Book on Arduino programming, electronics, and how to think computationally

Audience = someone who knows nothing or little to nothing about electronics and knows nothing about programming or someone who is looking to develop a better intuitive understanding of the basics of electronics and programming via physical computing.

Code is provided via github

**Three sections:**

The book will be split into three distinct sections

1. Electronics
2. Programming
3. Physical Computing

The first two sections will be all theory and will cover the why of the final section. The final section is focused on applying the theory presented in the first two sections. The idea is that people can skip to the end of the book easily and just do the projects if they want to build cool shiz, but if they are interested in also learning how to perhaps create their own cool projects then they can reference back to the theory. The project section will have specific page references for learning the how/why.

**Section 1: Electronics**

* Building a conceptual model of electronics
  + Resistance/Conductance
  + Capacitance/Inductance
  + Voltage/Volts
  + Amps/Current
  + Watts/Power
* Ohms/kirchoff – framed as tools which give us insight to the nature of electronics
* Voltage Dividers
* Pullup/pulldown resistors
* Current limiting resistors
* RC filters

**Section 2: Programming**

* Github
  + How people gain access to the code in the book
  + Code is included in the github repo which is not directly discussed in the book
  + The code in github needs to be very well commented and follow strict conventions.
  + Talk about how EVERYONE should be using git or another form of version tracking with their projects
  + Cover some git basics
    - Push, pull, clone, commit, add
    - Branching
    - Point them to good git resources for going further
* Terminal basics – use mostly arduino but also perhaps a second language (processing or p5js)
  + Used to explain the github workflow and the basics of working with a terminal
  + Cover some basic terminal commands
    - Cd, mv, rm, cp, open, source
* Arduino – What is this thing?
  + The language
  + Digital Pins
  + PWM pins
  + I2C
* Programming fundamental concepts (specific to arduino)
* Sensors
  + Variable resistors
    - Photocells
    - FSR’s
    - Flex sensor
  + Potentiometers (deconstructed)
    - Softpots
    - Slides/knobs
  + Buttons/Switches
  + Light
    - RGB color
* Lights
  + LEDs
  + RGB LEDs
  + Neo Pixels
  + LCD Displays
  + lasers
* Actuators
  + Relays
  + Solenoids

**Section 3: Projects**

OUTLINE

1. How to think like a computer
   1. Slowing your thought process down
   2. How to work with 1’s and 0’s
2. Different numbering systems
   1. Base 10
   2. Base 2
   3. Base 16
3. The Arduino
   1. What is a microcontroller
   2. Microcontroller vs.