Lab 02: How big is a picofarad?

You'll measure capacitance from an AC current, separate a plate capacitor's area term from its edge term, and price one picofarad in each of the capacitor types SKY130 offers. Then you'll size two capacitors of your own, measure the inductance, Q and self-resonance of the PDK's on-chip spirals, and build a 2.4 GHz resonator from one.

Package: `labs/lab-02-how-big-is-a-picofarad/`

Before you start

- Finish Lab 01: `make mine check` in `labs/lab-01-a-resistor-you-designed` prints `PASS`.
- Read Capacitors on a chip, What a picofarad costs, Parasitic capacitance and Inductors on a chip.
- Open a terminal in the workbench and go to the lab folder:

```
cd /foss/designs/modules/ad102/labs/lab-02-how-big-is-a-picofarad
```

  You don't need to set up any environment for this lab, because every deck names its model files by absolute path.

Parts 1 to 8 are required. Everything under Going further, at the end, is optional.

Time: about 2.5 hours, not counting the optional parts. A full `make` runs four ngspice decks (`pf`, `mine`, `henry` and `tank`) and then a checker script, `src/check_cap.py`, which fits and prints the results. Three of the decks print nothing for anywhere from about 10 seconds to a minute each while the SKY130 model cards load, so don't take the silence for a hang; the Makefile budgets about three and a half minutes for the whole `make` on the pinned image (Warnings and delays).

The theory this lab uses is all on Capacitors on a chip, and here it is in the form the lab needs. Two plates of area A separated by a dielectric of thickness t give $C = \varepsilon A/t$. On a chip, ε and t are both fixed by the process, so capacitor design comes down to one constant, farads per square micrometre, and one decision, how much area to spend. You can't make the dielectric thinner, so the only way you can get more capacitance is to make the plate bigger. The process engineers could raise the density only by making t smaller, which means a lower breakdown voltage and more leakage, so density isn't free for them either. Real plates also have edges, where the field bulges out and adds capacitance that scales with the perimeter. For a square plate of side s:

$$C = \alpha s^2 + \beta s$$

The bigger the plate, the less the edge term matters, which is the same pattern you saw with Lab 01's end resistance.

Every capacitance in this lab's package is measured the same way: the deck hangs the device on a 1 V AC source at 1 MHz and reads the current the source supplies. For a capacitor $I = 2\pi f C V$, so $C = |I|/(2\pi f)$, which is a single division with no fitting.

Part 1: The area ladder

Does the density on the model card predict what a square MIM plate measures?

`spice/picofarad.spice` builds five square MIM capacitors with sides from 1 µm to 20 µm.

1. Read the density off the MIM model's parameter file:

```
grep -hE "^\+ c[ap]mimc" /foss/pdks/sky130A/libs.tech/ngspice/r+c/
res_typical__cap_typical__lin.spice
```

```
+ camimc=  2.00e-15  ; Units: farad/micrometer^2
+ cpmimc = 0.19e-15 ; Units: farad/micrometer
```

2. On paper, predict all five capacitances using only the first number, `camimc`.
3. Run the lab:

   ```
   make
   ```

   It ends in `FAIL`, which you'll fix in Part 6. Leave the output in your terminal, because the other parts read from it.
4. Find block A and compare it with your predictions.

You should see:

```
--- A. MIM area ladder, square plates (farads) ---
c_s1 = 2.642250e-15
c_s2 = 9.302250e-15
c_s5 = 5.328225e-14
c_s10 = 2.065822e-13
c_s20 = 8.131822e-13
```

For the 20 µm plate you predicted 800.0 fF, and it measures 813.18 fF, 1.6 % high. For the 1 µm plate you predicted 2.00 fF, and it measures 2.64 fF, 32 % high. Both come out high, but the relative error on the small plate is twenty times larger.

The two capacitors an analog designer most wants to match are usually a big one and a small one, such as a sampling capacitor and a reference, or the two arms of a divider. What happens to their ratio if you size both from the area term alone?

Answer: A1, in Answers at the end.

Part 2: Fit the area and edge terms

How many terms does it take to describe a MIM plate exactly?

$C = \alpha s^2 + \beta s$ has two unknowns, and you have five measurements.

1. Take the two extremes, $s = 1$ and $s = 20$, in femtofarads, and solve for α and β.
2. Compare with what the checker printed:

```
two plates, two terms:  C[fF] = area*s^2 + edge*s
  area term   2.000887 fF per um^2       reference 2.000887
  edge term   0.641363 fF per um of side  reference 0.641363

  against the three plates it was not fitted to
    side   2.0 um   fit     9.2863 fF   measured     9.3022 fF   (-0.1717 %)
    side   5.0 um   fit    53.2290 fF   measured    53.2822 fF   (-0.0999 %)
    side  10.0 um   fit   206.5024 fF   measured   206.5822 fF   (-0.0386 %)
```

3. Look at the signs of those three residuals. Before you read the checker's one-line comment under them, decide what they tell you.

   Answer: A2, in Answers at the end.

4. Find the checker's three-term fit:

```
three plates, three terms:  C[fF] = area*s^2 + edge*s + const
  area    2.000000    edge   0.659991    const  -0.017742
  against all five plates
    side   1.0 um   fit     2.6422 fF   measured     2.6422 fF   (+0.0000 %)
    side   2.0 um   fit     9.3022 fF   measured     9.3022 fF   (-0.0001 %)
    side   5.0 um   fit    53.2822 fF   measured    53.2822 fF   (-0.0001 %)
    side  10.0 um   fit   206.5822 fF   measured   206.5822 fF   (+0.0000 %)
    side  20.0 um   fit   813.1822 fF   measured   813.1822 fF   (+0.0000 %)
```

You should see: three coefficients fitted to three plates reproduce all five plates to one part in a million, and the area term comes out as 2.000000, which is the model card's `camimc = 2.00e-15` to seven figures.

This fit describes the device better than the headline density does, and you can invert it to answer the question you'll actually be asking when you design: what side length gives me 250 fF?

The card says the perimeter term is `cpmimc = 0.19e-15` farads per micrometre of edge, and a square of side s has $4s$ of edge, so you would expect $\beta = 0.76$. You measured 0.659991. Where did the difference go?

Answer: A3, in Answers at the end.

Part 3: Size one picofarad

How far off is a picofarad sized from the area term alone?

1. Work out the naive size: $A = 1000\,\text{fF}/2.00 = 500\,\mu\text{m}^2$, a square of side $\sqrt{500} = 22.3607$ µm. The deck contains it as `c_naive`.
2. Solve $2.000000\,s^2 + 0.659991\,s - 0.017742 = 1000$ for the corrected side. The deck contains it as `c_fixed`.
3. Predict how far from 1 pF the naive plate lands. Then find block B.

You should see:

```
--- B. one picofarad, naive vs fixed (farads) ---
c_naive = 1.014742e-12
c_fixed = 1.000001e-12
```

The naive plate gives 1.0147 pF, 1.47 % high. The corrected side, $s = 22.1965$ µm, measures 1.000001 pF, which is within one part in a million.

An error of 1.47 % on a single capacitor is usually something a circuit can live with. When isn't it?

Answer: A4, in Answers at the end.

Part 4: Three ways to build a capacitor

Which construction gives the most capacitance per square micrometre, and what does it cost you?

Block C measures the other two kinds of capacitor from Capacitors on a chip: MOS (an nFET gate and a pFET gate, biased at 1.8 V) and VPP (interleaved metal fingers). Block D then takes the bias away from one of the MOS capacitors.

1. Before reading the output, guess which construction is densest.
2. Find block C:

```
--- C. three technologies (farads) ---
c_nmos10 = 7.876883e-13
c_nmos11 = 9.999739e-13
c_pmos10 = 8.429616e-13
c_vpp = 1.164030e-13
```

`c_nmos10` is a 10 µm × 10 µm nFET gate, and `c_nmos11` is the 11.2699 µm square that makes one picofarad.

3. Find the checker's price list, which turns each density into the area one picofarad needs:

```
  what one picofarad costs
    recipe                              fF/um^2   um^2 for 1 pF   adders
    MIM  cap_mim_m3_1                    2.0297          492.68     4.47
    MOS  nfet_01v8 gate at 1.8 V         7.8732          127.01     1.15
    MOS  pfet_01v8 gate at 1.8 V         8.4296          118.63     1.08
    VPP  m1-m4 wafflecap                 0.9116         1096.96     9.96
```

   The last column is the area in DD103 4-bit ripple-carry adders, at 110.11 µm² each, as in Lab 01, Part 5.

4. Find block D, the 10 µm × 10 µm nFET capacitor with the gate at 0 V instead of 1.8 V.

You should see: one picofarad of MIM, drawn the way you would actually draw it, costs about four and a half ripple-carry adders. That's the 22.1965 µm square from Part 3, edges included, which is why its density is 2.0297 rather than 2.00. Block D reads:

```
--- D. the same MOS cap with the gate at 0 V (farads) ---
c_nmos_off = 2.207181e-13
```

That's 220.72 fF instead of the 787.69 fF of `c_nmos10`, a factor of 3.57 (the checker prints it too), and the only thing that changed was the gate voltage; every shape in the layout is the same.

The MOS capacitor is the densest by far. Why doesn't it go in a filter?

Answer: A5, in Answers at the end.

Part 5: The MOS capacitor's C–V curve

How does a MOS capacitor's value move with its gate voltage?

The numbers on the guide pages come from a second set of decks, in `labs/passives-decks`. Its `spice/c_area.spice` measures MIM, VPP and MOS capacitors at 1 kHz (the frequency cancels out of $C = |I|/(2\pi f)$, so any one works; see ngspice deck recipes, section 2), and its `spice/c_moscap_cv.spice` sweeps one MOS capacitor's gate from 0 V to 1.8 V in 100 mV steps.

1. Go to the `passives-decks` package and run both:

```
cd ../passives-decks
make capacitor
```

2. Divide each MIM capacitance in the first block, `--- capacitance in farads ---`, by its drawn area (1, 100 and 900 µm²).

3. Read `c_000mv` through `c_180mv` in the second block, `--- MOS cap, 10 um x 10 um gate ...`, and find the voltage range where the capacitance climbs fastest.

4. Go back to the lab folder:

```
cd ../lab-02-how-big-is-a-picofarad
```

You should see: MIM densities of 2.64, 2.07 and 2.02 fF/μm², falling as the plate grows. The sweep starts at 220.7182 fF at 0 V (`c_000mv = 2.207182e-13`), climbs steeply between 0.3 V and 0.7 V, and flattens out toward 787.6883 fF at 1.8 V (`c_180mv = 7.876883e-13`). This is the curve plotted on Capacitors on a chip.

What sets the voltage where the curve is steepest?

Answer: A6, in Answers at the end.

Part 6: Size your own capacitors

Can you size two MIM plates from your own three-term fit so that each lands within 1 % of its target?

`spice/my_cap.spice` has three capacitors, and you size the two MIM plates:

	device	target
C1	`cap_mim_m3_1`, square	250 fF
C2	`cap_mim_m3_1`, square	20 fF
C3	`nfet_01v8` gate at 1.8 V, square	250 fF

The file ships with all three sized from the area term alone, so the check in Part 1 failed on two of them:

```
  C1  cap_mim_m3_1, square
    target    250.00 fF
    got       257.36 fF   (+2.94 %)   FAIL
  C2  cap_mim_m3_1, square
    target     20.00 fF
    got        22.07 fF   (+10.35 %)   FAIL
  C3  nfet_01v8 gate at 1.8 V, square
    target    250.00 fF
    got       250.79 fF   (+0.31 %)   PASS
```

1. Before you change anything, work out why C3 passed with the naive sizing and the other two did not. C3 is in the file so that you'll ask this question.

   Answer: A7, in Answers at the end.

2. Solve for the side lengths of C1 and C2 using the three-term fit from Part 2 ($2.000000\,s^2 + 0.659991\,s - 0.017742$ fF), and write them down. Leave C3 as it is. The checker's explanation of each failure gives the side it would use, but compare yours with it only after you've worked out your own.
3. Edit the sizes in `spice/my_cap.spice`. Write them as plain microns, with no `u` suffix.
4. Re-run your file and the checker:

```
make mine check
```

You should see: all three capacitors marked `PASS`, then

```
PASS -- every number matches the reference run.
```

Part 7: The three inductors

What inductors does SKY130 give you, and what do they cost in area?

`make` already ran `make henry`, which sweeps the three SKY130 inductor models from 100 MHz to 100 GHz.

1. Find out how many inductors SKY130 offers:

```
ls /foss/pdks/sky130A/libs.ref/sky130_fd_pr/spice/ | grep "__ind_"
```

```
sky130_fd_pr__ind_03_90.model.spice
sky130_fd_pr__ind_05_125.model.spice
sky130_fd_pr__ind_05_220.model.spice
```

   There are three fixed devices, with no width or turn-count parameters, so there is nothing to size. With the resistor in Lab 01, any value you wanted was just a number you typed.

2. Find the inductor table in the checker's output:

```
    inductor        L @100 MHz    Q max   at GHz   self-res GHz  inner um^2
    ind_05_220        9.9220 nH   12.718   1.3717         3.5236       48400
    ind_05_125        5.7857 nH   13.083   2.2539         6.5212       15625
    ind_03_90         1.5208 nH   21.126   6.3668        16.5015        8100
```

3. The last column comes from the PDK's own LVS deck, which describes these devices in a comment. Check it:

```
grep "# # RF Inductor" /foss/pdks/sky130A/libs.tech/klayout/lvs/sky130.lvs
```

```
# # RF Inductor with 3 turns and internal diameter of 90um
# # RF Inductor with 5 turns and internal diameter of 125um
# # RF Inductor with 5 turns and internal diameter of 220um
```

   A hole 220 µm across in the middle is 48,400 µm², and the five turns of metal go around the outside of it, so 48,400 µm² is only a lower bound on the inductor's footprint.

4. Now look at the low-frequency end, which is what the guide page quotes, by running the `passives-decks` inductor deck:

```
cd ../passives-decks
make inductor
cd ../lab-02-how-big-is-a-picofarad
```

You should see: `q_1khz = 1.50516e-05` for `ind_05_220` in the `make inductor` output, which means that at 1 kHz the largest inductor is almost entirely resistance, as Inductors on a chip explains. The best Q for `ind_03_90` is about 21.1, near 6.4 GHz, which matches the `henry` table.

Here are the three passives side by side, all from your own runs in the same process:

you drew	area	in ripple adders
1 kΩ resistor (Lab 01, strip only, no contact heads)	1.96 µm²	0.018

you drew	area	in ripple adders
1 pF MIM capacitor	492.68 µm²	4.5
9.92 nH inductor	≥ 48,400 µm²	≥ 440

On a schematic the three symbols are the same size, but in silicon their areas span a factor of twenty-five thousand.

A switching regulator needs inductors measured in microhenries. Could you put 10 µH on the chip?

Answer: A8, in Answers at the end.

Part 8: A 2.4 GHz resonator

Can you land a resonator on 2.4 GHz using only numbers you measured?

`spice/tank.spice` puts `ind_03_90` in parallel with a MIM capacitor and looks for the peak in its impedance. The capacitor is sized from numbers you already have:

$$L(2.4\,\mathrm{GHz}) = 1.554019\mathrm{nH} \Rightarrow C = \frac{1}{(2\pi f)^2 L} = 2829.85\mathrm{fF} \Rightarrow s = 37.4508\mu\mathrm{m}$$

using the three-term fit from Part 2.

1. Find the `ind_03_90` lines in the `henry` output from Part 1:

   ```
   --- B. ind_03_90 at 2.4 GHz ---
   l_090_2g4 = 1.554019e-09
   q_090_2g4 = 1.086413e+01
   r_090_2g4 = 2.157013e+00
   ```

   Notice that 1.554019 nH is a little more than the 1.5208 nH the inductor shows at 100 MHz. At 2.4 GHz you are already at a seventh of its self-resonant frequency, and its apparent inductance has risen by 2.2 %.

2. Do the sizing arithmetic above yourself, from that inductance and the three-term fit.

3. Find the tank result in the output from Part 1. `fpeak` is the frequency of the impedance peak, and the number after `with=` is the peak impedance in ohms:

   ```
   fpeak               =  2.40000e+09 with=  2.50130e+02
   ```

You should see: resonance at 2.4000 GHz, exactly where you aimed, with a peak impedance of 250.13 Ω. Since the resonance lands on the digit, your capacitor model, your inductor reading and your arithmetic must all agree. At 2.4 GHz the inductor has a Q of 10.86 and 2.157 Ω of series resistance.

The peak impedance is 250.13 Ω and the inductor's Q is 10.86. Work out how the two are related. Could a better capacitor have made the resonator better?

Answer: A9, in Answers at the end.

The whole resonator takes 8,100 µm² of inductor (a lower bound) plus 1,403 µm² of capacitor, the area of 86 ripple adders for one tuned circuit.

Finishing

You're done when `make mine check`, run on the C1 and C2 sides you worked out in Part 6, prints `PASS -- every number matches the reference run.`

An AC analysis has no randomness in it, so on the pinned image (ngspice-47, `sky130A`, `tt` corner, 27 °C) your numbers will match the ones on this page exactly.

Question: C1 and C2 were sized with the same mistake. Why did C2 miss by 10.35 % and C1 by only 2.94 %?

Answer: A10, in Answers at the end.

After yours pass, read `solutions/README.md` for the reference sizes, why C3 passed, and a check of what inductor geometry the PDK's GDS actually contains.

Submit: nothing for this lab.

Going further (optional)

None of these is in the time estimate, and no later part depends on them. The last line of this page says what to do next on the Course path.

Plot the inductor sweep

`make henry` also leaves the whole 100 MHz to 100 GHz sweep in `results/ind_sweep.txt`. To see it as a plot (there's nothing to install), run:

```
python3 src/plot_inductors.py      # -> results/inductors.svg
```

It prints `wrote results/inductors.svg`. The plot marks each self-resonance, where the apparent inductance crosses zero. Above that frequency the device behaves as a capacitor, even though it is still called an inductor.

Part 9 (optional): Parasitic capacitance in SiliWiz

Parasitic capacitance is easier to believe once you see that a wire is a bar hanging over a plane, and SiliWiz draws that cross-section live as you edit the layout.

1. Open **https://app.siliwiz.com/?preset=blank**.
2. Draw a **metal2** rectangle. Draw a second one beside it, not touching.
3. Label one `in` and the other `out` (click, then **Set Label** or press `S`).
4. Tick **Show SPICE** at the bottom and find the line starting `C0 out in`. That line is the coupling capacitance between two wires you never meant to connect.
5. Drag the rectangles closer together and watch the number.
6. Now put one on **metal1** and the other on **metal2**, overlapping.

You should see: the `C0 out in` value climbs as the two metal2 wires get closer. With one wire on metal1 and the other on metal2, overlapping, the capacitance is lower than for the two metal2 wires at their closest, because the vertical gap between metal layers is bigger than the horizontal gap you can draw between two wires on the same layer.

There is a guided walkthrough at https://tinytapeout.com/siliwiz/parasitics/, and the capacitor tutorial at https://tinytapeout.com/siliwiz/capacitors/ builds the same three kinds of capacitor as Capacitors on a chip.

> **Warning:** SiliWiz isn't SKY130. Its layer stack, permittivities and numbers are simplified for teaching, so use it for the shape of the answer (closer means more, a thicker gap means less, and edges matter), and use the PDK's `parameters/typical.spice` (see Parasitic capacitance) for the value.

Next on the Course path: read What to build instead of an inductor, then do Lab 03: A time constant in silicon, which puts your resistor and capacitor into the same circuit.

Answers

A1. Part 1: The area ladder

The small one comes out proportionally bigger, so the ratio you built differs from the ratio you designed, and measuring either capacitor on its own won't show you that.

A2. Part 2: Fit the area and edge terms

All three are negative, and they shrink as the plate grows. If the fit were only noisy, it would miss in both directions. A miss that always goes the same way and fades with size means the model is missing something small and constant, a term that doesn't scale with the plate at all. The checker's comment says the same thing:

```
-- every residual is negative. Something small and constant
   is missing, and it does not scale with the plate at all.
```

A3. Part 2: Fit the area and edge terms

The plate comes out smaller than the size you drew. The model computes with `wc = w + m3_dw`, where `m3_dw` is the **metal-3 width bias**, how much narrower the etched metal comes out than the mask. Substituting $s \to s - d$ into $2s^2 + 0.76s$ gives $2s^2 + (0.76 - 4d)s + (2d^2 - 0.76d)$. Setting $0.76 - 4d = 0.659991$ gives $d = 0.025002$ µm, and that same d predicts a constant term of −0.017751 fF, against the −0.017742 you measured. The checker prints both:

```
the edge term implies a metal etch bias of 25.00 nm per side,
which predicts a constant term of -0.017751 fF against the -0.017742 you measured.
```

So from five capacitance numbers you've measured a 25 nm etch bias, without anyone telling you it existed. It's the same thing you saw with `weff`, the effective width, in Lab 01, Part 4: the shapes on the silicon differ from the shapes on the mask, and the model accounts for the difference even when the datasheet's headline number doesn't mention it.

A4. Part 3: Size one picofarad

When the error lands on a ratio of two capacitors. A 1.47 % error in the ratio gives you a switched-capacitor filter with the wrong corner frequency, or an ADC that is a quarter of a bit non-linear. Capacitor ratios are the most precise thing you can get on a chip (better than 0.1 % is routine), and ignoring the fringe term throws that precision away.

A5. Part 4: Three ways to build a capacitor

It uses the gate oxide, the thinnest insulator in the process, which is why it is the densest. But it is a transistor, so it only works as a capacitor while there is a channel under the gate to act as the other plate. Remove the bias and 72 % of the capacitance goes with it. A capacitance that depends on the voltage across it causes distortion, so MOS capacitors go on the supply rails by the million, where the voltage never moves, and stay out of signal paths. Density, linearity and mask cost (MIM needs extra masks; see the table on Capacitors on a chip) are separate properties, and no device is best at all three.

A6. Part 5: The MOS capacitor's C–V curve

The transistor's threshold voltage sets it. As you found in Part 4, a MOS capacitor only has its full value while there is a channel under the gate to act as the other plate, and the channel forms around the threshold.

A7. Part 6: Size your own capacitors

All three naive sizings dropped the edge term. The edge term is a perimeter effect, so what matters is the ratio of perimeter to area, which for a square of side s is $4/s$ and grows as the plate shrinks. C3′s dielectric is gate oxide, so its area term is about four times the MIM's, while its edge term comes mostly from the small gate-to-source and gate-to-drain overlaps. With an area term that large, the edges add comparatively little. The lab's `solutions/README.md` explains this at more length.

A8. Part 7: The three inductors

No. An inductance of 9.92 nH is no use for anything but radio. Even if inductance scaled linearly with area (it doesn't; a spiral does considerably worse), 10 µH would need 4.9×10^7 µm², which is 49 mm², larger than most entire chips. That's why every switching regulator you've seen has an inductor soldered next to the chip rather than inside it. It also answers the question on the course's home page, how you fabricate an inductor on a chip: mostly, you don't.

A9. Part 8: A 2.4 GHz resonator

$250.13/(2\pi \times 2.4\ \text{GHz} \times 1.554\ \text{nH}) = 10.7$, close to the inductor's own Q at 2.4 GHz, 10.86. So the capacitor contributed almost no loss, and nearly all of the resonator's loss comes from the inductor, which leaves a better capacitor almost nothing to improve. That is the usual case on silicon, and it is why people quote on-chip Q for inductors and never for capacitors.

A10. Finishing

They're the same type of plate, but C2′s is much smaller, and the edge term's share of a square plate grows as $4/s$ as the plate shrinks (Part 6, step 1). That's why the same mistake costs C2 3.5 times as much.