

# Intro to KiCad

KiCon 2019 (based on KiCad 5.1.0)

Worksheet by Shawn Hymel


Name: \_\_\_\_\_

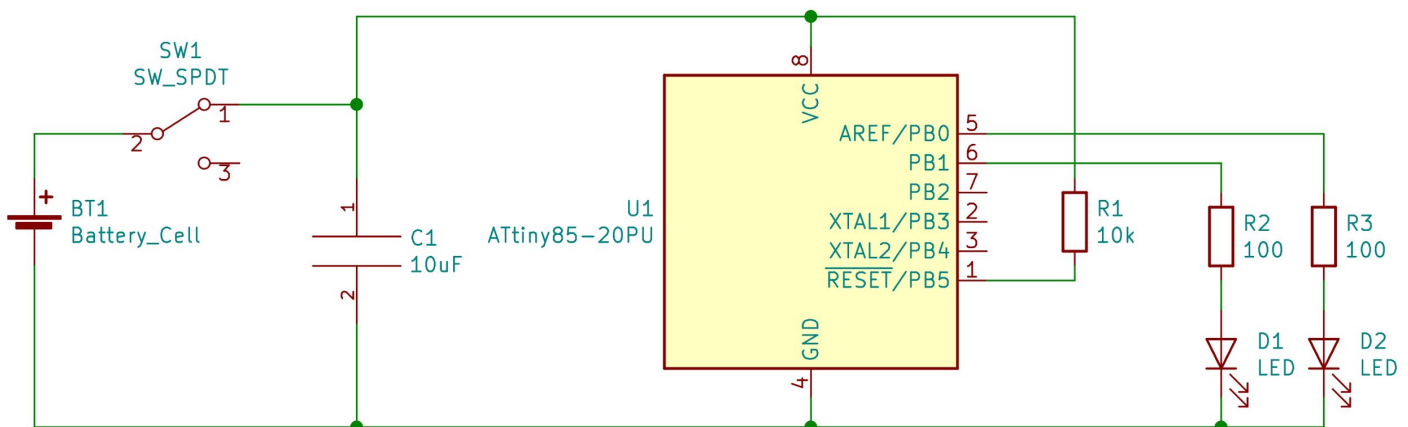
Date: \_\_\_\_\_

## Installation


- Navigate to **[kicad-pcb.org/download/](http://kicad-pcb.org/download/)** and download the installer for your OS
- Run the installer, accepting all defaults
- Wanna see video tutorials of KiCad? Go here: **[bit.ly/LearnKiCad](http://bit.ly/LearnKiCad)**
- Project files can be found here: **[github.com/ShawnHymel/kicon19-blinky](https://github.com/ShawnHymel/kicon19-blinky)**

## Schematic Capture

- Start KiCad; you should be looking at the Project Manager window
- File > New > Project, create a new project named **kicon19-blinky.pro**
- Click **Schematic Layout Editor** 
- Place > Symbol, click anywhere in work area
- Search for **ATtiny85-20PU**, click **OK**
- Place anywhere within schematic sheet
- Repeat for:
  - 1x **Battery\_Cell**
  - 1x **SW\_SPDT** (for “switch”)
  - 1x **CAP** (for “capacitor”)
  - 3x **R** (for “resistor”)
  - 2x **LED**
- Update component values (Right-click on symbol > Properties > Edit Value...)
  - Change capacitor value from **CAP** to **10uF**
  - Change 2x resistor values from **R** to **100**
  - Change 1x resistor value from **R** to **10k**
- Move parts around to match the schematic in the image below
  - Right-click on symbol > Move to move symbols
  - Right-click on symbol > Orientation > Rotate Clockwise to rotate
- Connect the components as shown in the schematic
  - Place > Wire
  - Click on symbol end to start wire, click on another symbol end to stop drawing wire
- Add reference designators to symbols
  - Tools > Annotate Schematic
  - Keep defaults, click **Annotate** then click **Close**
  - The symbols should have reference designators (e.g. R1, R2)



## Assign Footprints to Symbols

- This is where we tell KiCad to associate a PCB part footprint with each schematic symbol
- Click Tools > Assign Footprints...
- Make sure only the *Pin Count* and *Library* filters are selected → 
- Assign **BatteryHolder\_Keystone\_103\_1x20mm** footprint to *Battery\_Cell* symbol
  - In *Footprint Libraries* pane, select the *Battery* library
  - In *Footprint Assignments* pane, select *Battery\_Cell* symbol (BT1)
  - In *Filtered Footprints* pane, double-click *BatteryHolder\_Keystone\_103\_1x20mm*
- Assign **C\_Disc\_D5.0mm\_W2.5mm\_P2.50mm** to *10uF* capacitor symbol
  - In *Footprint Libraries* pane, select the *Capacitor\_THT* library
  - In *Footprint Assignments* pane, select *10uF* symbol (C1)
  - In *Filtered Footprints* pane, double-click *C\_Disc\_D5.0mm\_W2.5mm\_P2.50mm*
- Repeat this process for the rest of the symbols:
  - From the *LED\_THT* library, assign **LED\_D5.00mm** to both *LED* symbols (D1, D2)
  - From the *Resistor\_THT* library, assign **R\_Axial\_DIN0207\_L6.3mm\_D2.5mm\_P10.16mm\_Horizontal** to the 10k and 100 Ohm resistors (R1, R2, R3)
  - From the *Button\_Switch\_THT* library, assign **SW\_Slide\_1P2T\_CK\_OS102011MS2Q** to the *SW\_SPDT* switch (SW1)
  - From the *Package\_DIP* library, assign **DIP-8\_W7.62mm** to the *ATtiny85-20PU* (U1)
- Make sure your footprint assignments look like the ones below, and click **OK**



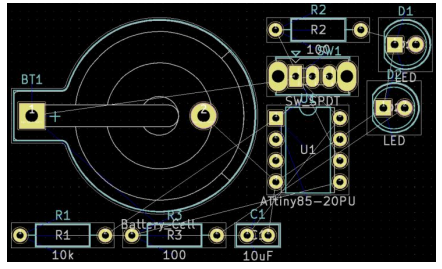
### Symbol: Footprint Assignments

1	BT1 -	Battery_Cell : Battery:BatteryHolder_Keystone_103_1x20mm
2	C1 -	10uF : Capacitor_THT:C_Disc_D5.0mm_W2.5mm_P2.50mm
3	D1 -	LED : LED_THT:LED_D5.0mm
4	D2 -	LED : LED_THT:LED_D5.0mm
5	R1 -	10k : Resistor_THT:R_Axial_DIN0207_L6.3mm_D2.5mm_P10.16mm_Horizontal
6	R2 -	100 : Resistor_THT:R_Axial_DIN0207_L6.3mm_D2.5mm_P10.16mm_Horizontal
7	R3 -	100 : Resistor_THT:R_Axial_DIN0207_L6.3mm_D2.5mm_P10.16mm_Horizontal
8	SW1 -	SW_SPDT : Button_Switch_THT:SW_Slide_1P2T_CK_OS102011MS2Q
9	U1 -	Attiny85-20PU : Package_DIP:DIP-8_W7.62mm

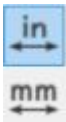
## Generate Netlist

- A netlist tells the PCB layout software where everything is connected, according to the schematic
- Click Tools > Generate Netlist File...
- Click **Generate Netlist**
- Save .net file in your project directory
- Close Schematic Layout Editor

## Load Netlist into PCB Layout Editor

- In the Project Manager, click **PCB Layout Editor** → 
- Click Tools > Load Netlist...
- Click the *folder* button in the top-right of the window → 
- Select your netlist file (.net), and click *Open*
- Click **Update PCB**
- All your components should be on your cursor
  - Zoom in/out with your mouse wheel
  - Click to place them in the center of the work area → 

## Configure Design Rules

- Change units to *inches* by clicking on this button on the left bar → 
- Click File > Board Setup...
- In *Design Rules*, enter the following minimums →
- In *Net Classes*, enter the following:
  - Clearance: 20 mils
  - Track Width: 20 mils
  - Via Size: 32 mils
  - Via Drill: 16 mils
  - $\mu$ Via Size: 32 mils
  - $\mu$ Via Drill: 16 mils
  - dPair Width: 20 mils
  - dPair Gap: 20 mils

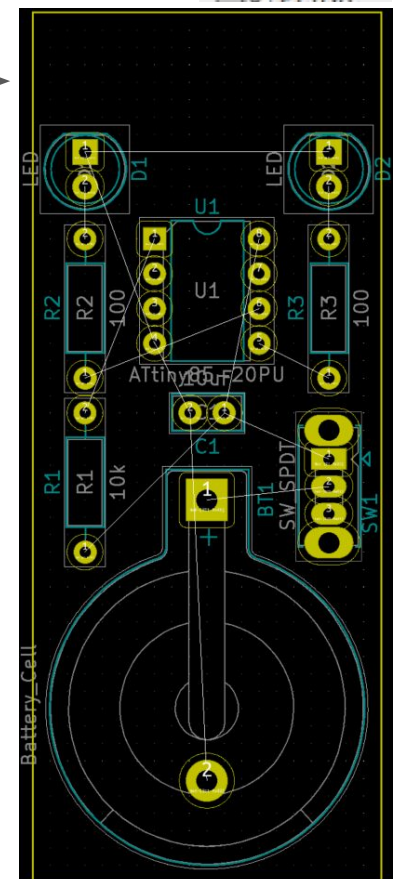
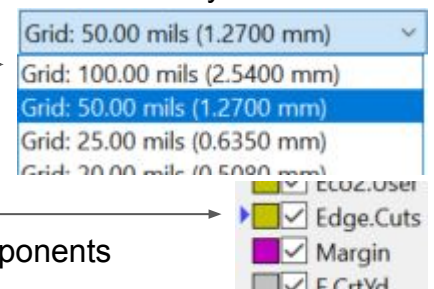
Minimum track width:	20	mils
Minimum via diameter:	32	mils
Minimum via drill:	16	mils
Minimum $\mu$ Via diameter:	32	mils
Minimum $\mu$ Via drill:	16	mils
Minimum hole to hole:	20	mils

Name	Clearance	Track Width	Via Size	Via Drill	$\mu$ Via Size	$\mu$ Via Drill	dPair Width	dPair Gap
Default	20 mils	20 mils	32 mils	16 mils	32 mils	16 mils	20 mils	20 mils

- Click **OK**. This sets our PCB trace and drill parameters, and it should match our manufacturing process (you can get these rules from your PCB manufacturer's website)

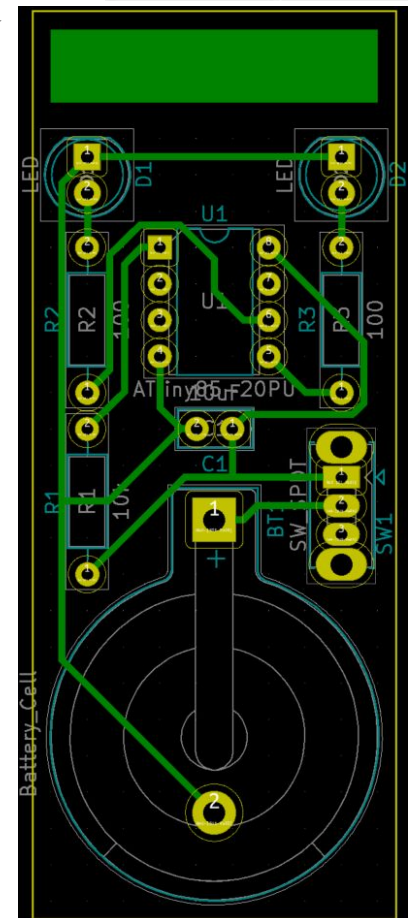
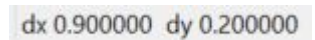
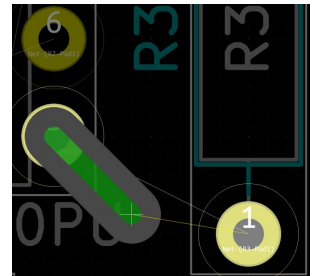
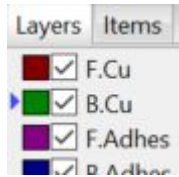
## Organize Components

- NOTE: This is a *single-sided PCB*, so our components will be on the *Front* layer and our traces will be on the *Back* layer
- At the top, change the grid to *50.00 mils (1.2700 mm)* →
- Organize the components
  - Right-click on the components, select *Move* or *Rotate*
  - Try to avoid *airwire* crossings as much as possible
- On the right bar, click the *Edge.Cuts* layer to select it →
- Click Place > Line, and draw a board outline around your components
  - Leave 0.3 inches at top to allow for pin back
- Check your board
  - It should look like this →
  - Outline should measure 1.0 x 2.5 inches
  - Check outline dimensions with Inspect > Measure



## Route Traces

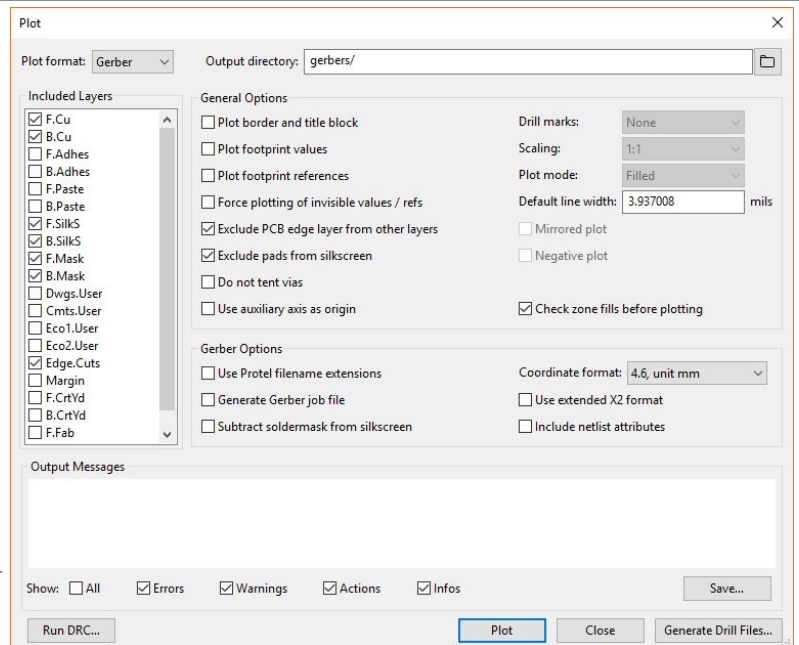
- At the top, change the grid to *10.00 mils (0.2540 mm)*
- Select *B.Cu* layer
- Route traces
  - Click Route > Single Track
  - Click on component through-hole to start
  - Click to place bend in track
  - Click on end component through-hole to end
  - You can also click on parts of a track to connect traces together
  - Where each trace goes will be highlighted by its airwire
  - Let the airwires guide you!
  - Do not cross traces from different nets.
  - Stay on the B.Cu layer
  - Keep connecting component leads together until the airwires are gone
- Add a copper “filled zone” for the pin back
  - Click Place > Zone
  - Click 0.05 in. down and 0.05 in. to the right of the top-left corner of the PCB outline
  - A pop-up will appear, keep all defaults, and click **OK**
  - Press spacebar to set a temporary origin point
  - Draw a 0.9 x 0.2 in. rectangle.
  - Use the dx/dy markers in the bottom-right to help
- Your final board should look like this:
- Check your design
  - Click Inspect > Design Rules Checker
  - Click **Run DRC**
  - You should see 4 errors pointing at the switch footprint, as the holes are too close together (but still OK)
  - You should also see 1 error showing the battery footprint has an incorrect courtyard (this is also OK)
  - Find and fix any other errors you might see
  - Click **List Unconnected**
  - You should see no unconnected items
  - Click **Delete All Markers**
  - Click **Close**



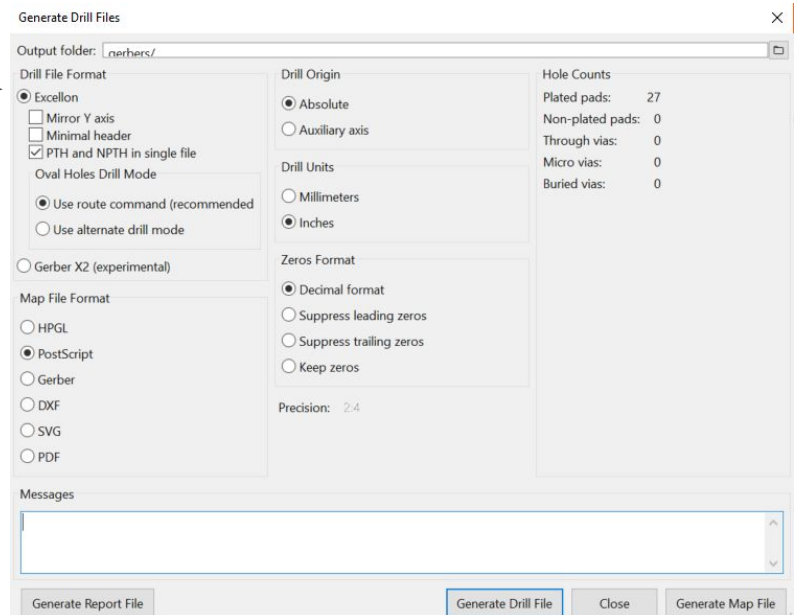


## Generate Gerbers

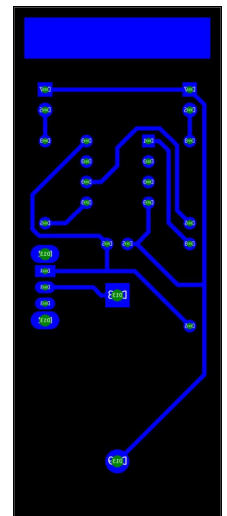
- Create Gerber files (ready for manufacturing!)
  - Click File > Plot...
  - Change *Output directory* to **gerbers/**
  - In *Included Layers*, make sure just the following are selected: **F.Cu, B.Cu, F.SilkS, B.SilkS, F.Mask, B.Mask, Edge.Cuts**
  - In *General Options*, deselect **Plot footprint values** and **Plot footprint references**
  - Make sure your options are as follows →
  - Click **Plot**



- Click **Generate Drill Files...**
- Leave everything as default →
- Click **Generate Drill File**
- Close out of both windows




- Save and exit out of the *PCB Layout Editor* program
- Inspect your drill file and Gerber files
  - Open the **Gerber Viewer** from the Project Manager window
  - Click File > Open Excellon Drill File(s)...
  - Open the .drl file in your *gerbers* directory
  - Click File > Open Gerber File(s)...
  - Open the B\_Cu.gbr file
  - Click File > Open Gerber File(s)...
  - Open the Edge\_Cuts.gbr file
  - Your drill holes, traces, and board outline should all line up →
  - Feel free to repeat for the other layers (if you plan to have your board professionally manufactured)



## Make Your Boards

- Send your boards off for fabrication!
  - One good place is [oshpark.com](https://oshpark.com)
  - Note that this is a *single-sided board*, which might save you on costs
- Another option: mill boards yourself (if you have a CNC mill)
  - The Bantam Tools Desktop Milling Machine is a good option

## Assemble Board

- Place components on the front side (the side without copper)
  - Refer to this assembly guide 
  - Note that U1 should be the DIP socket
  - Solder component leads on back side (side with copper)
  - Hold the safety pin to the copper rectangle on the back side a soldering iron and feed solder into the pin's holes
- Program the ATtiny85 with the fade-blinky Arduino program
  - Code found in GitHub project repository
  - This can be done with another Arduino
  - Alternatively, this board can help: [sparkfun.com/products/11801](https://sparkfun.com/products/11801)
- Test it!
  - Insert the ATtiny85 into the socket
  - Insert the battery
  - Flip the switch
  - Things should blink! 