Design Planning
What will you make and how?
Define Goals

What are you trying to make?

What does the board need to accomplish?

What inputs? What outputs?

Get as much information from others working on the project as soon as possible, hold many design reviews
Power

Any electronics will require a power source

Plugged into the wall? Running off battery?

- Highly suggest not plugging things directly into the wall; consider using pre-existing converter

How much power will it draw? Required voltage, current?

Will you need to regulate voltage on board? AC -> DC, stepping down 5V->3V?

If really high power, can use thicker copper on boards depending on ability of manufacturer
Power - Determining Requirements

This is a cyclic process as you determine what components you will need.

Big, power heavy components such as motors often set limits.

Use higher current rating than you think you need.

If you have >1 voltage level, you can regulate up or down using designated ICs.

- Easiest in small increments, i.e., 5V to 3V easier than 120V to 3V.
### Voltage Regulation - Switching vs Linear

<table>
<thead>
<tr>
<th>Linear Regulator</th>
<th>Switching regulator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step down in voltage only</td>
<td>Can step up or down</td>
</tr>
<tr>
<td>Smaller, cheaper</td>
<td>Larger, more complex</td>
</tr>
<tr>
<td>Only input/output capacitive filtering required</td>
<td>Often require more complex filtering/external components</td>
</tr>
<tr>
<td>Low power or low delta V</td>
<td>Can do high power, high delta V</td>
</tr>
<tr>
<td>Lower efficiency</td>
<td>Higher efficiency</td>
</tr>
<tr>
<td>Can be very low noise</td>
<td>Have switching noise at frequency of switching</td>
</tr>
</tbody>
</table>
# Types of batteries

Great for embedded, on the go applications

<table>
<thead>
<tr>
<th>Battery Type</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lipo</td>
<td>Very high power density, come in many voltages</td>
<td>Volatile, uneven charge/discharge curves</td>
</tr>
<tr>
<td>Coin Cell</td>
<td>Very small, cheap and easy</td>
<td>Small capacity, can’t recharge</td>
</tr>
<tr>
<td>Alkaline (AA/9V)</td>
<td>Cheap and easy, easy to find, safe</td>
<td>Often not rechargeable, low density/capacity</td>
</tr>
<tr>
<td>USB Battery Pack</td>
<td>High capacity, predictable, easy to recharge</td>
<td>Large, limited discharge rate</td>
</tr>
<tr>
<td>NiMH</td>
<td>Rechargeable, small</td>
<td>Not high density/capacity</td>
</tr>
<tr>
<td>Lithium Ion</td>
<td>High energy density, small</td>
<td>Volatile</td>
</tr>
</tbody>
</table>
Microcontroller (or other computer)

What will run your device? If dumb enough (blink an LED, etc) no need for computer

Simple tasks can use simple MCUs

Often nice to use well documented, oft-used MCUs at first for help debugging

MCU criteria

- Power level (5V, 3V, low power)
- Core processor
- # input/output ports
- Connections to peripherals (i2c, spi, serial)
- Memory size
Active Components

Will there be connected motors, LEDs, other boards, LCDs, potentiometers, etc?

Note what type of connections these have, how much power they draw, how they will be mounted to the board.

Will they require special inputs/outputs? ADC for pot, DAC for audio devices, etc.

If you have multiple non-bussed devices, ensure you have enough ports on MCU.
Connectors

What is connected to the board?

How will you connect power? Good to set up reverse polarity protection to stop yourself from frying board if plugged in incorrectly.

Anything that will be not permanently mounted, careful to use strong enough connectors

Useful to use directional (keyed) connectors so things can’t be plugged in backwards, particularly anything with power.