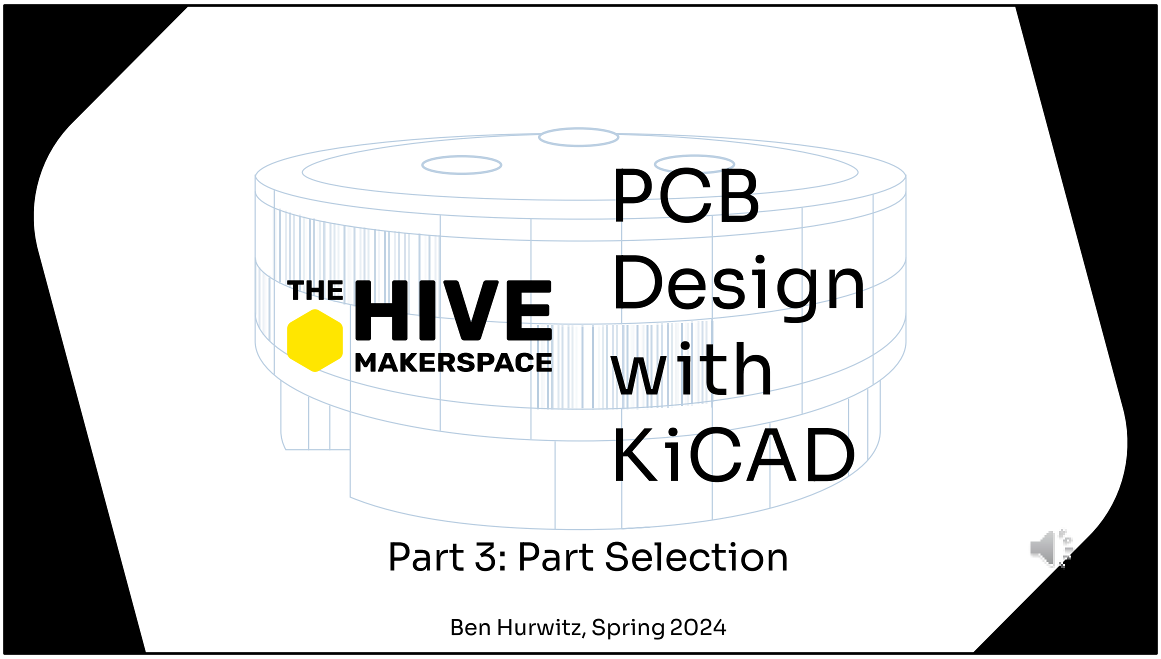


THE HIVE MAKERSPACE

PCB Design with KiCAD

Part 3: Part Selection

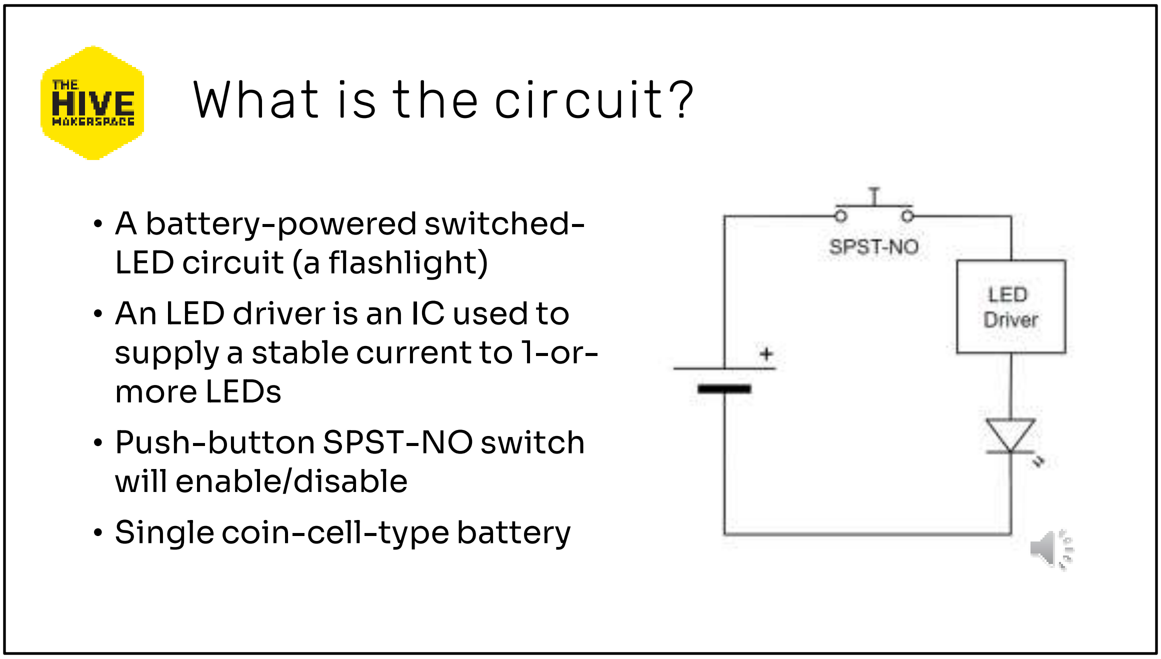
Ben Hurwitz, Spring 2024

Hi, and welcome to The Hive's series on PCB Design with KiCAD.

My name is Ben, and in this part 3, I'll be giving an overview of the circuit we're going to design a board for, and go through the part selection process.

I don't think you need any electrical engineering knowledge to understand this material since I'm not doing any actual theory; most of it is just why I picked the various parts, and how you might for your own projects.

Let's get into it.



What is the circuit?

- A battery-powered switched-LED circuit (a flashlight)
- An LED driver is an IC used to supply a stable current to 1-or-more LEDs
- Push-button SPST-NO switch will enable/disable
- Single coin-cell-type battery

SPST-NO
LED Driver
+

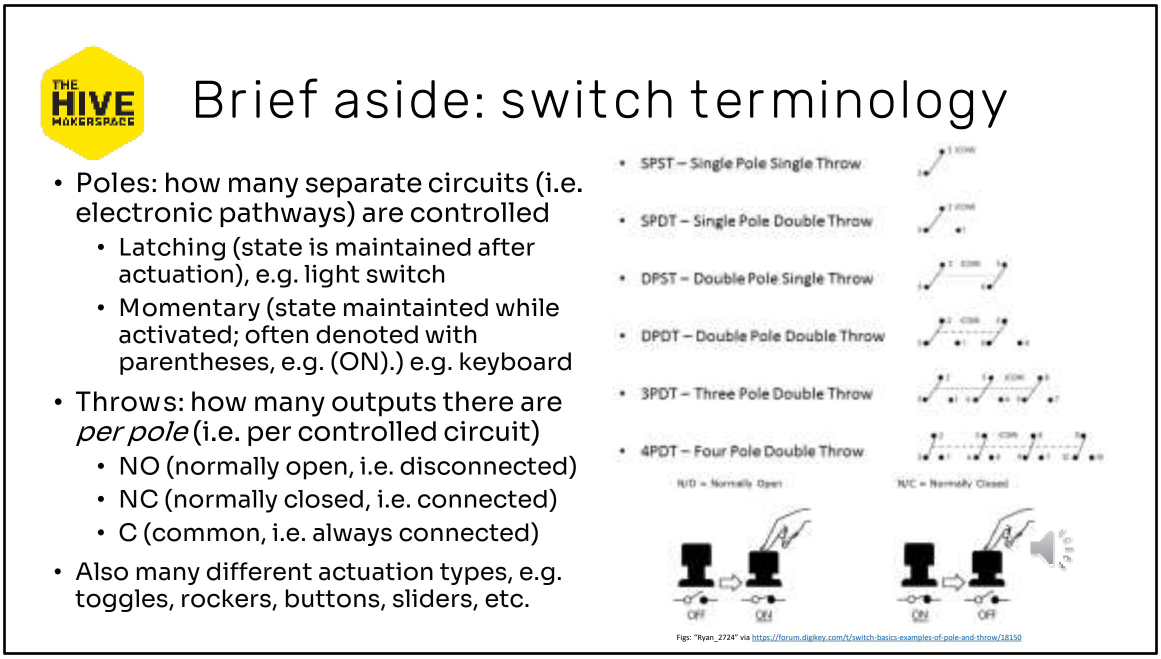
So what are we actually designing a board for?

It’s basically a flashlight, or more technically a battery-powered switched LED circuit.

An LED drive is an IC that provides a fixed and stable current and voltage at the output for driving a number of LEDs. It’s more stable than a battery and a resistor, and it can boost the input voltage up to drive many LEDs both in series and parallel.

A simple push-button tactile switch in a SPST-NO configuration will turn on and off the circuit.

The power will be provided by a single coin-cell battery.


SPST – Single Pole Single Throw
SPDT – Single Pole Double Throw
DPST – Double Pole Single Throw
DPDT – Double Pole Double Throw
3PDT – Three Pole Double Throw
4PDT – Four Pole Double Throw
N/O = Normally Open
N/C = Normally Closed
OFF
ON
ON
OFF

Brief aside: switch terminology

- Poles: how many separate circuits (i.e. electronic pathways) are controlled
  - Latching (state is maintained after actuation), e.g. light switch
  - Momentary (state maintainted while activated; often denoted with parentheses, e.g. (ON).) e.g. keyboard
- Throws: how many outputs there are *per pole* (i.e. per controlled circuit)
  - NO (normally open, i.e. disconnected)
  - NC (normally closed, i.e. connected)
  - C (common, i.e. always connected)
- Also many different actuation types, e.g. toggles, rockers, buttons, sliders, etc.

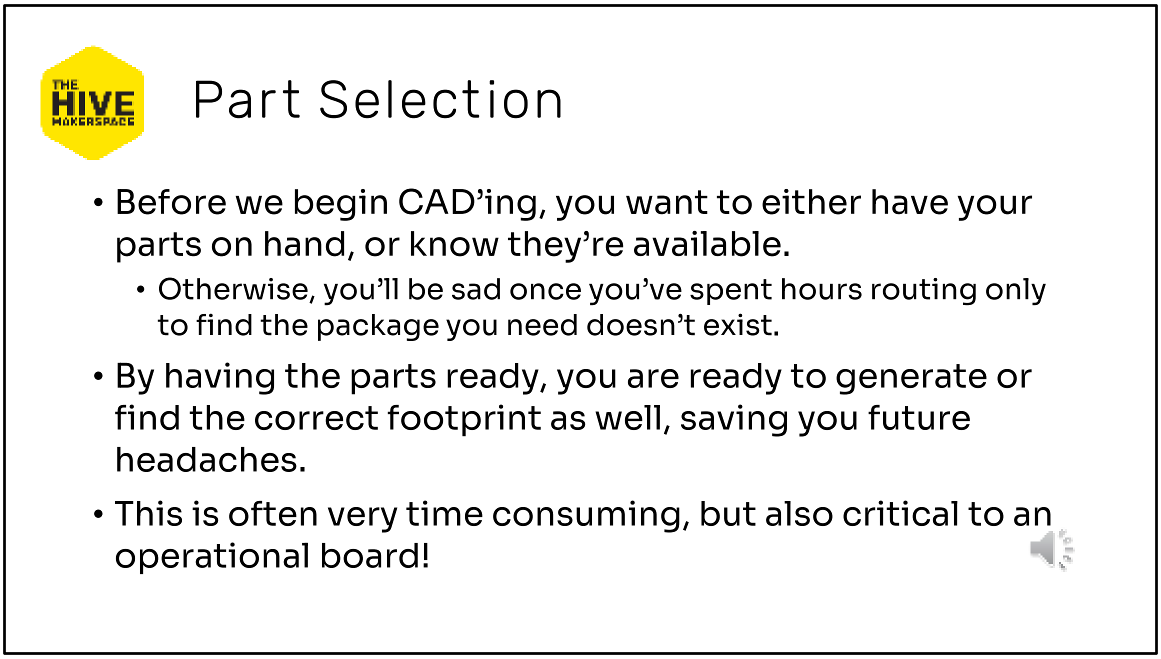
Figs: "Ryan_2724" via https://forum.digikey.com/t/switch-basics-examples-of-pole-and-throw/18150

Okay, so for those who don't know (or, like me, always forget) about switch jargon, here's a rundown.

The number of poles describes how many circuits the switch controls. This is not how many outputs, but literally how many circuits, how many electronic pathway selections. You should be able to see the difference between the first two on the right, which are both single pole, and the second two, which are double pole. Each pole can either be latching, where the state is maintained like a light switch, or momentary, where the state only changes for the duration of the actuation, like a keyboard key.

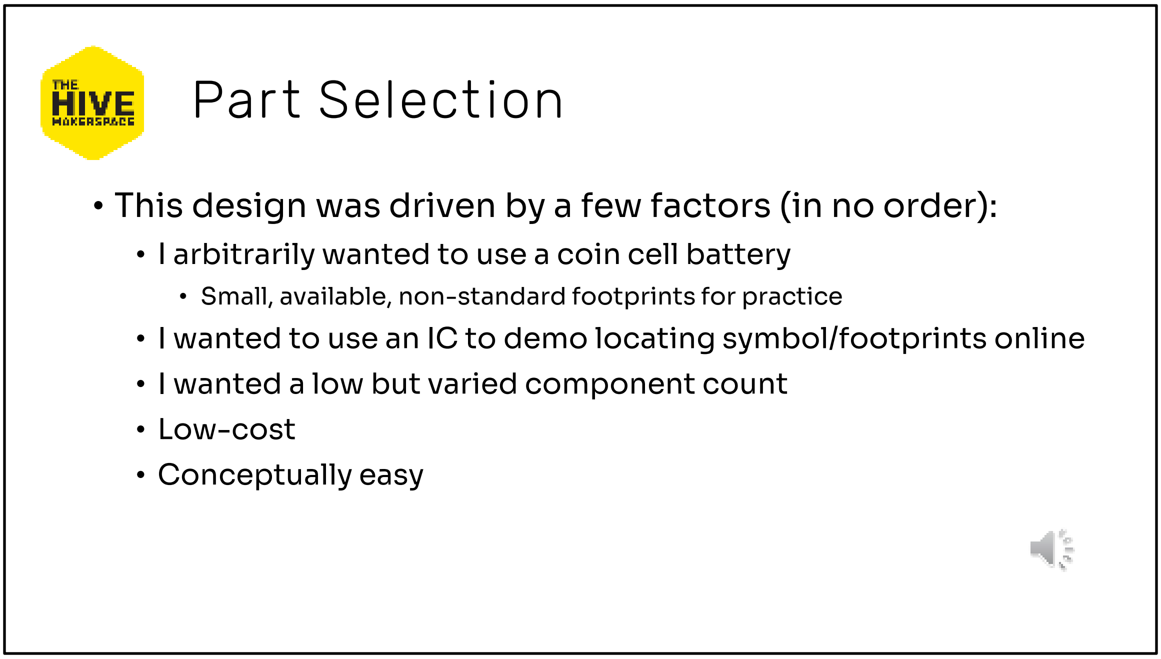
The second important term is the number of throws, which described the number of output per pole. So the first switch, the single pole single throw, controls one circuit with a single output. The second switch, single pole double throw, has one circuit with two outputs. The third is double pole single throw, so two circuits each with one output. And so on. Throws can be normally open, meaning disconnected, like the button on the lower right, or normally closed, meaning connected, like the button on the lower left. One of the terminals will be common as well, meaning always connected, and is mostly relevant for double throw (two output) switches.

Switches are typically named by the number of poles then the number of throws, single pole single throw, with the shorthand SPST. Single throw switches will often have a notation for whether they’re normally open or normally closed.



Part Selection

- Before we begin CAD’ing, you want to either have your parts on hand, or know they’re available.
  - Otherwise, you’ll be sad once you’ve spent hours routing only to find the package you need doesn’t exist.
- By having the parts ready, you are ready to generate or find the correct footprint as well, saving you future headaches.
- This is often very time consuming, but also critical to an operational board!

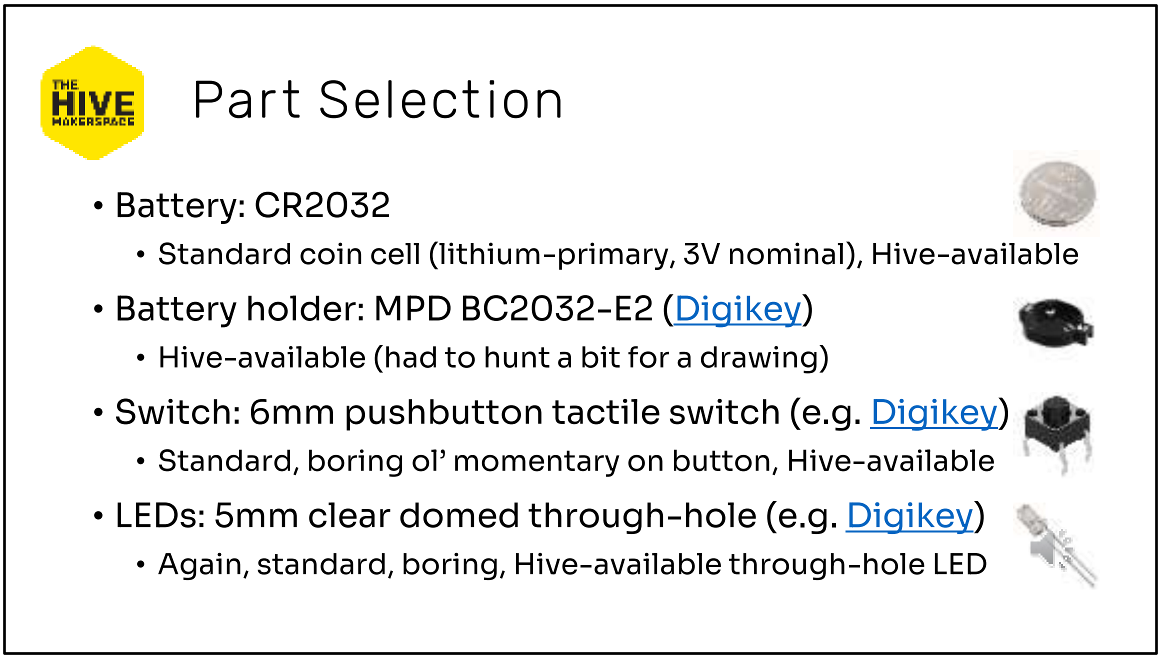


Part Selection

- This design was driven by a few factors (in no order):
  - I arbitrarily wanted to use a coin cell battery
    - Small, available, non-standard footprints for practice
  - I wanted to use an IC to demo locating symbol/footprints online
  - I wanted a low but varied component count
  - Low-cost
  - Conceptually easy

Part Selection

- How did I locate these parts?
  - I tried to find parts that are readily available in The Hive
    - These included Rs, Cs, Ls, LEDs, the switch, the battery, and its holder
  - Other components were sourced from Digikey
    - I'm familiar with them and navigating their website, but I'm sure I could have found most of this from Amazon, Mouser, or other supplier



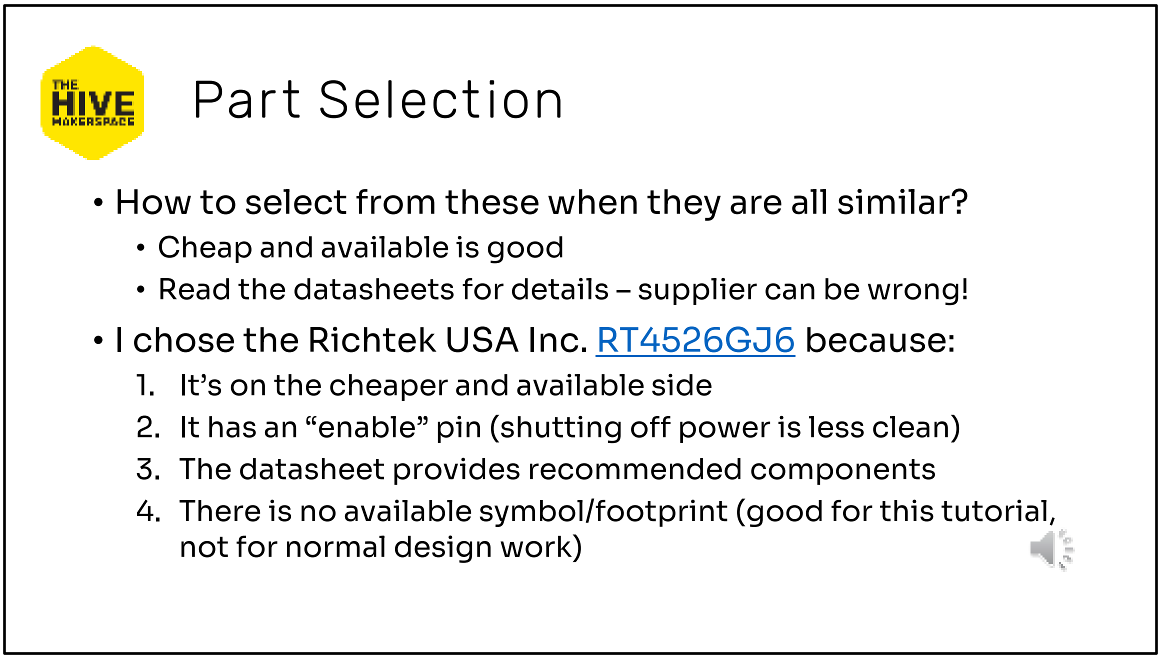
Part Selection

- Battery: CR2032
  - Standard coin cell (lithium-primary, 3V nominal), Hive-available
- Battery holder: MPD BC2032-E2 (Digikey)
  - Hive-available (had to hunt a bit for a drawing)
- Switch: 6mm pushbutton tactile switch (e.g. Digikey)
  - Standard, boring ol' momentary on button, Hive-available
- LEDs: 5mm clear domed through-hole (e.g. Digikey)
  - Again, standard, boring, Hive-available through-hole LED

I went with a coin cell because it's small

Part Selection

- The LED driver was a bit more involved, as ICs usually are
- Digikey filtered for:
  - “in stock” and “active” status (always a good filter set)
  - a minimum supply voltage of ≤3V (so it works with the battery)
  - a surface-mounted package (arbitrary) that is hand-solderable (so no super-tiny or BGA-style packages)
  - Boost topoliges only (need to increase the voltage for the LED)
- This resulted in 26 options

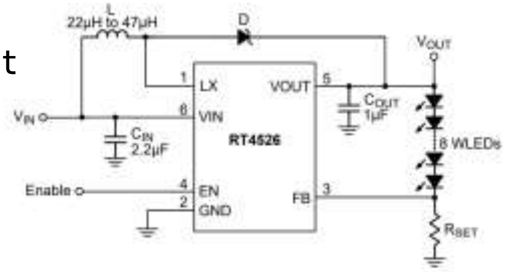


Part Selection

- How to select from these when they are all similar?
  - Cheap and available is good
  - Read the datasheets for details – supplier can be wrong!
- I chose the Richtek USA Inc. RT4526GJ6 because:
  1. It's on the cheaper and available side
  2. It has an "enable" pin (shutting off power is less clean)
  3. The datasheet provides recommended components
  4. There is no available symbol/footprint (good for this tutorial, not for normal design work)

Part Selection

- The “typical application” circuit
- I_LED = 0.3/Rset -> for 10mA, Rset = 30 ohm
- For simplicity, Rset, Cin, Cout, and L will be Hive-available
- Diode is unavailable on Digikey; similar selected



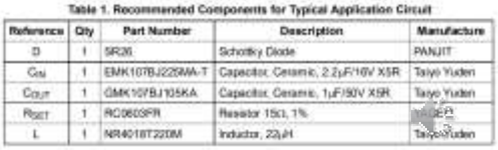
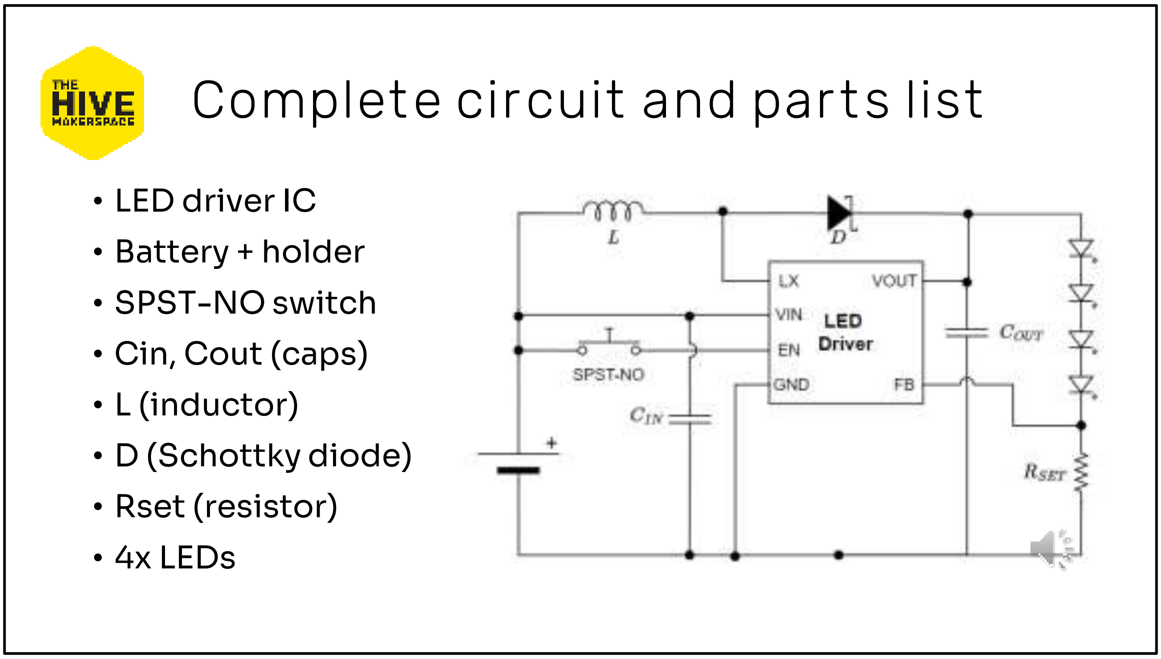


Table 1. Recommended Components for Typical Application Circuit

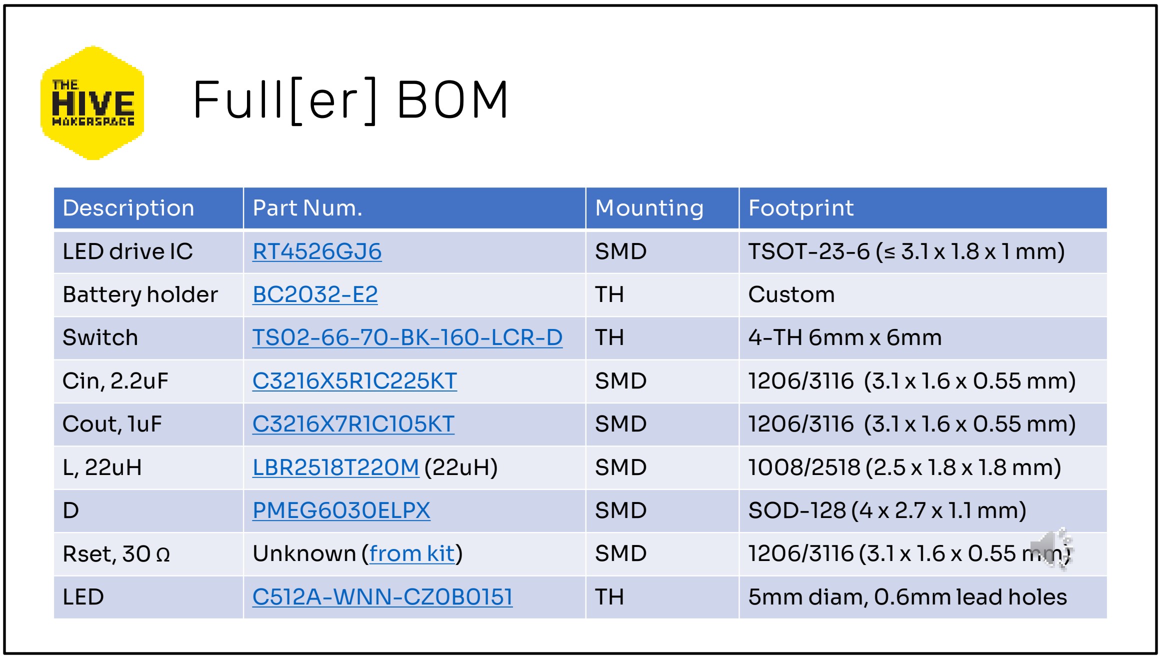
Reference	Qty	Part Number	Description	Manufacture
D	1	SR26	Schottky Diode	PANJIT
C_{IN}	1	EMK107BJ225MA-T	Capacitor, Ceramic, 2.2μF/16V X5R	Taiyo Yuden
C_{OUT}	1	GMK107BJ105KA	Capacitor, Ceramic, 1μF/50V X5R	Taiyo Yuden
R_{SET}	1	RC0603FR	Resistor 15Ω, 1%	YAGEO
L	1	NR4018T220M	Inductor, 22μH	Taiyo Yuden



Complete circuit and parts list

- LED driver IC
- Battery + holder
- SPST-NO switch
- Cin, Cout (caps)
- L (inductor)
- D (Schottky diode)
- Rset (resistor)
- 4x LEDs

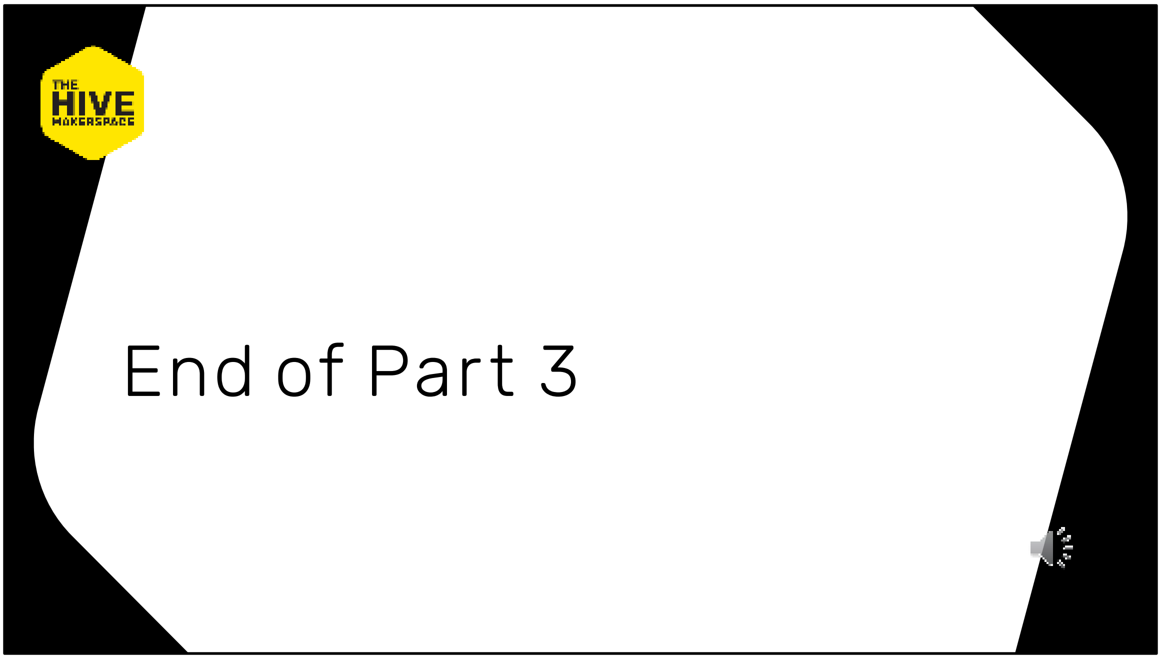
Here’s the full circuit and parts lists. Don’t worry too much about memorizing this or anything. I’ll bring it back frequently as needed during the upcoming videos (though feel free to print it or whatever).



Full[er] BOM

Description	Part Num.	Mounting	Footprint
LED drive IC	RT4526GJ6	SMD	TSOT-23-6 (≤ 3.1 x 1.8 x 1 mm)
Battery holder	BC2032-E2	TH	Custom
Switch	TS02-66-70-BK-160-LCR-D	TH	4-TH 6mm x 6mm
Cin, 2.2uF	C3216X5R1C225KT	SMD	1206/3116 (3.1 x 1.6 x 0.55 mm)
Cout, 1uF	C3216X7R1C105KT	SMD	1206/3116 (3.1 x 1.6 x 0.55 mm)
L, 22uH	LBR2518T220M (22uH)	SMD	1008/2518 (2.5 x 1.8 x 1.8 mm)
D	PMEG6030ELPX	SMD	SOD-128 (4 x 2.7 x 1.1 mm)
Rset, 30 Ω	Unknown (from kit)	SMD	1206/3116 (3.1 x 1.6 x 0.55 mm)
LED	C512A-WNN-CZ0B0151	TH	5mm diam, 0.6mm lead holes

And this is a large bill of materials without any of the "bill" portion. We'll see this again later as well.



End of Part 3

And with that, we've reached the end of Part 3, in which I introduced the circuit and went through the process of part selection. Hopefully this provided some insight on doing this process on your own. A PDF of this video is available as well, linked in the description and hosted on The Hive's Wiki.

In the next video, part 4A, we'll finally get into actual design with an introduction to KiCAD's schematic capture view and placing basic symbols.

See you there!