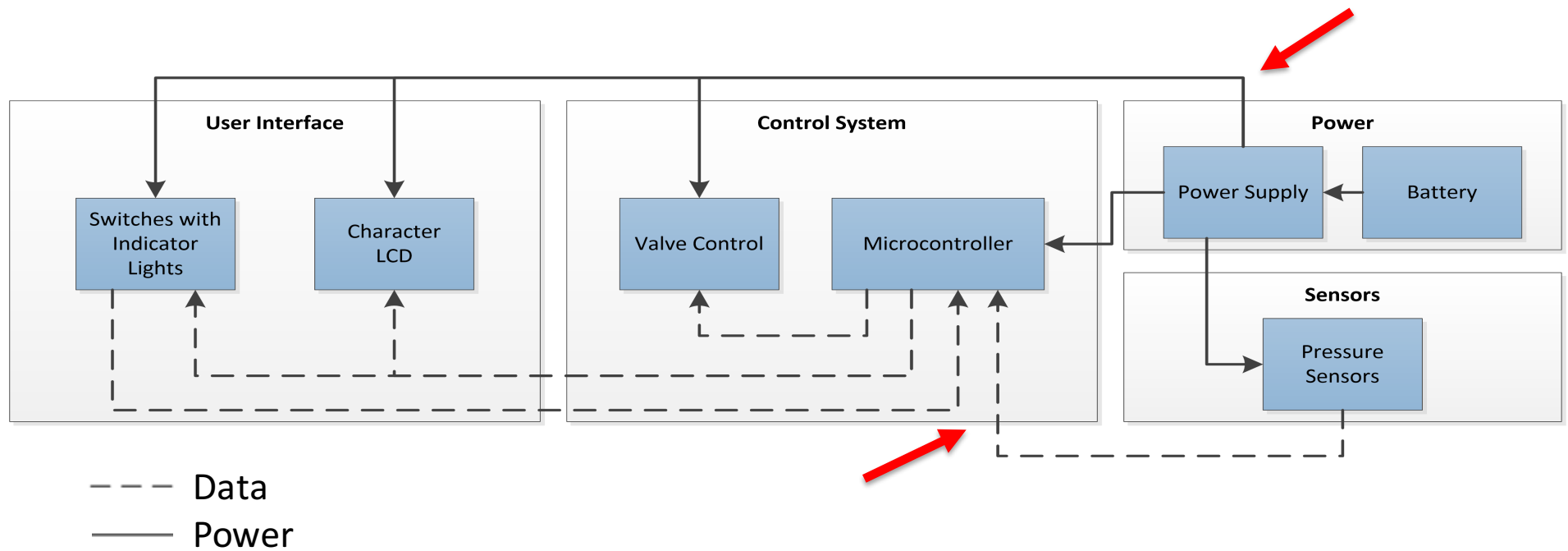
Better Modular Design

The block diagram is neatly divided into appropriate subsystems.



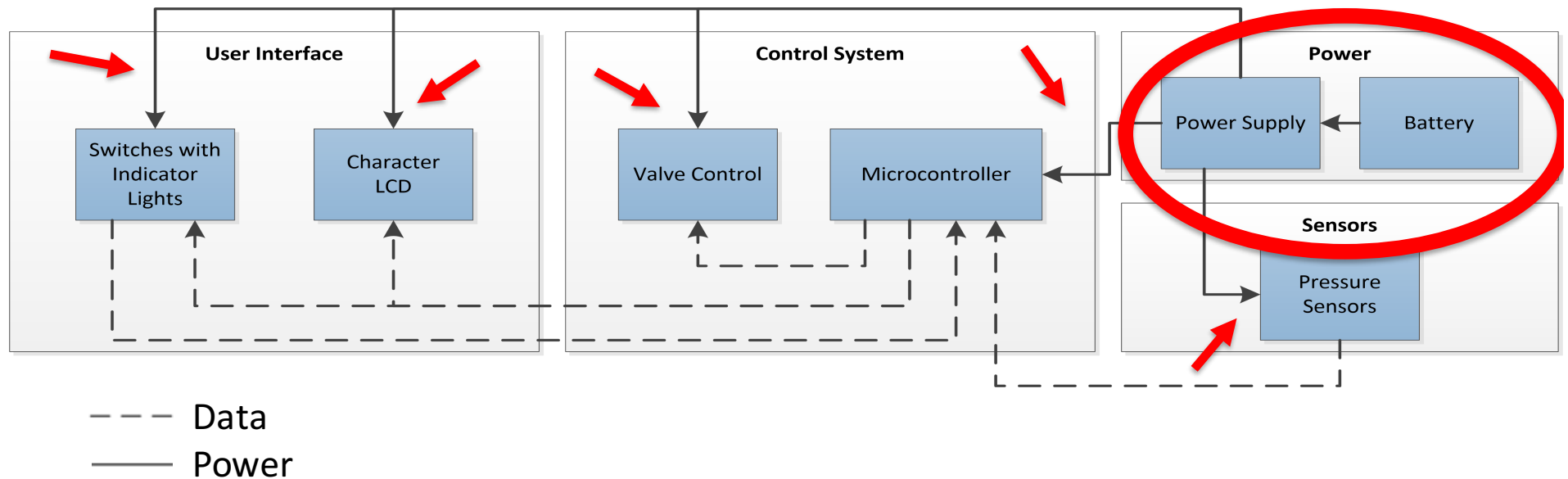
Better Modular Design

The inputs and outputs (both data and power) to each part of the design are known and clearly labeled.



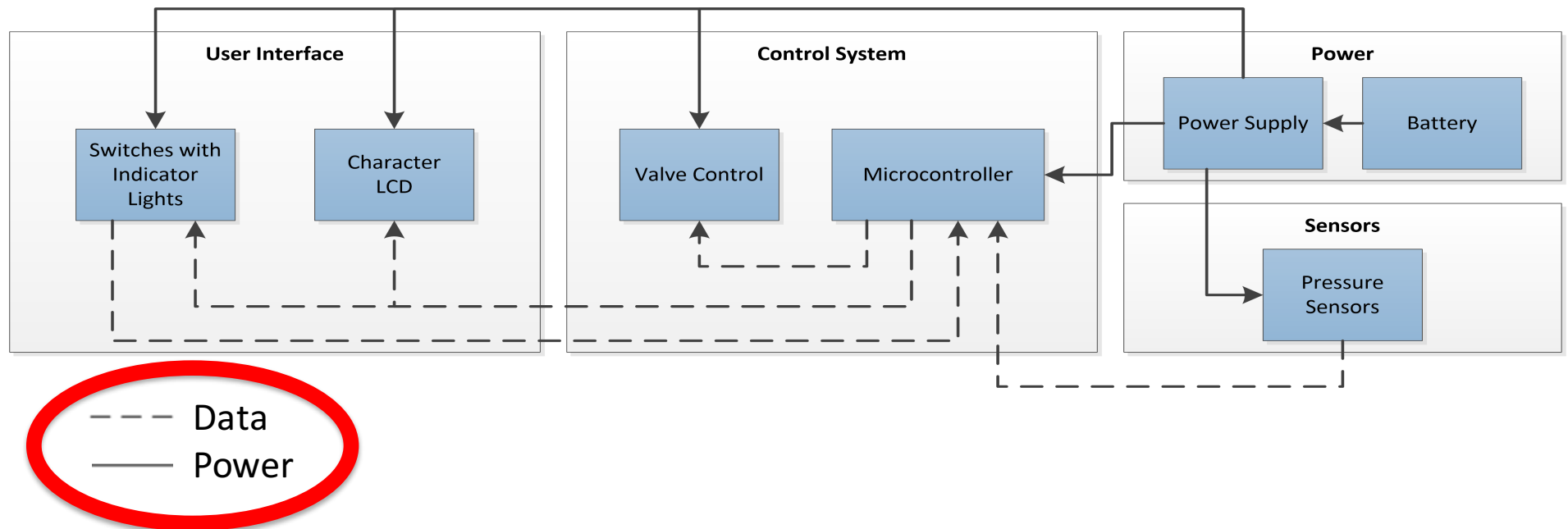
Better Modular Design

It is clear how power is provided to every block. It is clear what the power source is.



Better Modular Design

A key helps us understand what we are looking at.



Modularity for Senior Design

The purpose is to help with:

- Separation of work
- Communicating your design to others
- Requirements and Verification
- Schematics and PCB design
- Debugging

You need to be thinking about modularity
in design from day one!

Division of Labor

Separation of work:

- Modules are “packets” of work
- Responsibility for module
- Scheduling of work

Responsibility \neq workload

Communication

Communicating your design to others:

- Goes hand in hand with block diagram
- Explain design by function of various parts
- Easy to draw “blocks” on a napkin

Requirements and Verification Table

Requirements and Verification:

- Group requirements into modules (by block in block diagram or similar)
- If all requirements are fulfilled for a module, the module should be guaranteed to work.

Diagrams

Schematic:

- Structure of having modules helps with layout of schematics

PCB design:

- For some projects, several PCBs are desired. Modules are nice clean breaks in functionality.

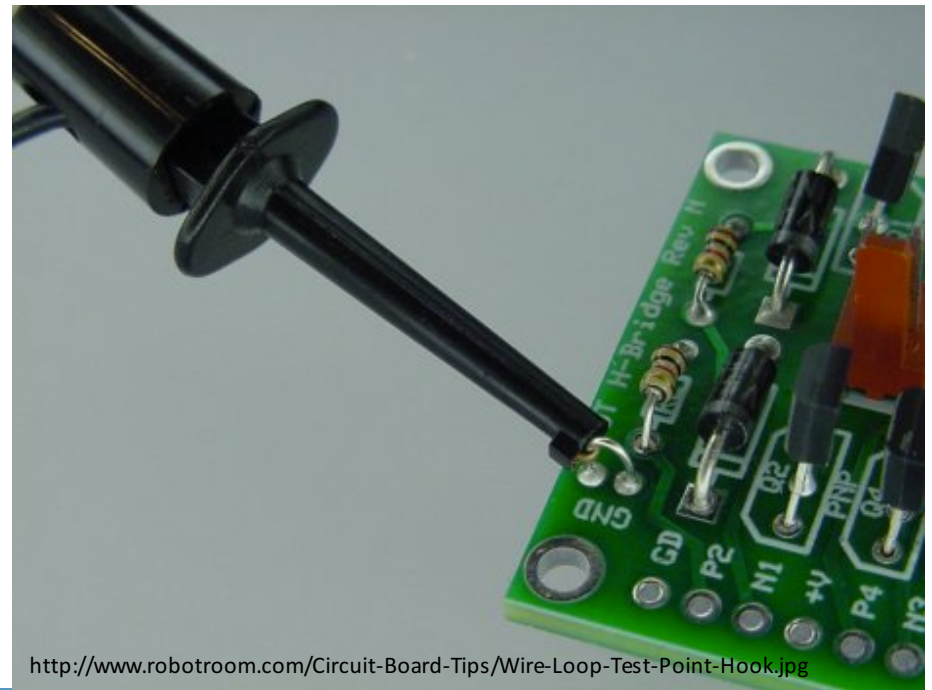
~~“PCB circuit”~~ ← BAD

The schematic shows the circuit. The PCB is the implementation.

Debugging

Debugging:

- Testing each module separately
 - Easier to troubleshoot than entire system at once
 - Requires well-defined block diagram to be useful
- Test points
 - Places on circuit where you can easily hookup test probes to verify your design



We try to get you to use modularity
in your design because we want to
help!

If you have any questions, contact
your TA.