4. Full-Custom Analog Design Methodology

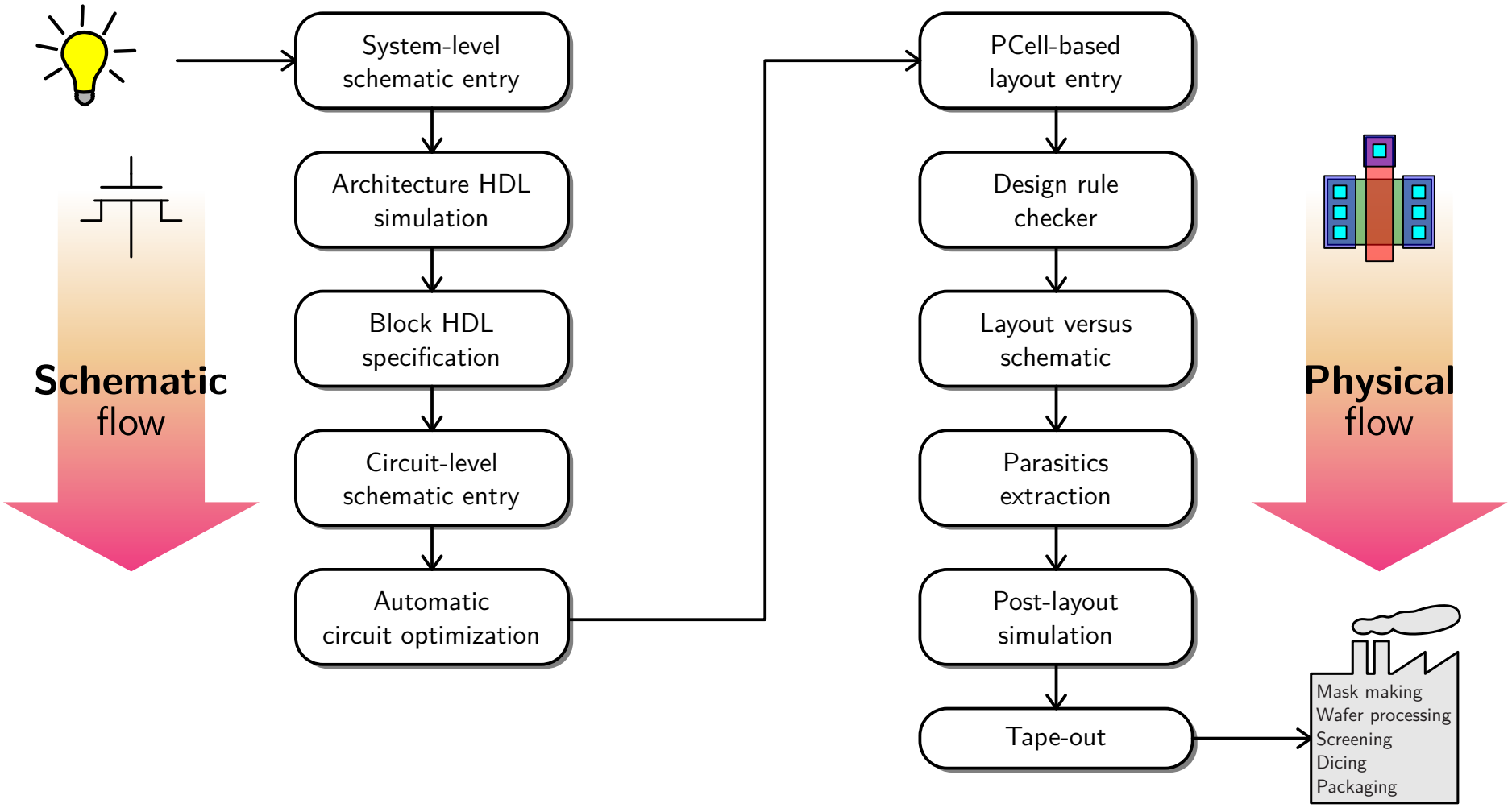
Francesc Serra Graells

francesc.serra.graells@uab.cat
Departament de Microelectrònica i Sistemes Electrònics
Universitat Autònoma de Barcelona

paco.serra@imb-cnm.csic.es
Integrated Circuits and Systems
IMB-CNM(CSIC)

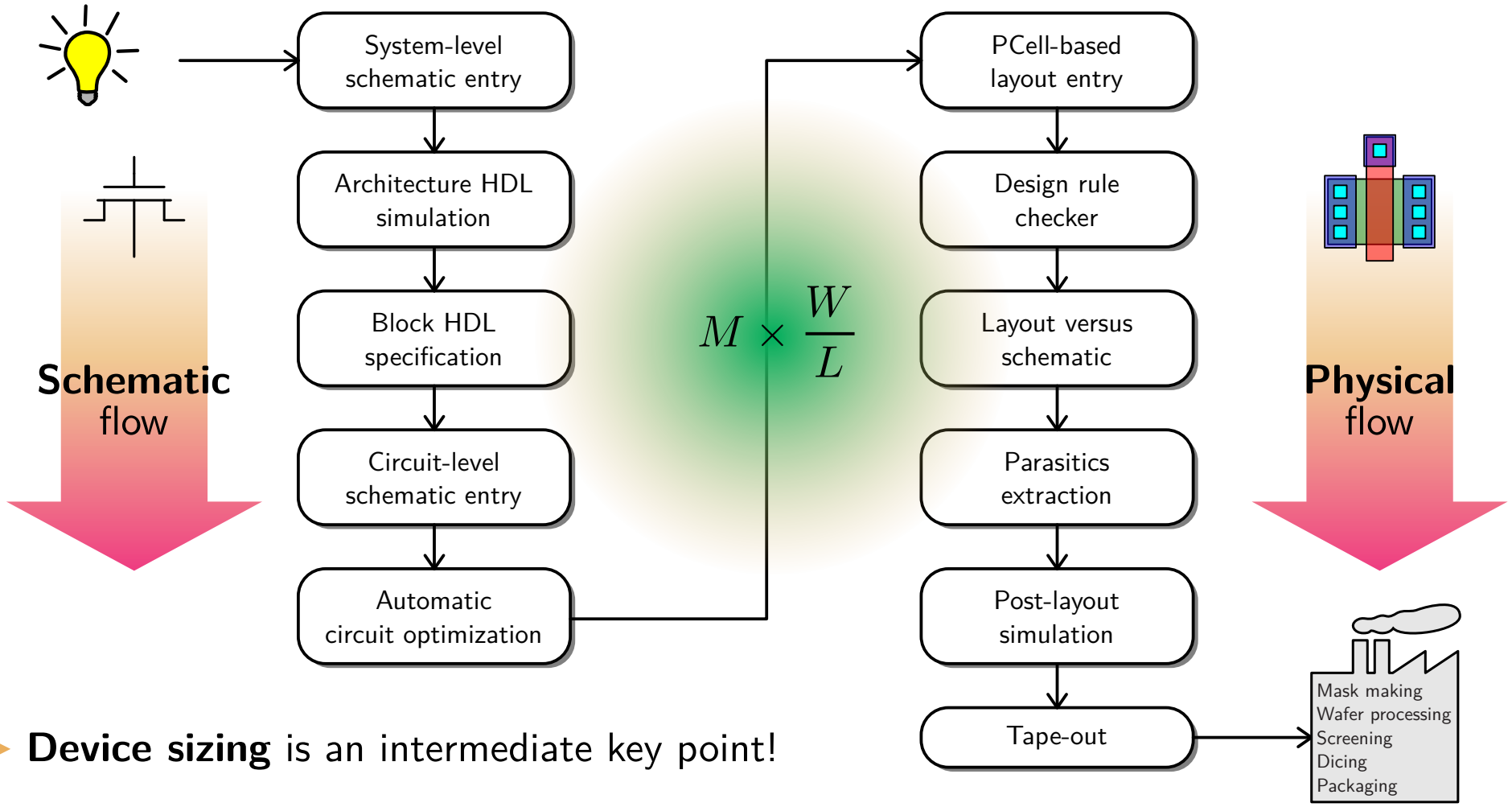
Full-Custom Analog IC Design Flow

- From circuit **idea to layout** masks...
- **EDA**-assisted methodology



Full-Custom Analog IC Design Flow

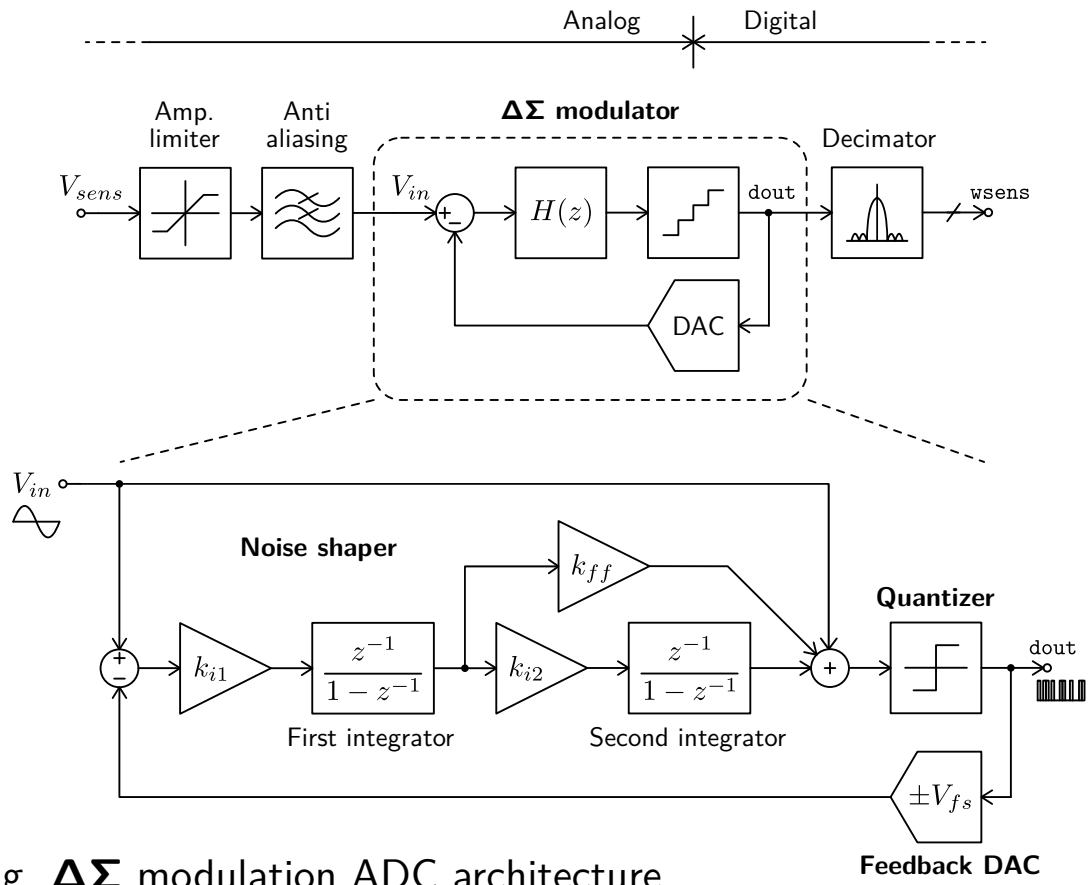
- From circuit **idea to layout** masks...
- **EDA**-assisted methodology



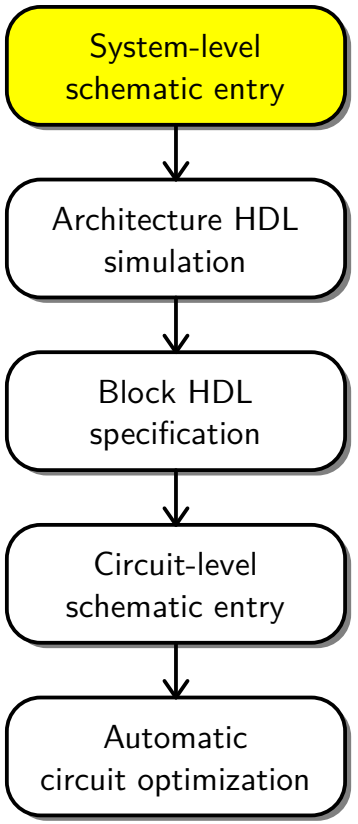
- **Device sizing** is an intermediate key point!

Full-Custom Schematic Design

- **Architecture** selection according to system and application specifications:



e.g. **ΔΣ** modulation ADC architecture



- Signal processing performance
- Communications standards
- Power constraints
- Testbility requirements
- ... and many more!

Full-Custom Schematic Design

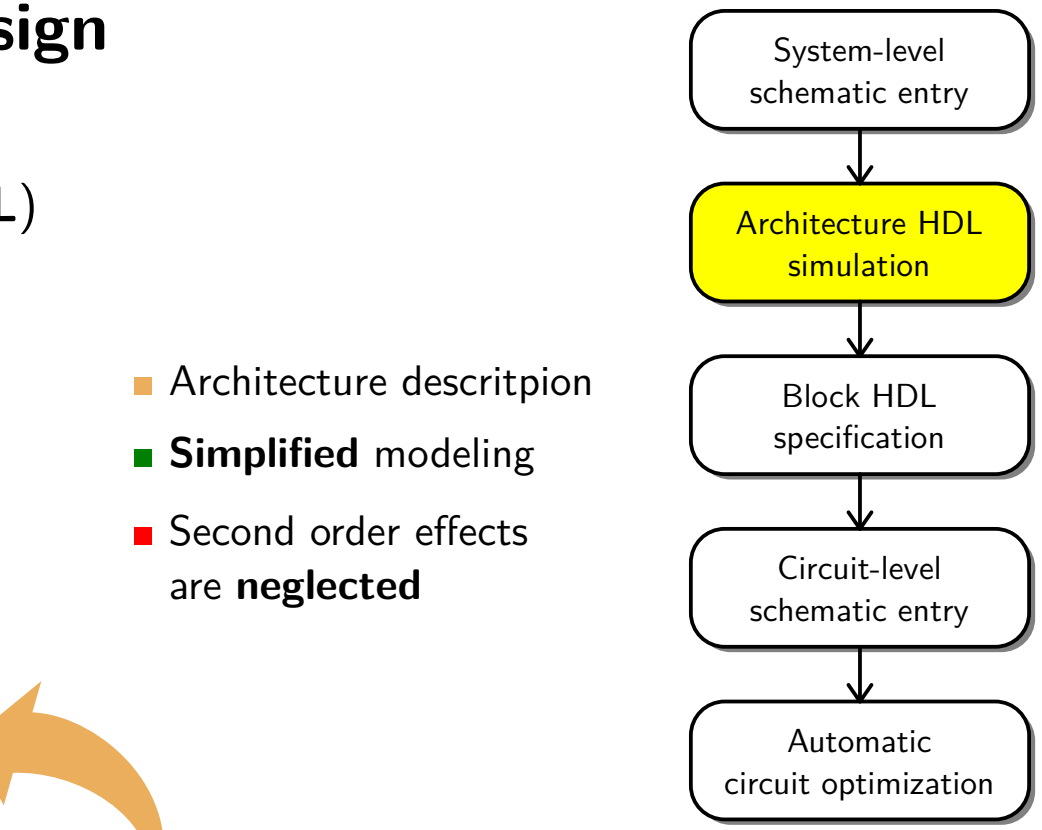
- System modeling through any hardware descitpion language (**HDL**)

zinteg2lim.mod

```
void cm_zinteg2lim(ARGS) {
 inp = INPUT(inp);            /* Retriving input values */
 clk = INPUT_STATE(clk);
 pos_edge = PARAM(pos_edge);  /* Retrieving parameters */
 out_max = PARAM(out_ax);
 ...
 switch (ANALYSIS) {
  case TRANSIENT:
   if ((*clk_mem==ONE)&&(clk==ZERO)) {/* Neg. clk edge */
    if (pos_edge==FALSE)
     action = SAMPLING_INTEGRATION;
   } else {
    if ((*clk_mem==ZERO)&&(clk==ONE)) {/* Pos. clk edge*/
     if (pos_edge==TRUE)
      action = SAMPLING_INTEGRATION;
    } else {                          /* No clock edge */
     action = HOLDING;
   }
 ...
   switch (action) {
    case SAMPLING_INTEGRATION:
     *inp_mem = inp;
     out = *out_mem+*inp_mem;
     if (out<out_min) { out = out_min; }    /* Limiter */
     if (out>out_max) { out = out_max; }
     *out_mem = out;
     break;
    case HOLDING:
     out = *out_mem;
}}}
```

e.g. **XSpice**

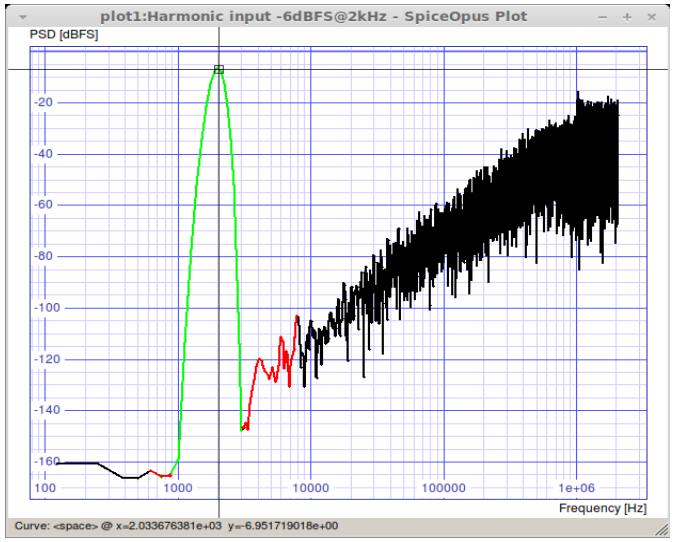
- Architecture descritpion
- **Simplified** modeling
- Second order effects are **neglected**

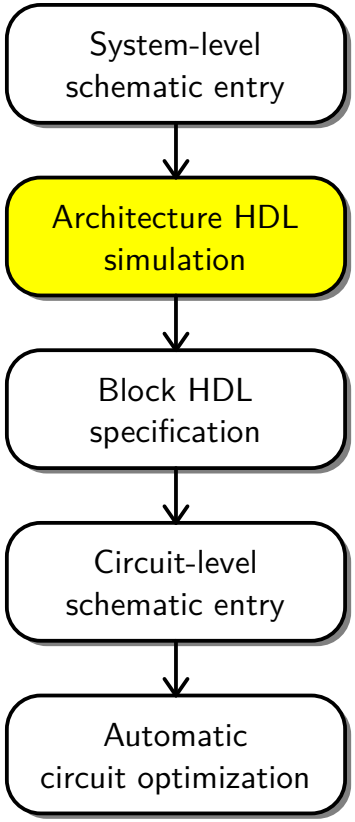


Full-Custom Schematic Design

- HDL numerical simulation of the full system using **event-based** engines
  - Architecture evaluation
  - Simulation **speed-up** by orders of magnitude
  - **Ideal** response (maximum possible performance)

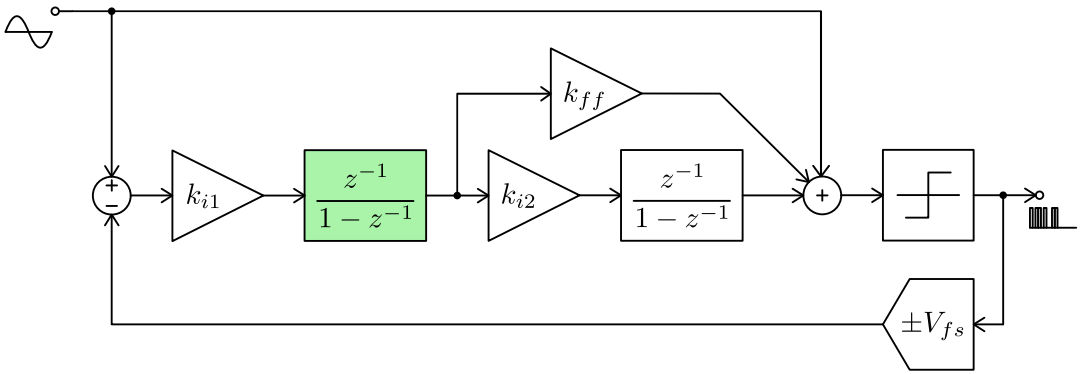
e.g. **SpiceOpus**





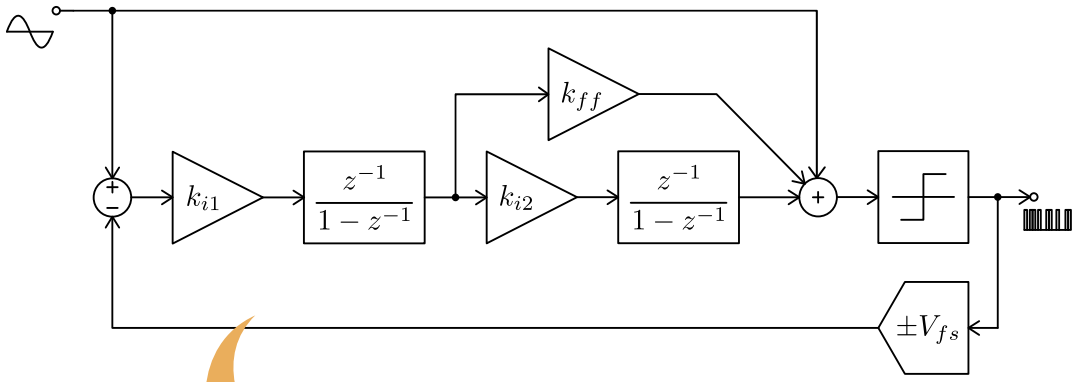
dsm-arch.sub

```
.subckt dsm_arch vin dclk dout
asumin [%v(vin) %v(vdac)] %v(verr) msumin
.model msumin usummer(sign=[1.0 -1.0])
aki1 %v(verr) %v(vint1in) mki1
.model mki1 kgain(k=0.3)
azint1 %v(vint1in) %d(dclk) %v(vint1out) mzint1
.model mzint1 zinteg2lim(pos_edge=0 out_ic=0.0
 + out_min=-5.0 out_max=5.0)
aki2 %v(vint1out) %v(vint2in) mki2
.model mki2 kgain(k=0.7)
azint2 %v(vint2in) %d(dclk) %v(vint2out) mzint2
.model mzint2 zinteg2lim(pos_edge=0 out_ic=0.0
 + out_min=-5.0 out_max=5.0)
akff %v(vint1out) %v(vkffout) mkff
.model mkff kgain(k=2.0)
asumout [%v(vint2out) %v(vkffout) %v(vin)]
 + %v(vquantin) msumout
.model msumout usummer(sign=[1.0 1.0 1.0])
aquant %v(vquantin) %d(~dclk) %d(dout) mquant
.model mquant quant2lsh(inp_th=0.0 out_ic=0 pos_edge=0
 + t_rise=1e-9 t_fall=1e-9)
adac %d(dout) %v(vdac) mdac
.model mdac dac2lsym(out_level=2.0)
.ends
```

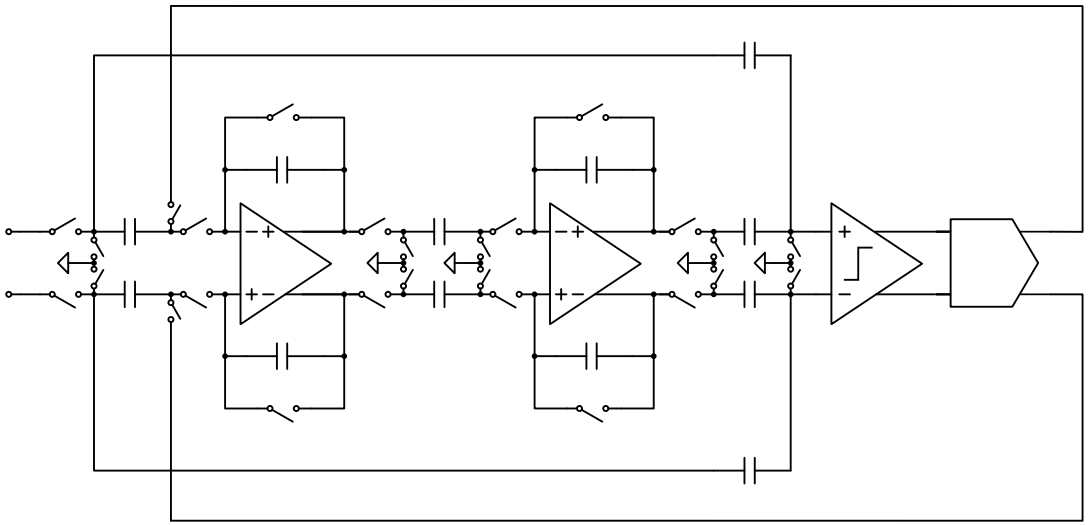


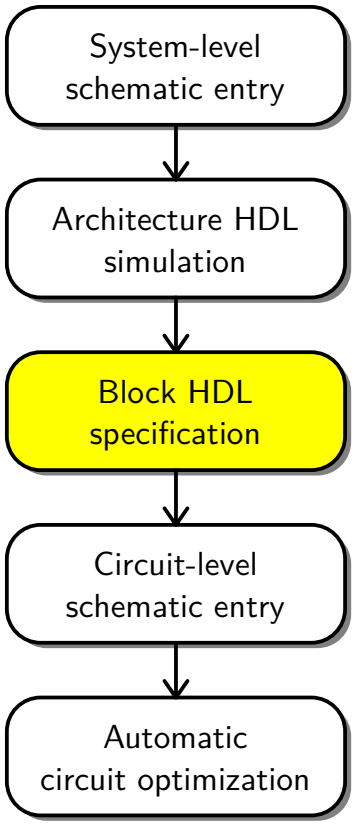
Full-Custom Schematic Design

- Choosing most suitable **circuit technique** according to:



e.g. switched-capacitor (**SC**) technique

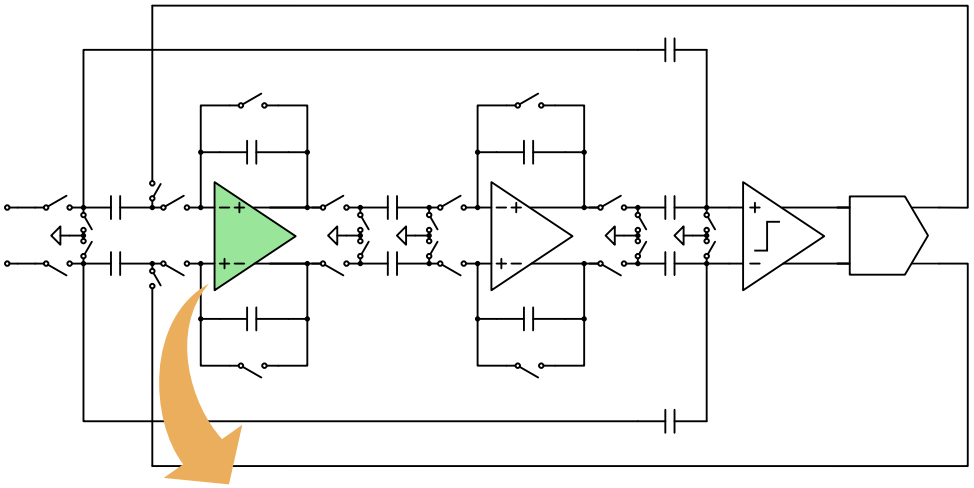




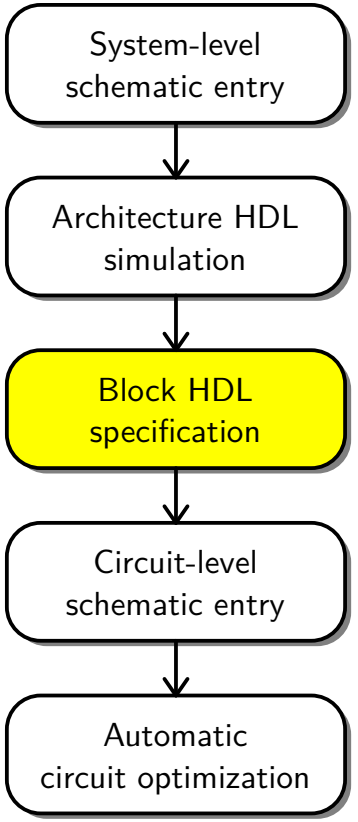
- **CMOS** technology options
- Circuit sensitivity against process, supply and temperature (**PVT**)
- IC **operation** conditions (calibration, testability...)
- **External** components available

Full-Custom Schematic Design

- **Splitting** system design problem into independent **blocks**:

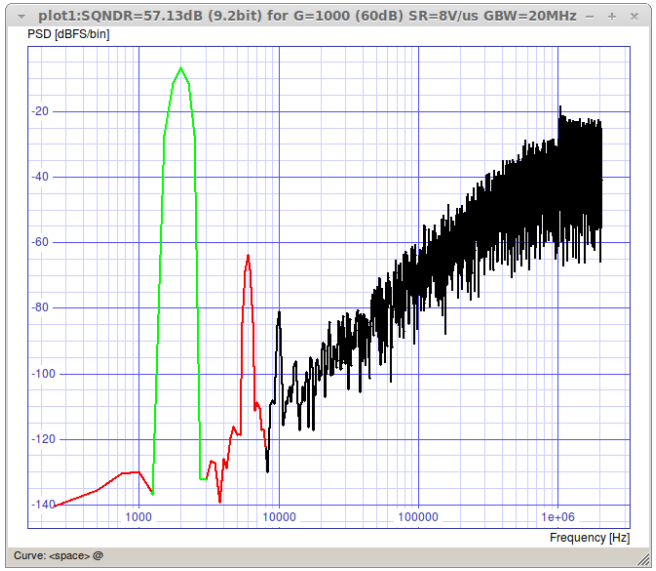


- Modeling **second order** effects due to limited block performance
- Electrical simulation of each **block** is feasable
- Transistor-level **final** simulation of overall system is still required!



opamp.ifs

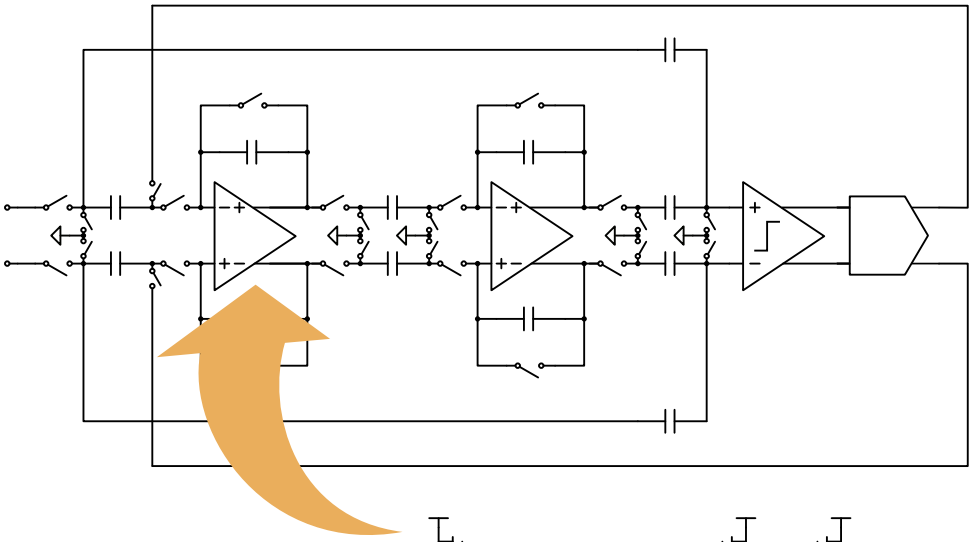
```
NAME_TABLE:
Spice_Model_Name:  opamp
C_Function_Name:   cm_opamp
Description:       "OpAmp macro"

PORT_TABLE:
Port_Name:         inp          inn          out
Description:       "pos. input" "neg. input" "output"
Direction:         in           in           out
Default_Type:      v            v            v
...
PARAMETER_TABLE:
Parameter_Name:    G            GBW          SR
Description:       "DC OL gain" "GBW"        "slew-rate"
Data_Type:         real         real         real
Default_Value:     1000         1e6          1e6
Limits:            [0 -]        [0 -]        [0 -]
...
PARAMETER_TABLE:
Parameter_Name:    out_min              out_max
Description:       "lower out limit"    "upper out limit"
Data_Type:         real                 real
Default_Value:     0                    5.0
Limits:            [0 5.0]              [0 5.0]
...
```

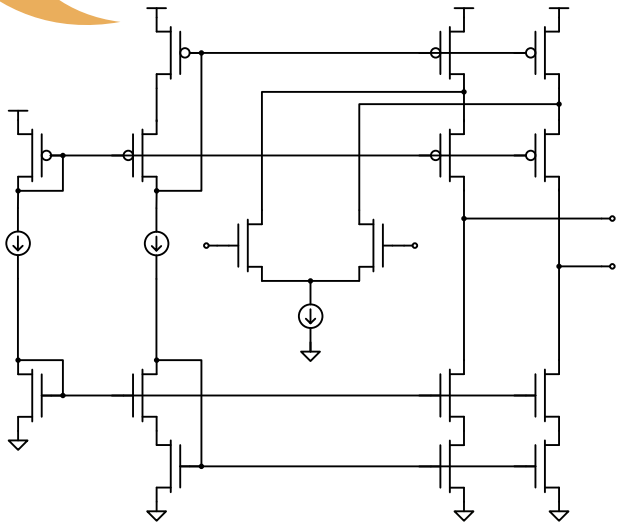


e.g. limited {G,GBW,SR} **Opamp** macro model

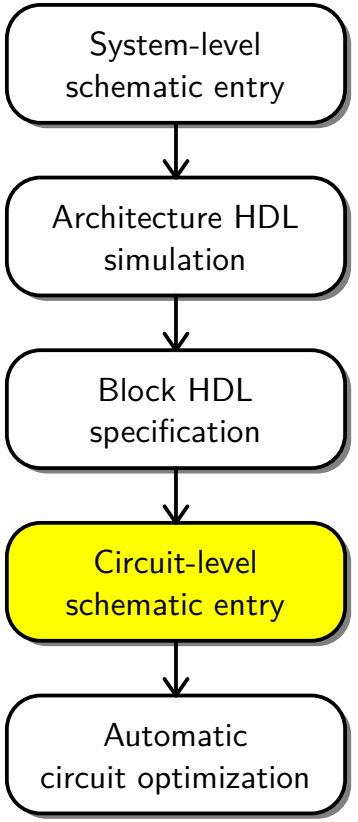
Full-Custom Schematic Design

- Choosing the particular **circuit topology** for each system block:





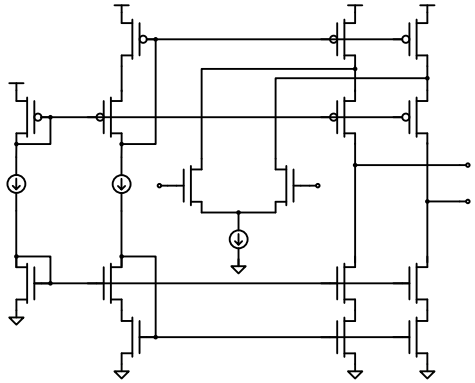
e.g. fully-differential cascode and folded **OpAmp**



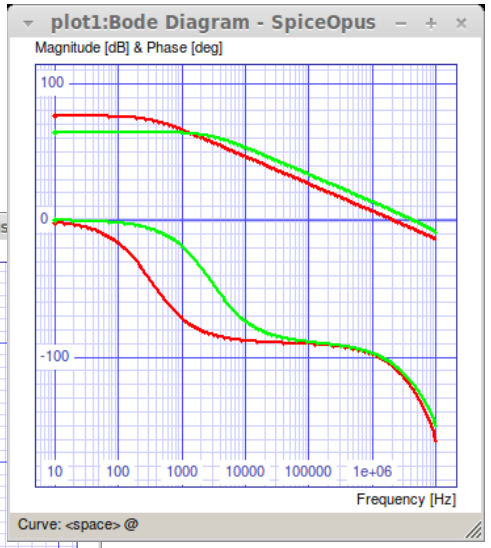
- **Separated** block tests
- **Target specs** from previous step e.g. {G,GBW,SR}
- Fast and accurate **electrical simulations**
- **Several** topologies can be easily investigated
- Inter-block **coupled** effects not covered!

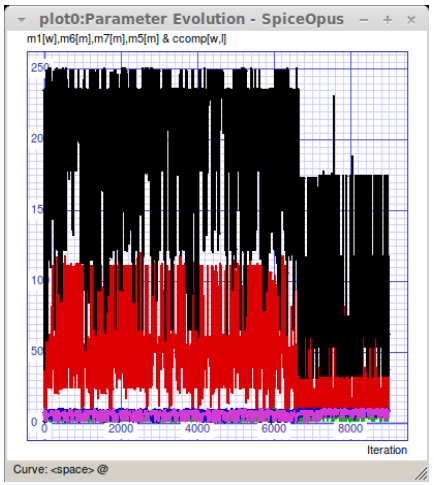
Full-Custom Schematic Design

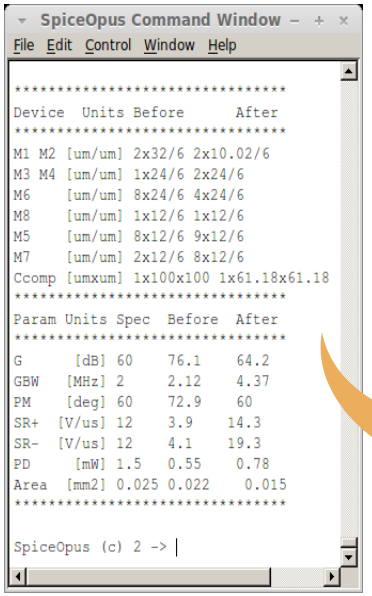
- Device **size optimization** at block level:



e.g. OpAmp
automatic optimization

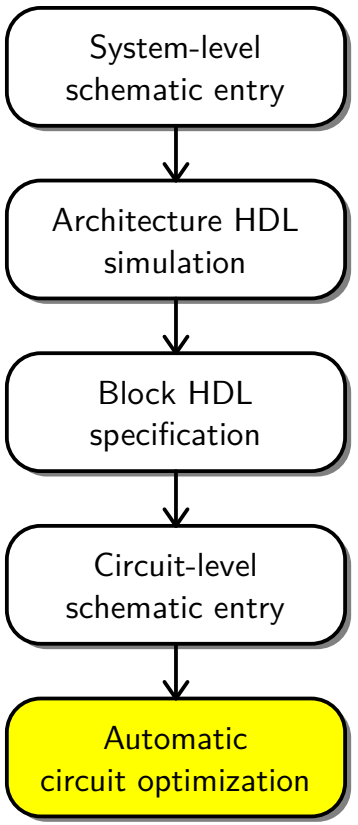






opamp_optimize.sp3

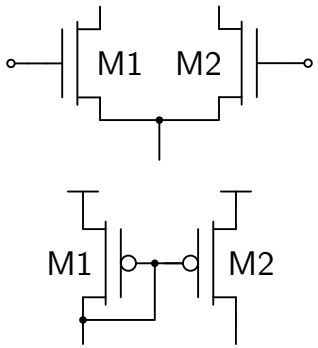
```
optimize
   parameter 0 @m1:xopamp[w] low 6u high 120u initial 32u
 " parameter 1 @m6:xopamp[m] low 1 high 10 initial 8
 " parameter 3 @ccomp:xopamp[w] low 25u high 250u
...
 " analysis 25 ac dec 50 10 10e6
 " analysis 26 let gmag=20*log10(mag(v(vout)))
 " analysis 27 let gph=phase(v(vout))
 " analysis 28 cursor c right gmag 0
 " analysis 29 let gbw=abs(frequency[%c])/1e6
 " analysis 30 let pm=180+gph[%c]
...
 " analysis 46 tran 1n 5u
 " analysis 47 cursor c right vout 2.1
 " analysis 48 let t1=time[%c]
 " analysis 49 cursor c right vout 2.9
 " analysis 50 let t2=time[%c]
 " analysis 51 let srpos=0.8/(t2-t1)*1e-6
...
 " implicit 0 op2.pd lt 1.5
 " implicit 1 op2.area lt 0.025
 " implicit 4 ac2.pm gt 60
 " implicit 5 tran2.srpos gt 12
 ...
 " cost 1/tran2.srneg+1/tran2.srpos+abs(60-ac2.pm)
 " method genetic elitism yes maxgen 1000
```



$$M \times \frac{W}{L}$$

MOSFET Sizing Analog Guidelines

- Transistor (also other devices) matching **ratios**:



multiplicity (**M**)
design for a common
channel aspect
ratio (W/L)

$$M \times \frac{W}{L}$$

non-minimum
channel length (**L**)
above design rules
for a given W/L

overall
M and **W/L**
design for a given
absolute L value

- Device output **impedance**:

$$\lambda \propto \frac{1}{L} \downarrow$$

- MOS **transconductance**:

$$M\frac{W}{L} \uparrow \quad g_{mg,s} \uparrow$$

- **Bandwidth**, technology **mismatching** and **flicker** noise:

$$C_{ox} = \frac{\epsilon_{ox}}{t_{ox}} MWL$$

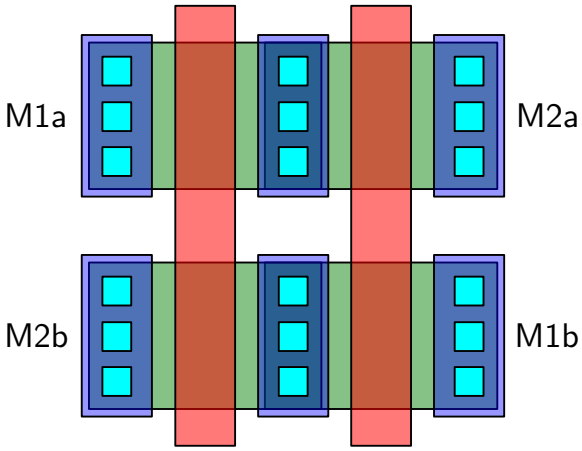
device area (**MWL**)
design for a given
M and W/L

$$\sigma\left(\Delta P\right) \simeq \frac{A_P}{\sqrt{MWL}}$$

$$\frac{dv_{nfk}^2}{df} = \frac{K_{fk}}{MWL}\frac{1}{f}$$

- Physical **layout** considerations:

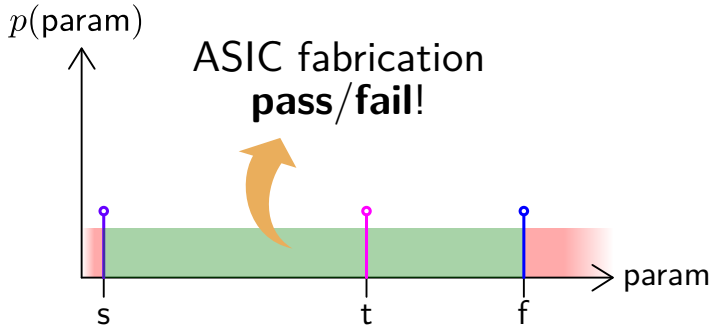
M integer
design for an
overall MW/L

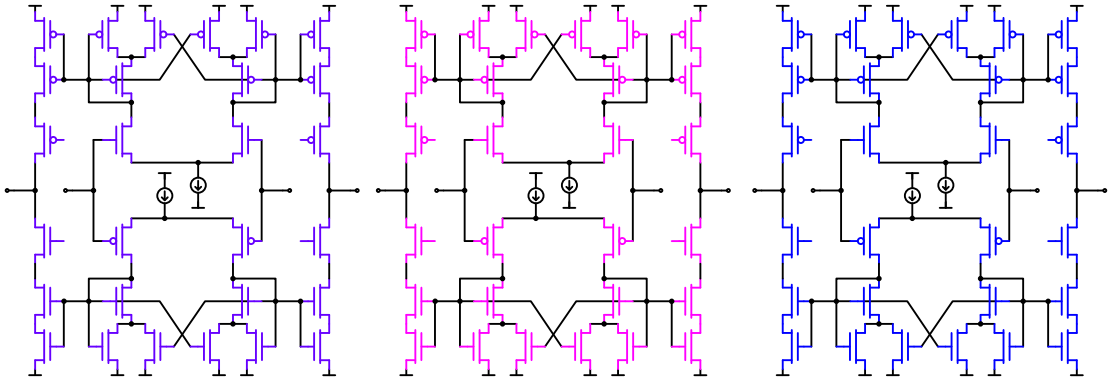


e.g. M **even** values for
common centroid layout techniques

Process Simulation

- **Global** deviations of model parameters:
  - Same change in **all devices** of the ASIC
  - Worse-case **corner** analysis: (t)yp, (f)ast, (s)low...





cnm25proc.lib

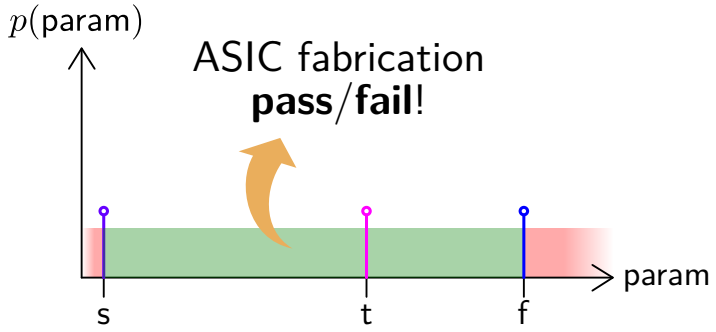
```
* CNM25 process corners

.lib common
.model cnm25modn nmos LEVEL = 2
+ TOX = 380E-10     VTO = {vton}     NSUB = 2.64E16   UO = {uon}
+ UCRIT = 1E4       UEXP = 6.86E-2   NFS = 7.11E11
+ DELTA = 2.20      RS = 93.8        LD = 9.13E-7     XJ = 8.24E-8
+ VMAX = 5.96E4     NEFF = 1.48      CJ = 3.50E-4     MJ = .40
+ CJSW = 5.95E-10   MJSW = .29       PB = .65
+ AF    =1.33  KF    =1e-29
.model cnm25modp pmos LEVEL = 2
+ TOX = 380E-10     VTO = {vtop}     NSUB = 1.36E16   UO = {uop}
+ UCRIT = 1E4       UEXP = 1.16E-1   NFS = 6.62E11
+ DELTA = 1.82      RS = 134.9       LD = 8.10E-7     XJ = 2.78E-9
+ VMAX = 1.20E5     NEFF = 6.67E-2   CJ = 3.82E-4     MJ = .35
+ CJSW = 7.38E-10   MJSW = .39       PB = .56
+ AF    =1.33  KF    =1e-29
.model cnm25cpoly c CJ= 4.227E-4  CJSW=0.0
.endl

.lib tt
.param vton=.942
.param vtop=-1.139
.param uon=648
.param uop=213
.lib 'cnm25proc.lib' common
.endl

.lib ss
.param vton=1.1
.param vtop=-1.3
.param uon=415
.param uop=131
.lib 'cnm25proc.lib' common
.endl

.lib ff
.param vton=0.7
.param vtop=-0.9
.param uon=881
.param uop=295
.lib 'cnm25proc.lib' common
.endl
```

$V_{TH} \uparrow$, $\beta \downarrow$ slow

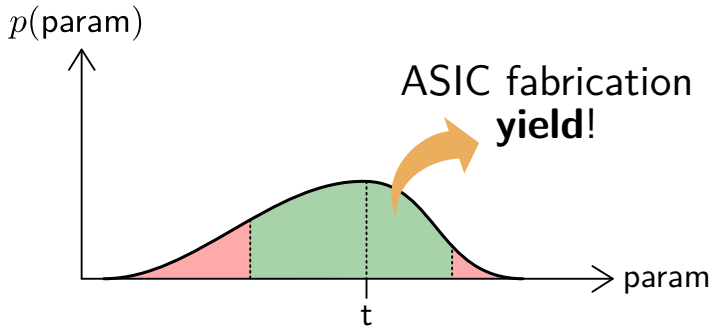
$V_{TH} \downarrow$, $\beta \uparrow$ fast

Process Simulation

- **Global** deviations of model parameters:
  - Same change in **all devices** of the ASIC
  - Worse-case **corner** analysis: (t)yp, (f)ast, (s)low...
  - Combined corners: process/voltage/temperature (**PVT**)



  - **Montecarlo** statistical analysis:

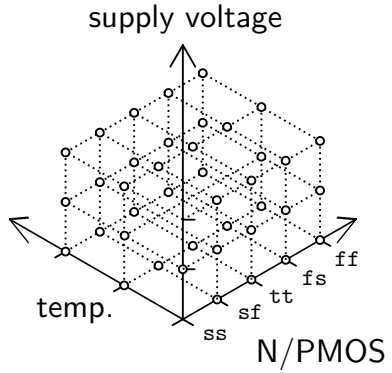


cnm25proc.lib

```
* CNM25 process corners

.lib common
.model cnm25modn nmos LEVEL = 2
+ TOX = 380E-10     VTO = {vton}     NSUB = 2.64E16   UO = {uon}
+ UCRIT = 1E4       UEXP = 6.86E-2   NFS = 7.11E11
+ DELTA = 2.20      RS = 93.8        LD = 9.13E-7     XJ = 8.24E-8
+ VMAX = 5.96E4     NEFF = 1.48      CJ = 3.50E-4     MJ = .40
+ CJSW = 5.95E-10   MJSW = .29       PB = .65
+ AF    =1.33  KF    =1e-29
.model cnm25modp pmos LEVEL = 2
+ TOX = 380E-10     VTO = {vtop}     NSUB = 1.36E16   UO = {uop}
+ UCRIT = 1E4       UEXP = 1.16E-1   NFS = 6.62E11
+ DELTA = 1.82      RS = 134.9       LD = 8.10E-7     XJ = 2.78E-9
+ VMAX = 1.20E5     NEFF = 6.67E-2   CJ = 3.82E-4     MJ = .35
+ CJSW = 7.38E-10   MJSW = .39       PB = .56
+ AF    =1.33  KF    =1e-29
.model cnm25cpoly c CJ= 4.227E-4  CJSW=0.0
.endl

.lib tt
.param vton=.942
.param vtop=-1.139
.param uon=648
.param uop=213
.lib 'cnm25proc.lib' common
.endl

.lib ss
.param vton=1.1
.param vtop=-1.3
.param uon=415
.param uop=131
.lib 'cnm25proc.lib' common
.endl

.lib ff
.param vton=0.7
.param vtop=-0.9
.param uon=881
.param uop=295
.lib 'cnm25proc.lib' common
.endl
```

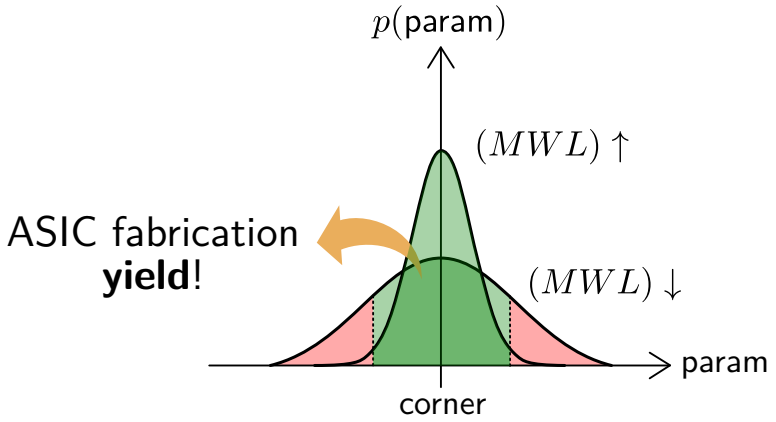
$V_{TH} \uparrow$ slow
$\beta \downarrow$

$V_{TH} \downarrow$ fast
$\beta \uparrow$



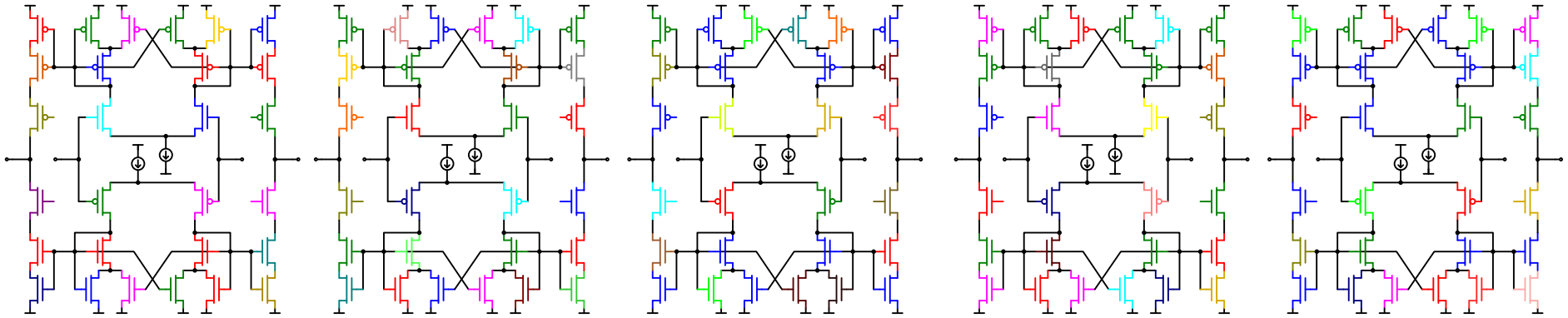
Mismatching Simulation

- **Local** deviations of model parameters:
  - Different change for **each device** of the ASIC
  - **Pelgrom Law**
  - **Montecarlo** statistical analysis:



cnm25match.lib

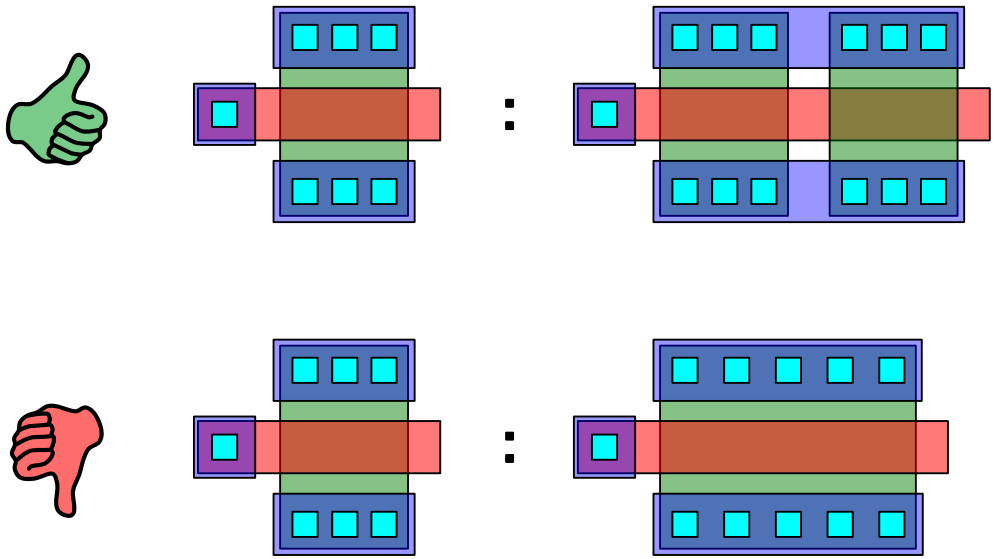
```
.param avto=30e-9
.param rauo=5e-8
.param rac=7.3e-8

.subckt cnm25modn d g s b param: w=3u l=3u
 + ad=0 as=0 pd=0 ps=0 m=1
.param vton=.942+rndgauss(avto/sqrt(m*w*l))
.param uon=648*(1+rndgauss(rauo/sqrt(m*w*l)))
.model modnlocal nmos level=2 vto={vton} uo={uon} tox=...
mn d g s b modnlocal w={w} l={l}
 + ad={ad} as={as} pd={pd} ps={ps} m={m}
.ends

.subckt cnm25cpoly t b param: w=30u l=30u m=1
.param cj=4.227E-4*(1+rndgauss(rac/sqrt(m*w*l)))
.model cpolylocal c cj={cj} cjsw=0.0
cpip t b cpolylocal w={w} l={l} m={m}
.ends
```



General Matching Rules

- **Unitary** elements

e.g. 1 : 2 ratio

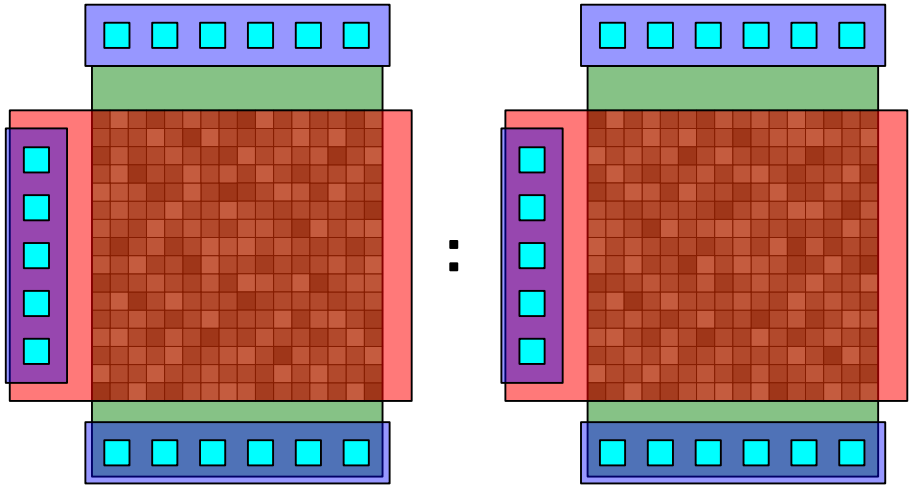


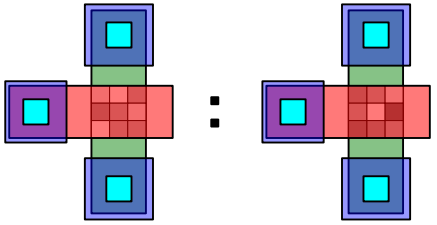
- Play with **multiplicity** only
- Same ratio for **second order** effects also (e.g. area and perimeter ratio in caps)
- **Larger area** for the overall array

General Matching Rules

- **Unitary** elements
- Large **area** devices

e.g. 1 : 1 ratio



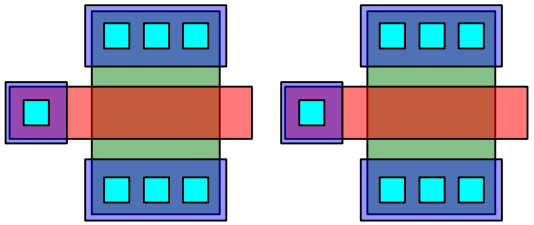


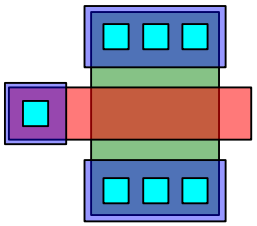
- Technology local **granularity** (e.g. distribution of dopants)
- **Pelgrom Law** $\propto \frac{1}{\sqrt{WL}}$

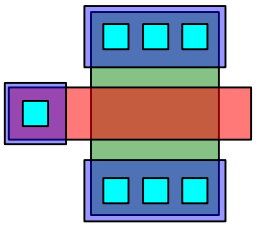
General Matching Rules

- **Unitary** elements
- Large **area** devices
- Minimum **distance**

e.g. 1 : 1 ratio



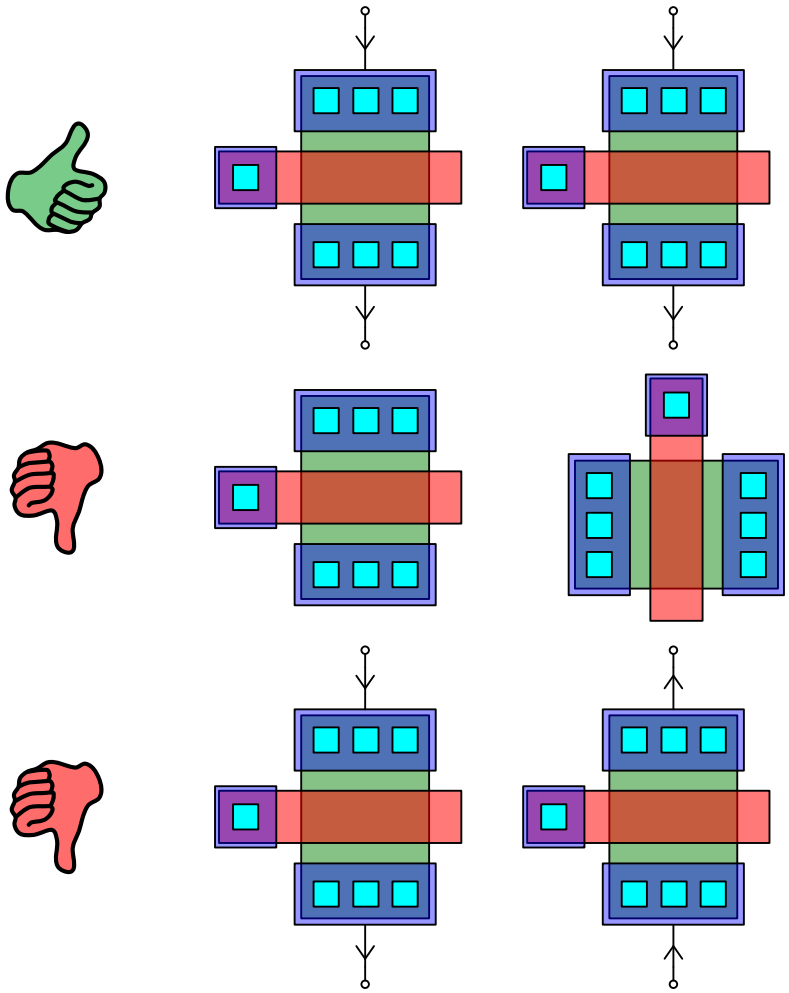




- Technology global **drifts** (e.g. gate oxide thickness slope)
- **Design rule** spacing limits

General Matching Rules

- **Unitary** elements
- Large **area** devices
- Minimum **distance**
- Same **orientation**

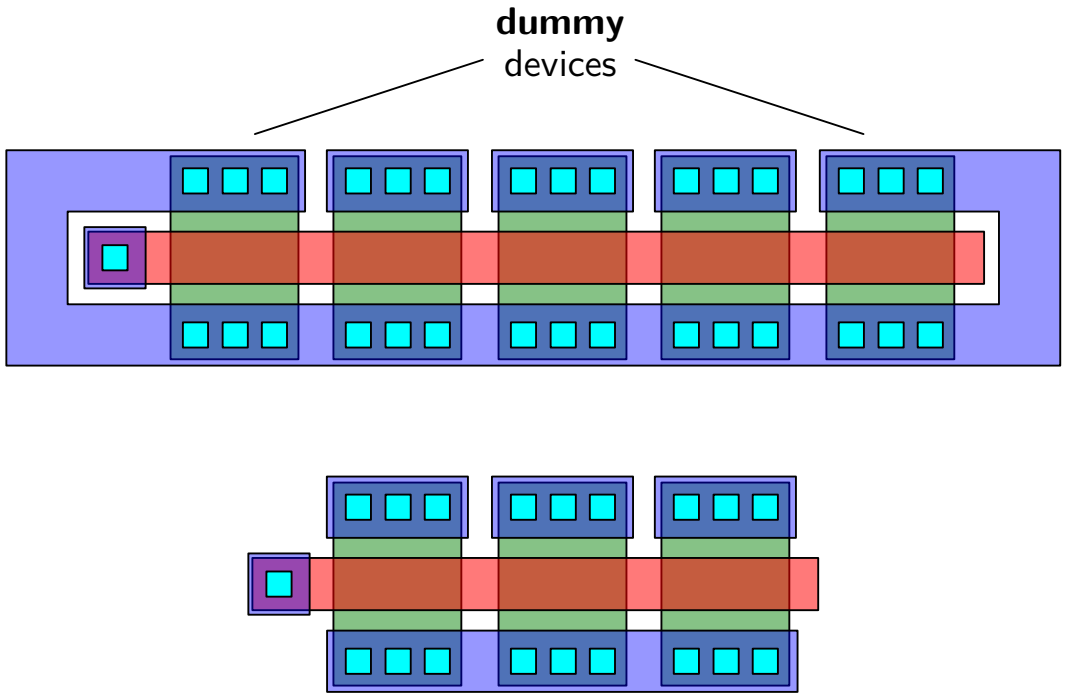


- **Anisotropic** materials
  (e.g. wafer crystal lattice orientation)
- Longer **routing** may be required...

General Matching Rules

- **Unitary** elements
- Large **area** devices
- Minimum **distance**
- Same **orientation**
- Same **sorround**

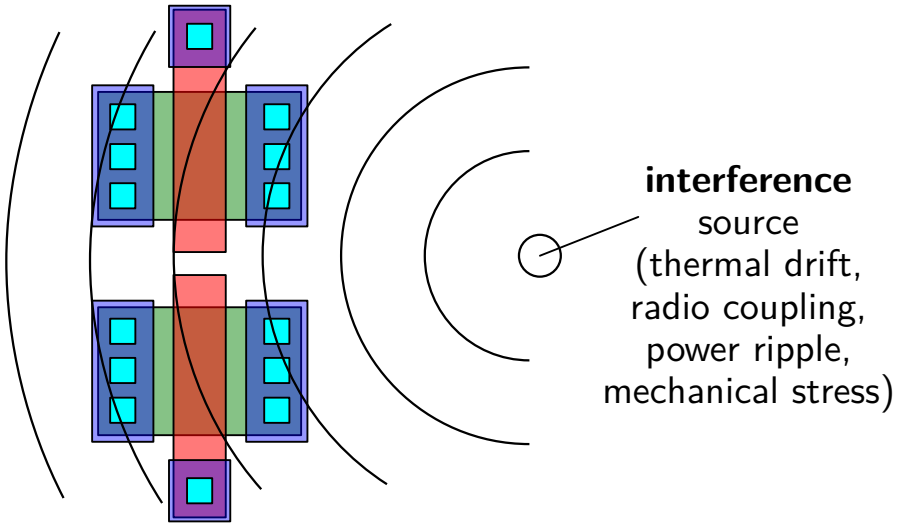
e.g. 1 : 1 : 1 ratios

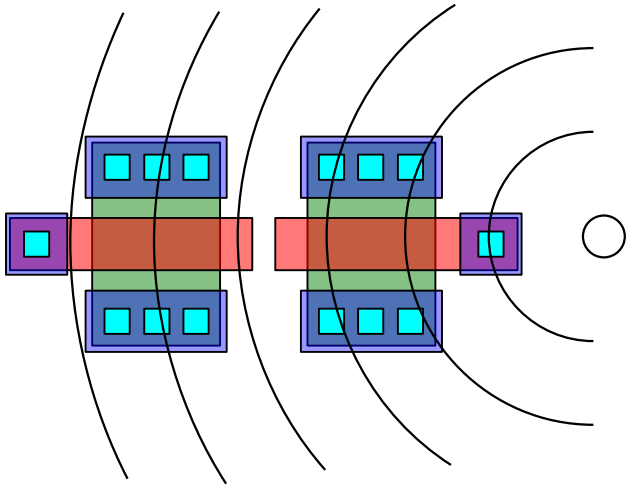


- **Inter-device** second order effects (e.g. parasitic RLC)
- **Larger area** for the overall array

General Matching Rules

- **Unitary** elements
- Large **area** devices
- Minimum **distance**
- Same **orientation**
- Same **sorround**
- Same **symmetry**





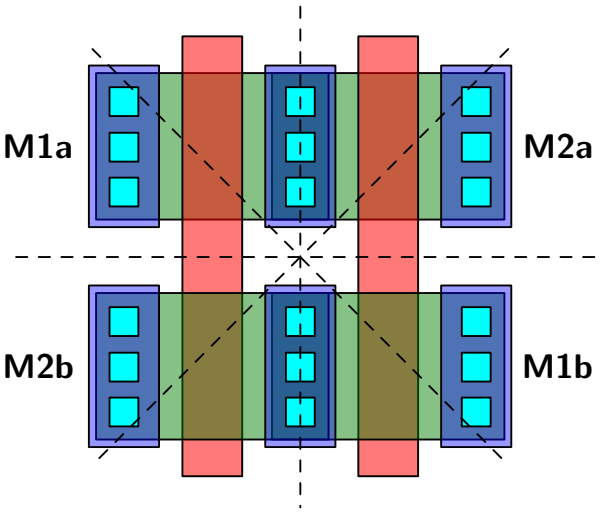
- **Differential** circuits (common mode interference)
- Complex **floorplan**

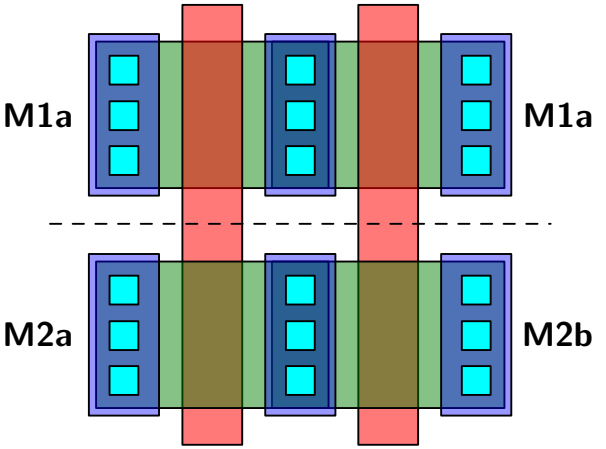
General Matching Rules

- **Unitary** elements
- Large **area** devices
- Minimum **distance**
- Same **orientation**
- Same **sorround**
- Same **symmetry**

- Compensation of **linear** gradients (and non-linear at short distance)
- **Longer routing** is usually required...

e.g. 2 : 2 ratio

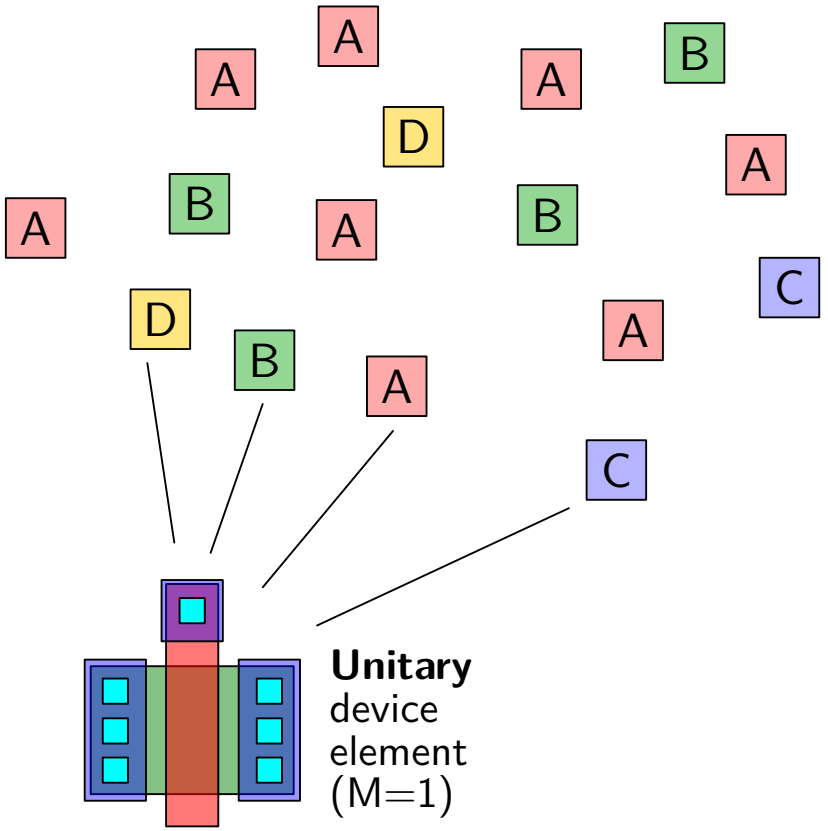




Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements

e.g. A : B : C : D ratios
8 4 2 2



Common Centroid Arrays

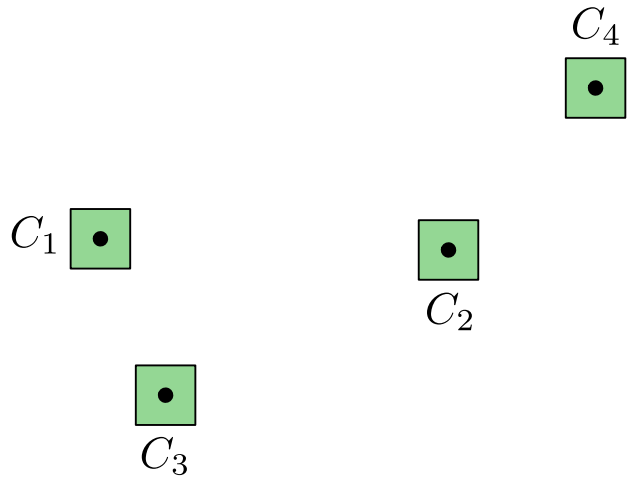
- Difficult to achieve for **large** and **multiple groups** of unitary elements
- Centroid as a **center-of-mass** concept, but with area weights...

each element: area A_i, center-of-mass coordinates C_i

$$\sum_{i=1}^{M} A_i \left(C_i - C_0\right) = 0$$

centroid coordinates

$$C_0 = \frac{\sum_{i=1}^{M} A_i C_i}{\sum_{i=1}^{M} A_i}$$



If all elements are of **same area**:

$$C_0 \equiv \frac{1}{M}\sum_{i=1}^{M} C_i$$

Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements
- Centroid as a **center-of-mass** concept, but with area weights...

each element: area A_i, center-of-mass coordinates C_i

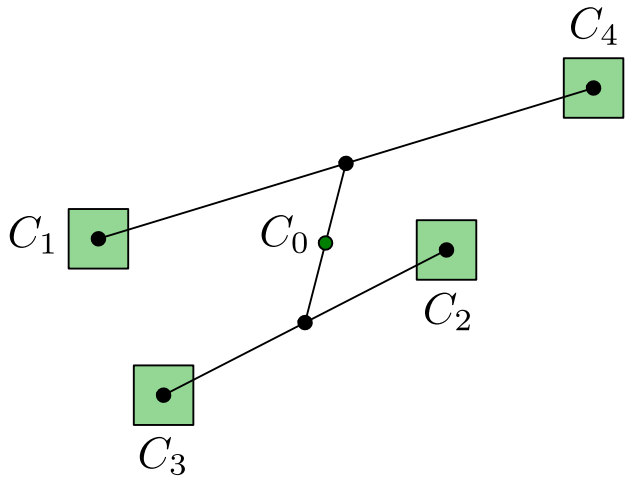
$$\sum_{i=1}^{M} A_i \left(C_i - C_0\right) = 0$$

centroid coordinates

$$C_0 = \frac{\sum_{i=1}^{M} A_i C_i}{\sum_{i=1}^{M} A_i}$$

If all elements are of **same area**:

$$C_0 \equiv \frac{1}{M} \sum_{i=1}^{M} C_i$$

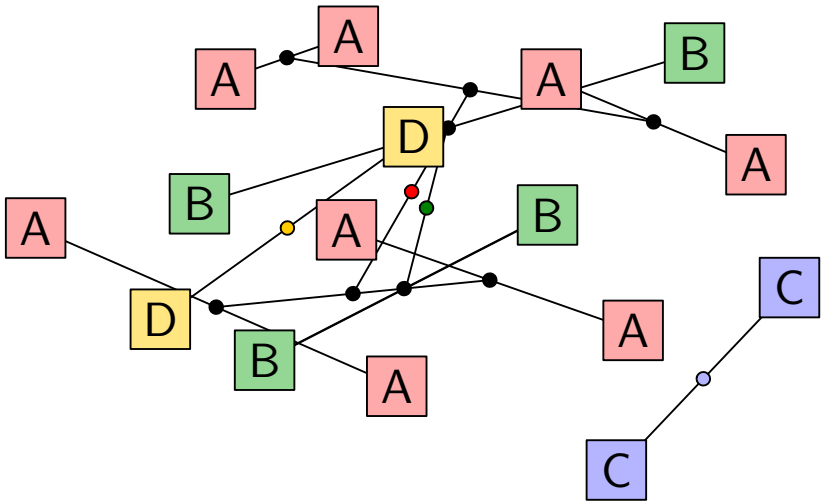


Center-of-area is the **average** of elements positions...

Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements
- Centroid as a **center-of-mass** concept, but with area weights...
- **Centroid**-based golden rules:
  - **Coincidence** of all centroids

e.g. A : B : C : D ratios
8 4 2 2

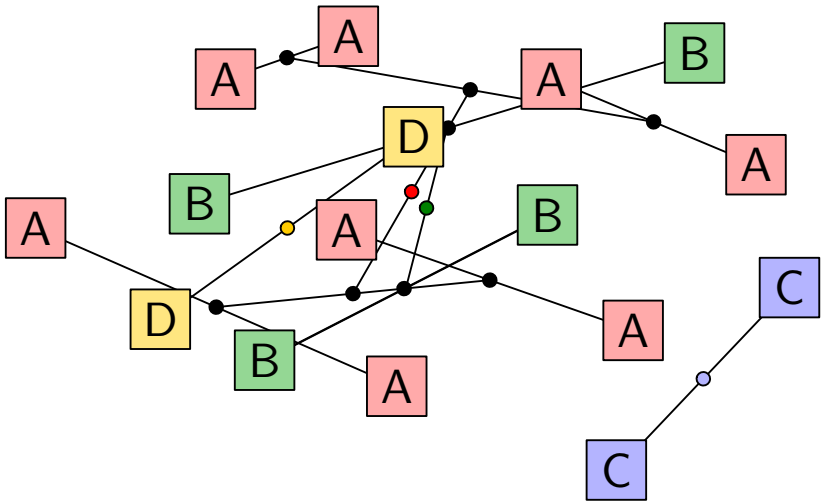


Not a common centroid arrangement!

Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements
- Centroid as a **center-of-mass** concept, but with area weights...
- **Centroid**-based golden rules:
  - **Coincidence** of all centroids
  - **Symmetry** for X and Y axes

e.g. A : B : C : D ratios
8 4 2 2



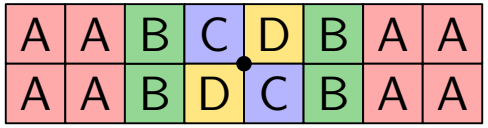
A	A	B	C	D	B	A	A
A	A	B	D	C	B	A	A

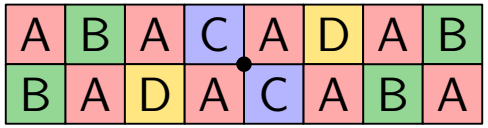
Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements

e.g. A : B : C : D ratios
8 4 2 2

- Centroid as a **center-of-mass** concept, but with area weights...
- **Centroid**-based golden rules:
  - **Coincidence** of all centroids
  - **Symmetry** for X and Y axes
  - **Dispersion** of groups as uniformly as possible

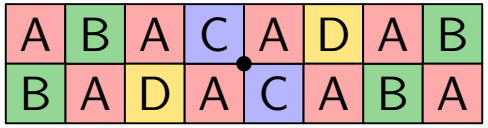


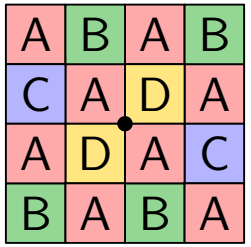


Common Centroid Arrays

- Difficult to achieve for **large** and **multiple groups** of unitary elements
- Centroid as a **center-of-mass** concept, but with area weights...
- **Centroid**-based golden rules:
  - **Coincidence** of all centroids
  - **Symmetry** for X and Y axes
  - **Dispersion** of groups as uniformly as possible
  - **Compactness** of overall array to ideal square shape

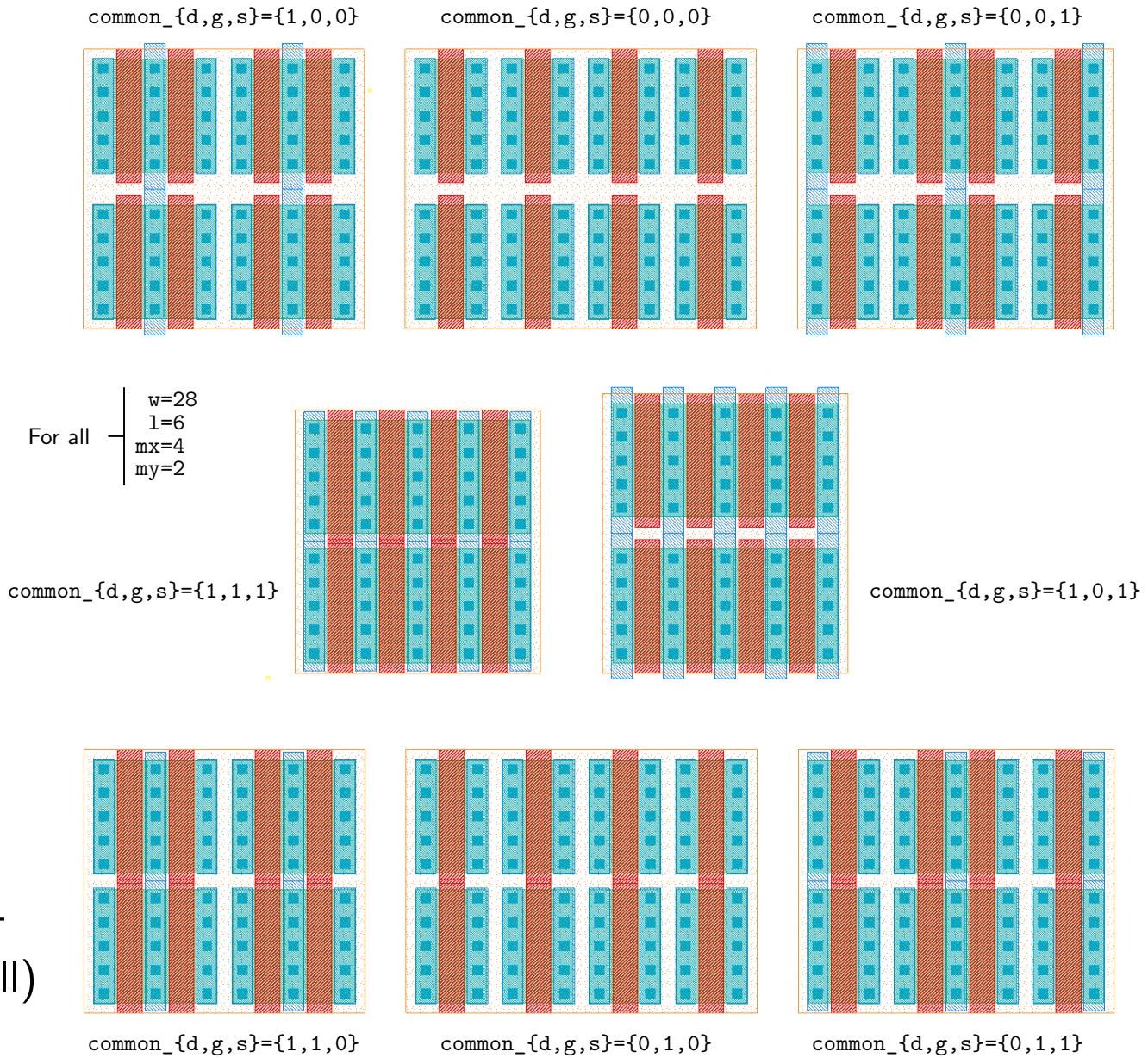
e.g. A : B : C : D ratios
8 4 2 2





PCell-Based Layout

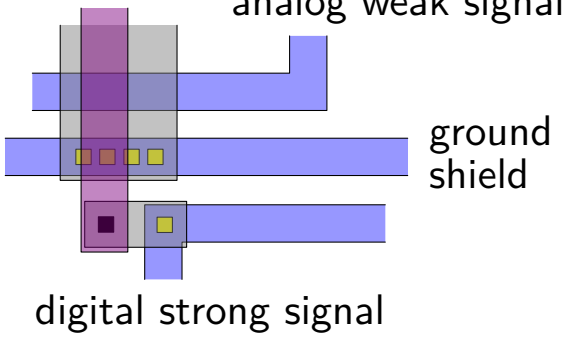
- **Unitary** elements
- **Regular** geometry
- **Design rule** compliant

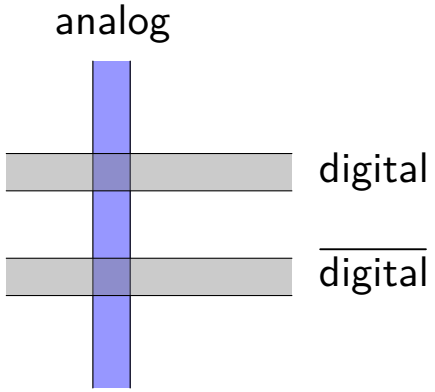


e.g. CNM25 NMOSFET parameterized cell (PCell)

Decoupling Guidelines

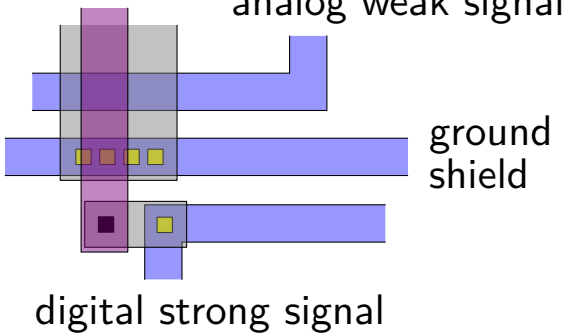
- **Signal integrity** between analog, digital, RF... domains

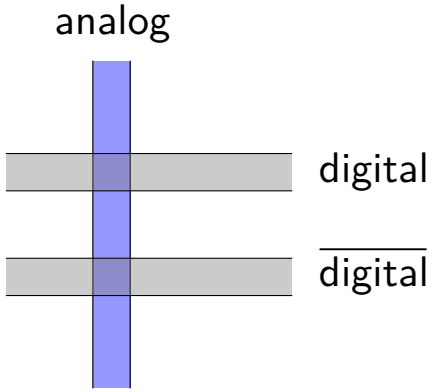




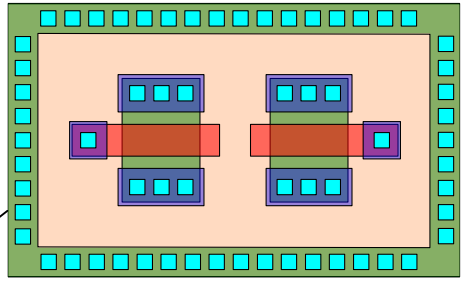
Decoupling Guidelines

- **Signal integrity** between analog, digital, RF... domains

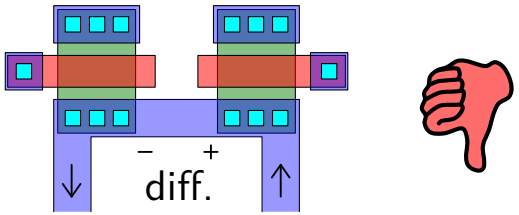


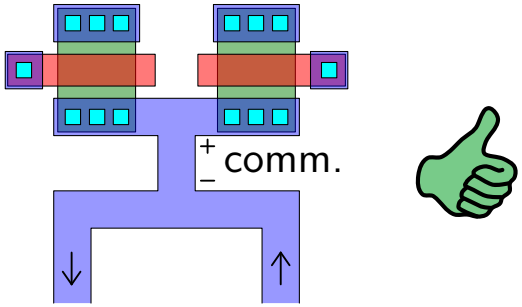


- Avoiding signal coupling through **power rails**



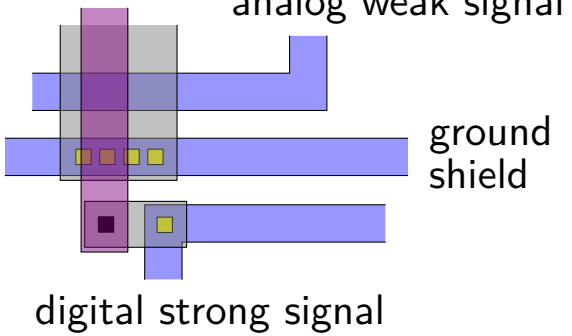
guard rings
against substrate noise

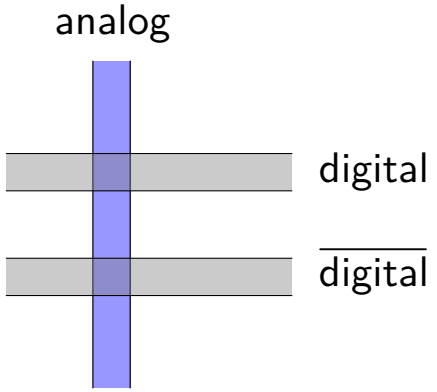




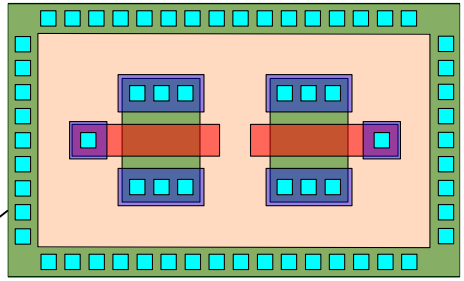
Decoupling Guidelines

- **Signal integrity** between analog, digital, RF... domains

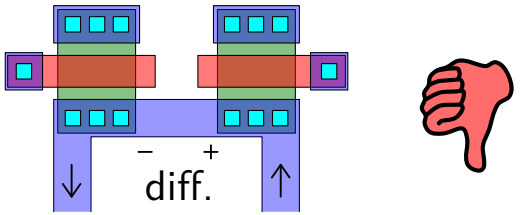




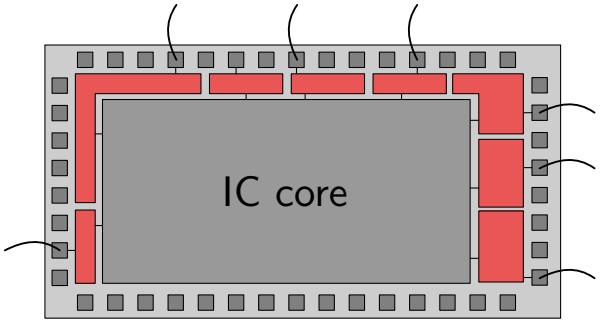
- Avoiding signal coupling through **power rails**

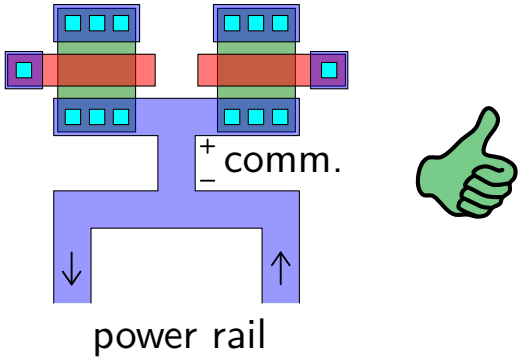


guard rings
against substrate noise



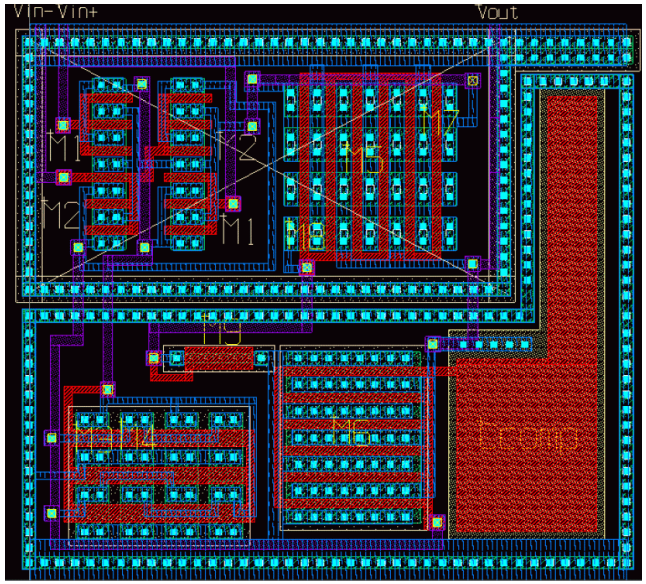
- Investing peripheral free area for on-chip **decoupling capacitors**

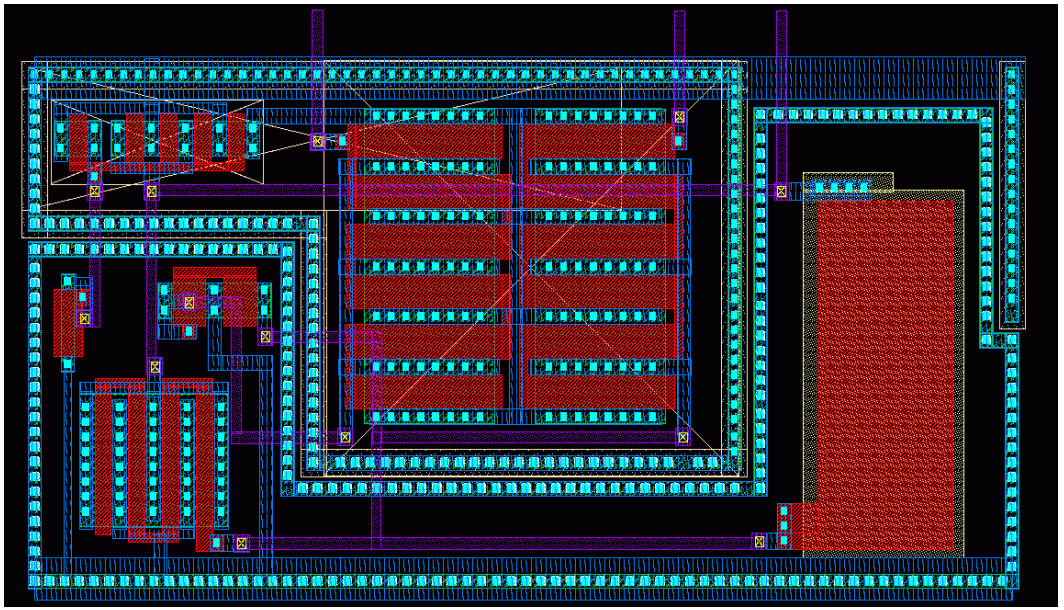


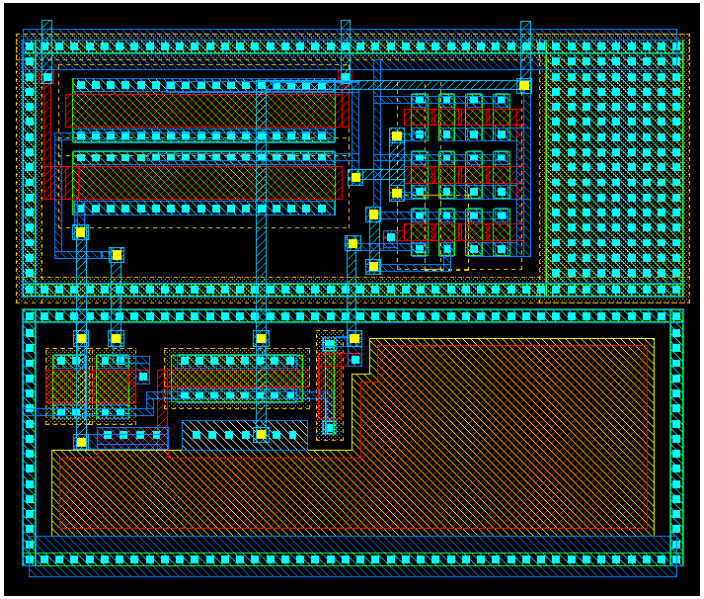


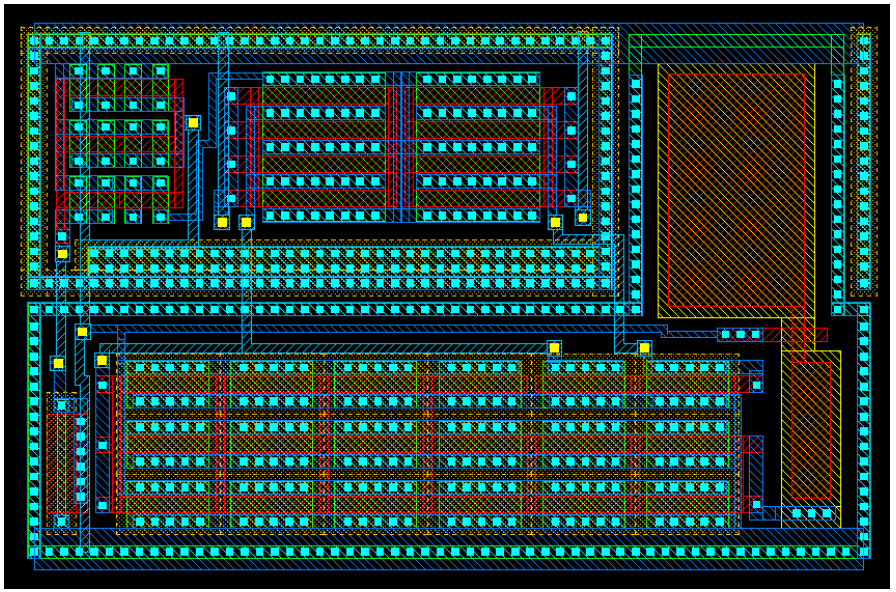
OpAmp Layout Examples

Let's evaluate other students work....



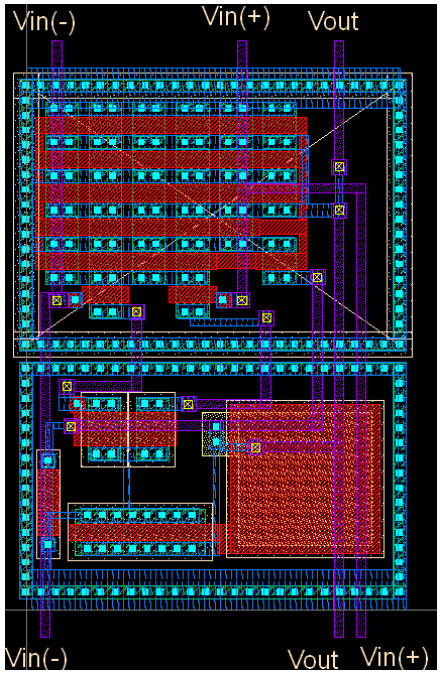


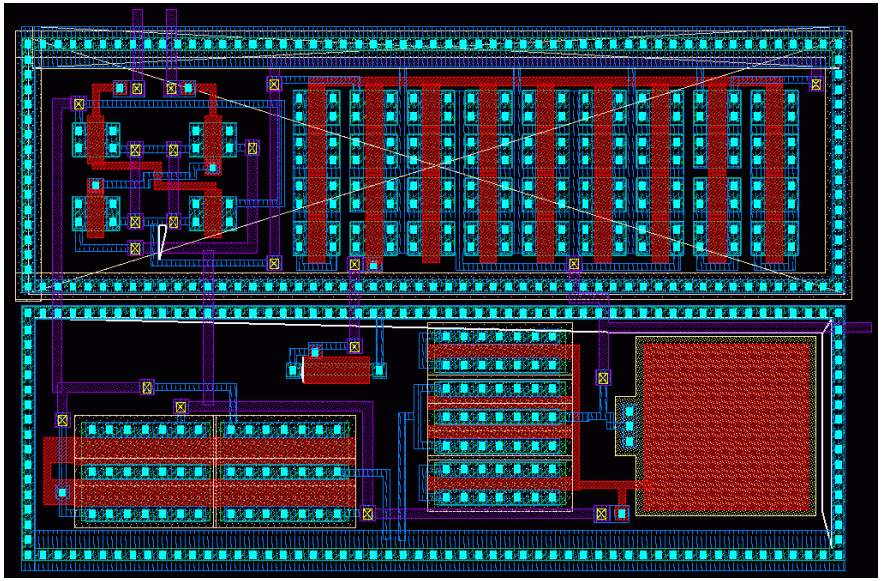


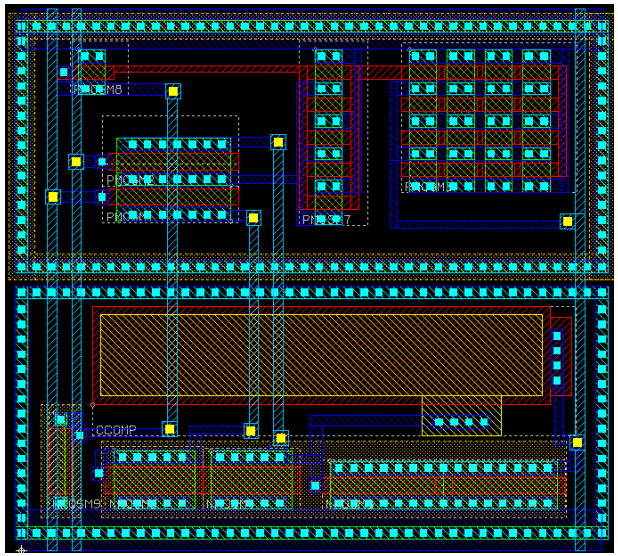


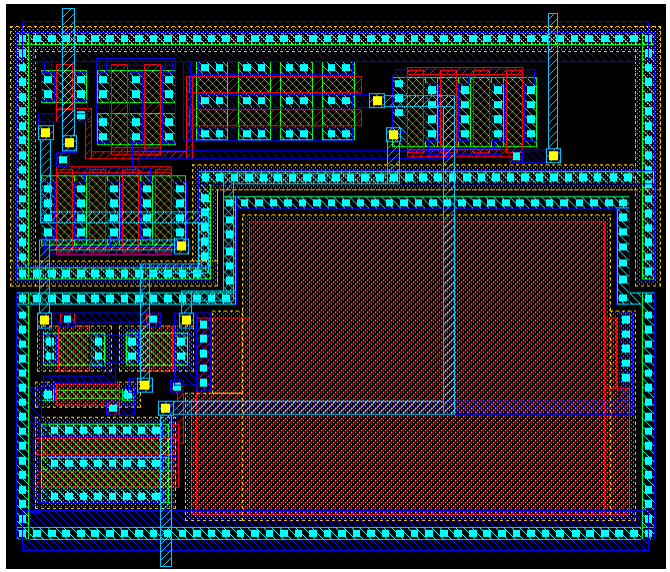
OpAmp Layout Examples

Let's evaluate other students work....





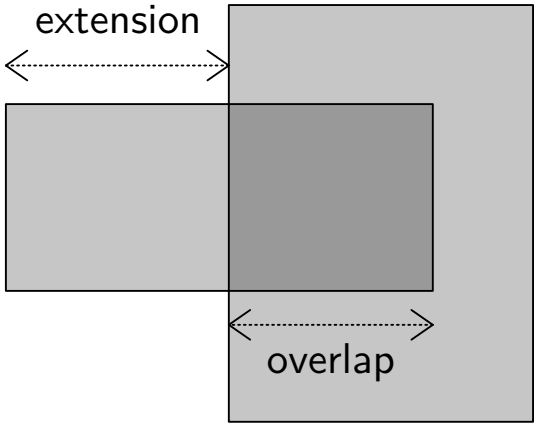




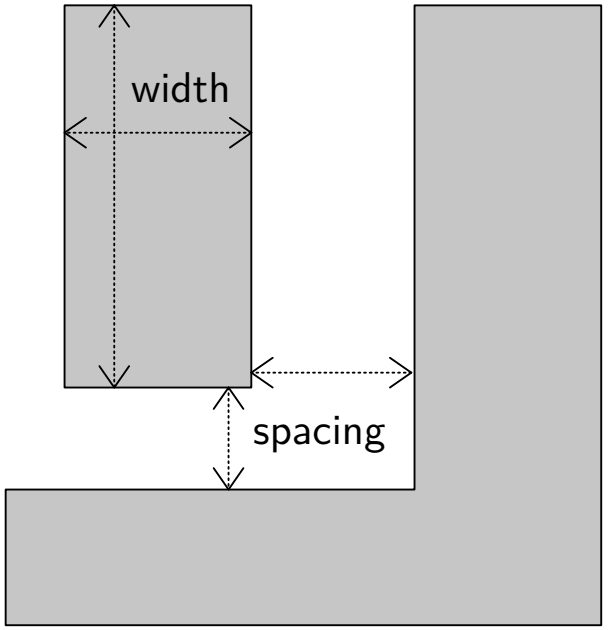
Geometrical Rules

- Basic **2D** concepts

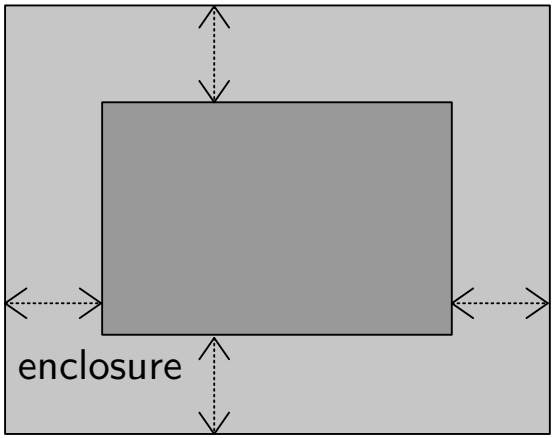
e.g. MOSFET gate

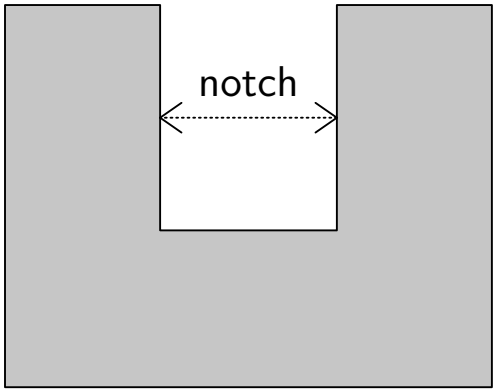


e.g. signal routing



e.g. MiM capacitor



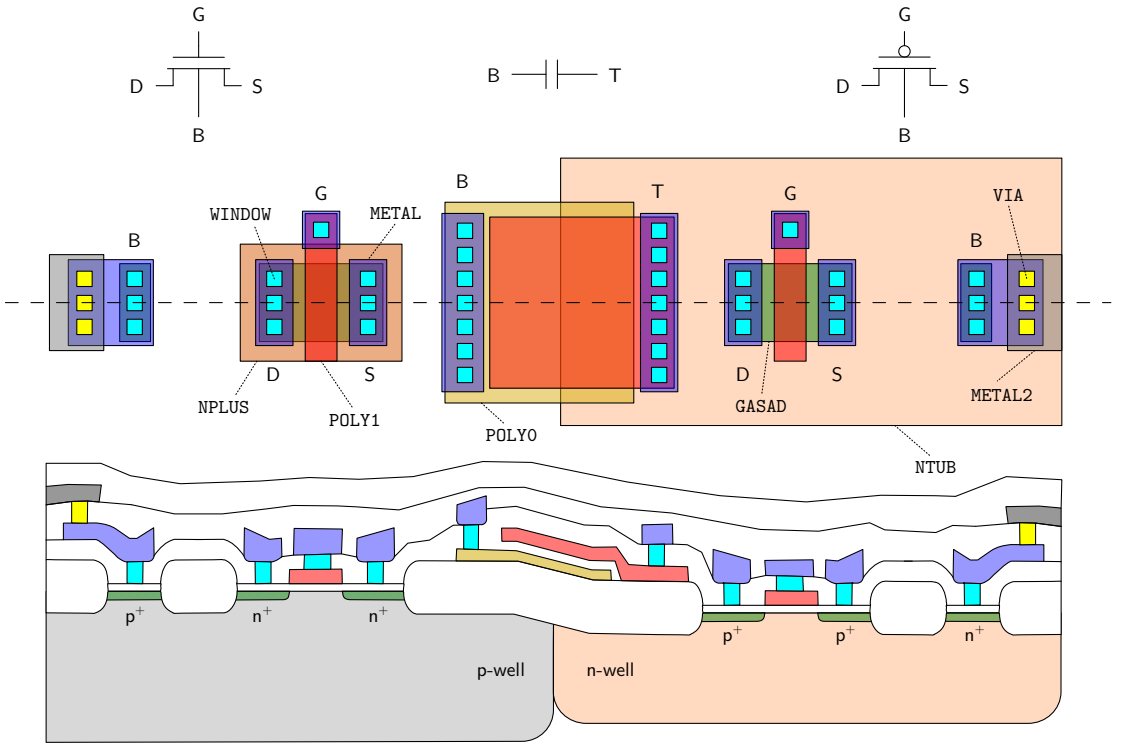


e.g. serpentine resistor

Geometrical Rules

- **Technology** rules set

e.g. 2.5um 2P2M CMOS (**CNM25**)



0.0	Design grid is 0.25um x 0.25um
1.1	N-well width >= 8um
1.2	N-well spacing and notch >= 8um
2.1	GASAD width >= 2um
2.2	GASAD spacing and notch >= 4um
2.3	N-well enclosure of P-plus active >= 5um
2.4	N-well spacing to N-plus active >= 5um
3.1	Poly0 width >= 2.5um
3.2	Poly0 spacing and notch >= 6um
3.3	Poly0 spacing to GASAD >= 6um
4.1.a	Poly1 width inside GASAD >= 3um
4.1.b	Poly1 width outside GASAD >= 2.5um
4.2	Poly1 spacing and notch >= 3um
4.3	GASAD extension of Poly1 >= 3um
4.4	Poly1 extension of GASAD >= 2.5um
4.5	Poly1 spacing to GASAD >= 1.25um
4.6	Poly0 enclosure of Poly1 >= 3um
5.1	N-plus enclosure of GASAD >= 2.5um
5.2	N-plus spacing to P-plus active >= 2.5um
5.3	N-plus spacing to Poly1 inside P-plus active >= 2um
5.4	N-plus extension of Poly1 inside N-plus active >= 1.5um
5.5	N-plus width >= 2.5um
5.6	N-plus spacing and notch >= 2.5um
6.1	Exact contact size = 2.5um x 2.5um
6.2	Contact spacing >= 3um
6.3	GASAD enclosure of Contact >= 1um
6.4	Poly1 enclosure of Contact >= 1.25um
6.5	Poly1 Contact spacing to GASAD >= 2.5um
6.6	Contact spacing to Poly1 inside GASAD >= 2um
6.9	Poly0 enclosure of Contact >= 4um
6.10	Contact spacing to Poly1 & Poly0 >= 4um
7.1	Metal1 width >= 2.5um
7.2	Metal1 spacing and notch >= 3um
7.3	Metal1 enclosure of Contact >= 1.25um
8.1	Exact via size = 3um x 3um
8.2	Via spacing >= 3.5um
8.3	Metal1 enclosure of Via >= 1.25um
8.4	Via spacing to Contact >= 2.5um
8.5	Via spacing to Poly1 >= 2.5um
9.1	Metal2 width >= 3.5um
9.2	Metal2 spacing and notch >= 3.5um
9.3	Metal2 enclosure of Via >= 1.25um
10.1	Exact passivation window size = 100um x 100um

Geometrical Rules

- **Technology** rules set

Process flow

Front end of line (**FEOL**)

Back end of line (**BEOL**)

0.0	Design grid is 0.25um x 0.25um
1.1	N-well width >= 8um
1.2	N-well spacing and notch >= 8um
2.1	GASAD width >= 2um
2.2	GASAD spacing and notch >= 4um
2.3	N-well enclosure of P-plus active >= 5um
2.4	N-well spacing to N-plus active >= 5um
3.1	Poly0 width >= 2.5um
3.2	Poly0 spacing and notch >= 6um
3.3	Poly0 spacing to GASAD >= 6um
4.1.a	Poly1 width inside GASAD >= 3um
4.1.b	Poly1 width outside GASAD >= 2.5um
4.2	Poly1 spacing and notch >= 3um
4.3	GASAD extension of Poly1 >= 3um
4.4	Poly1 extension of GASAD >= 2.5um
4.5	Poly1 spacing to GASAD >= 1.25um
4.6	Poly0 enclosure of Poly1 >= 3um
5.1	N-plus enclosure of GASAD >= 2.5um
5.2	N-plus spacing to P-plus active >= 2.5um
5.3	N-plus spacing to Poly1 inside P-plus active >= 2um
5.4	N-plus extension of Poly1 inside N-plus active >= 1.5um
5.5	N-plus width >= 2.5um
5.6	N-plus spacing and notch >= 2.5um
6.1	Exact contact size = 2.5um x 2.5um
6.2	Contact spacing >= 3um
6.3	GASAD enclosure of Contact >= 1um
6.4	Poly1 enclosure of Contact >= 1.25um
6.5	Poly1 Contact spacing to GASAD >= 2.5um
6.6	Contact spacing to Poly1 inside GASAD >= 2um
6.9	Poly0 enclosure of Contact >= 4um
6.10	Contact spacing to Poly1 & Poly0 >= 4um
7.1	Metal1 width >= 2.5um
7.2	Metal1 spacing and notch >= 3um
7.3	Metal1 enclosure of Contact >= 1.25um
8.1	Exact via size = 3um x 3um
8.2	Via spacing >= 3.5um
8.3	Metal1 enclosure of Via >= 1.25um
8.4	Via spacing to Contact >= 2.5um
8.5	Via spacing to Poly1 >= 2.5um
9.1	Metal2 width >= 3.5um
9.2	Metal2 spacing and notch >= 3.5um
9.3	Metal2 enclosure of Via >= 1.25um
10.1	Exact passivation window size = 100um x 100um

Design Rule Checker

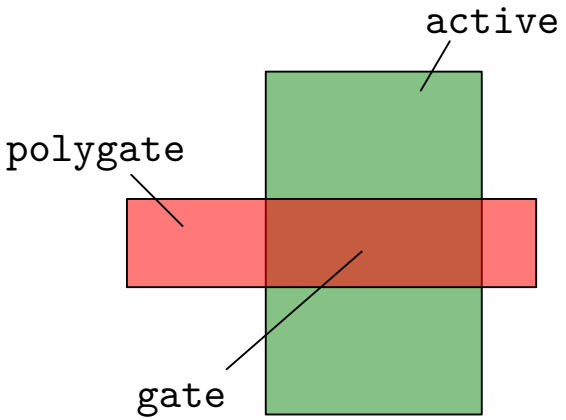
- **Programming** a rules set...

boolean operations

derived layers

cnm25drc.py

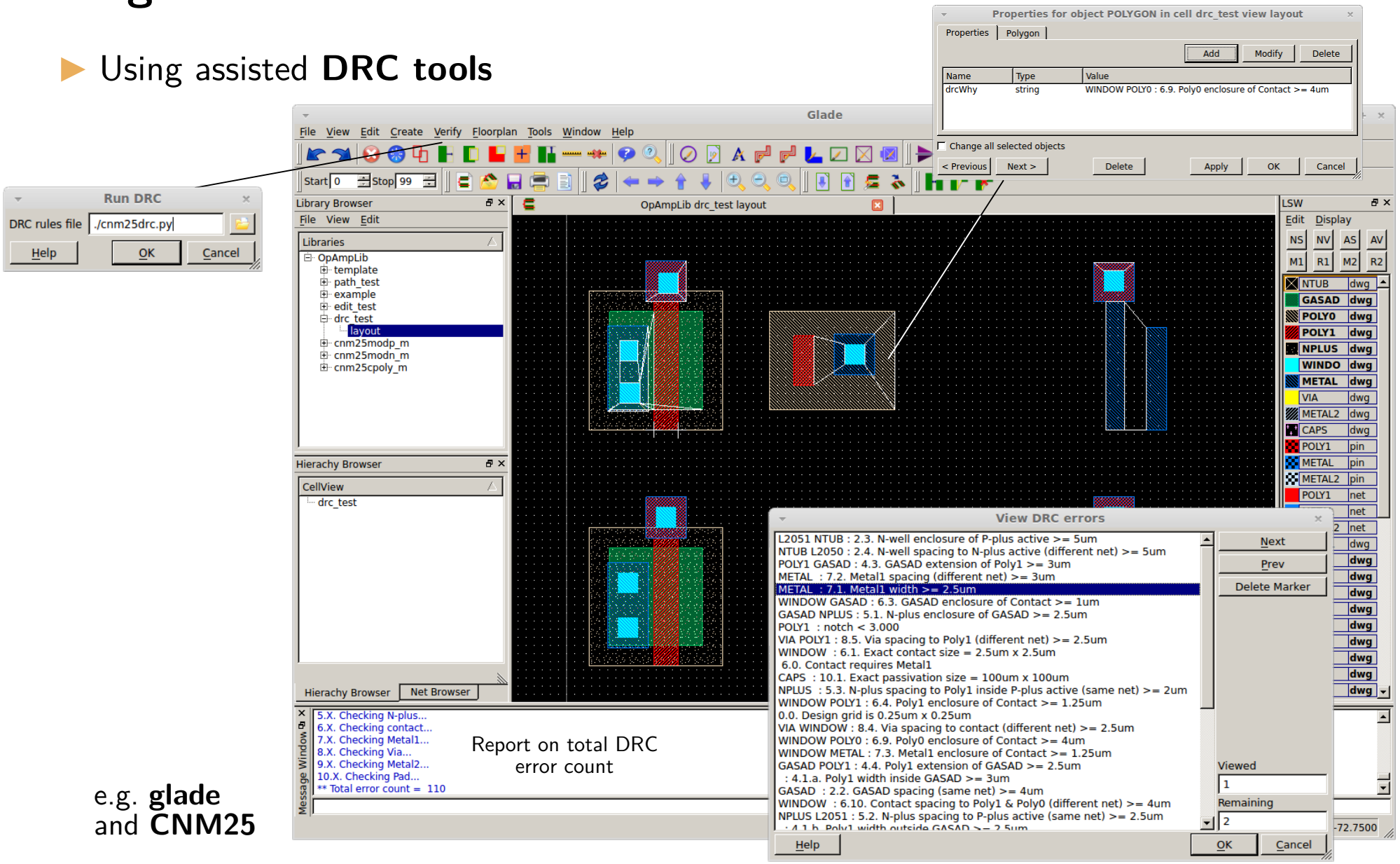
```
...
active   = geomGetShapes("GASAD", "drawing")
polygate = geomGetShapes("POLY1", "drawing")
polycap  = geomGetShapes("POLY0", "drawing")
gate     = geomAnd(polygate, active)
cpoly    = geomAnd(polygate, polycap)
geomOffGrid(polygate, 0.25, 1, "0.0. Design grid is ...
geomWidth(gate, 3, "4.1.a. Poly1 width inside GASAD >= ...
geomSpace(polygate, 3, diffnet, "4.2. Poly1 spacing...
geomNotch(polygate, 3, "4.2. Poly1 notch >= 3um")
geomExtension(polygate, active, 2.5, "4.4. Poly1 ext...
geomEnclose(polycap, cpoly, 3, "4.6. Poly0 enclosure...
...
```



e.g. 2.5um 2P2M CMOS (**CNM25**)

Design Rule Checker

- Using assisted **DRC tools**

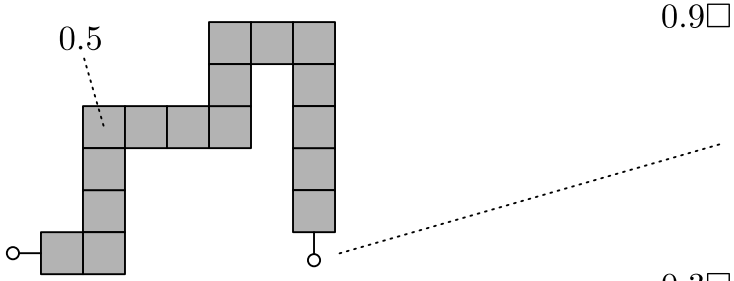


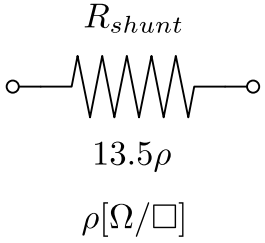
e.g. **glade** and **CNM25**

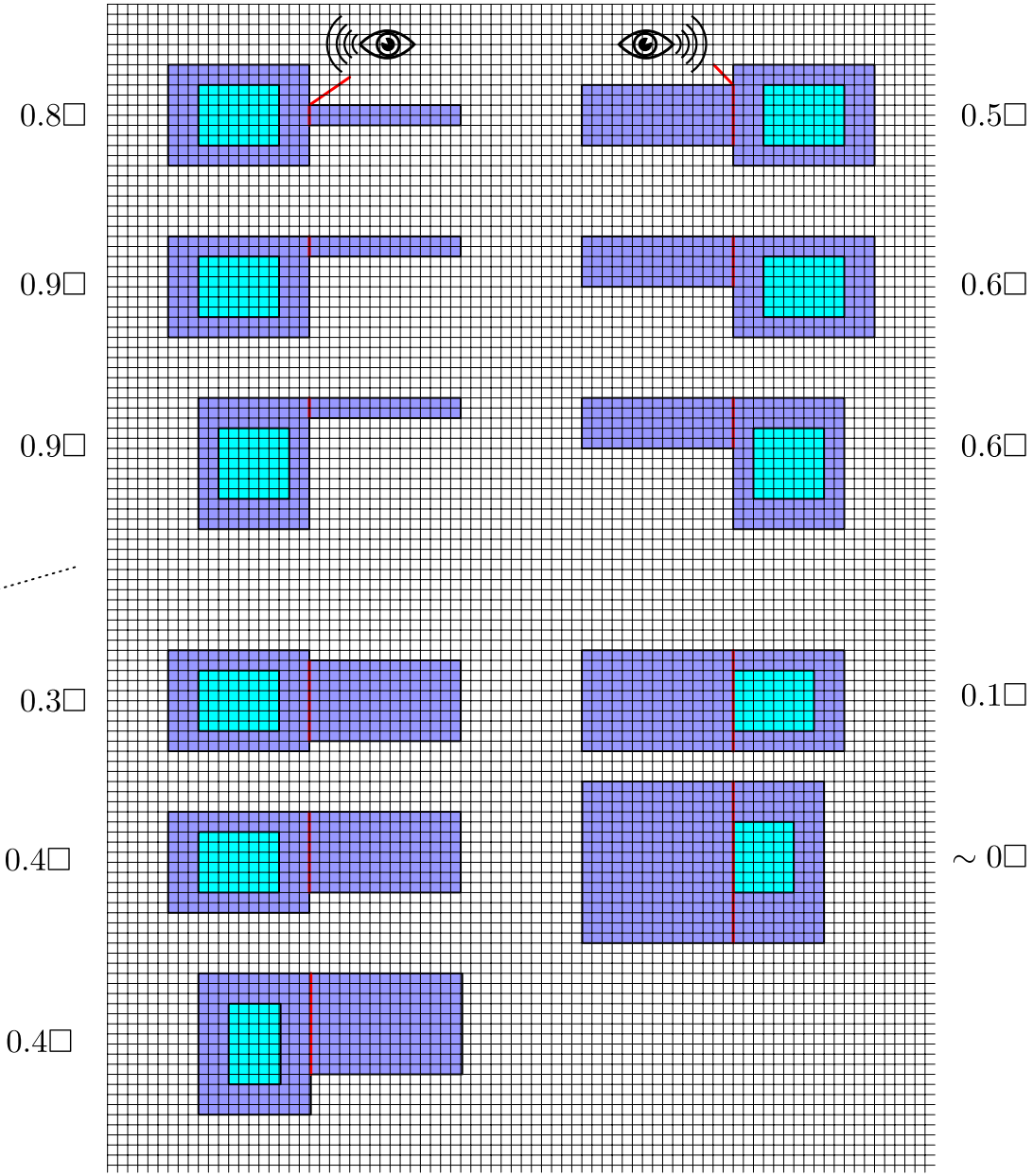
Motivation

- **Planar** technology parasitics

(2D layout view)



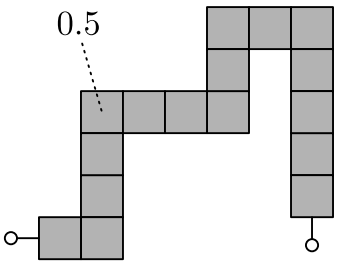




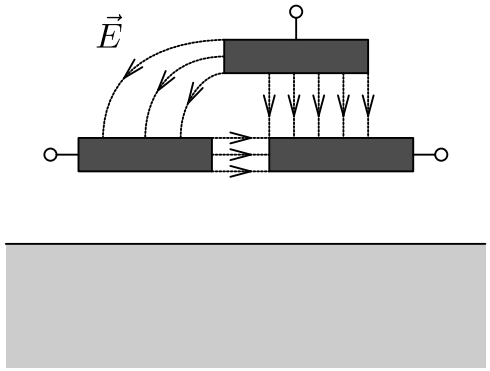
Motivation

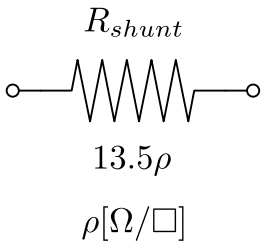
- **Planar** technology parasitics

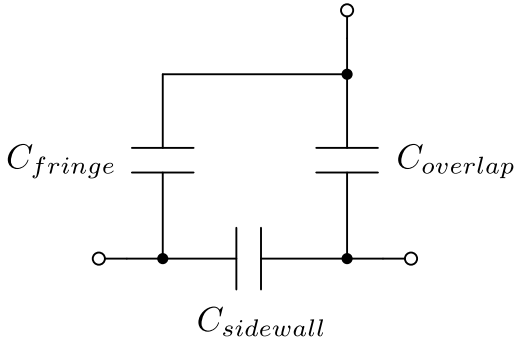
(2D layout view)



(2D cross section view)



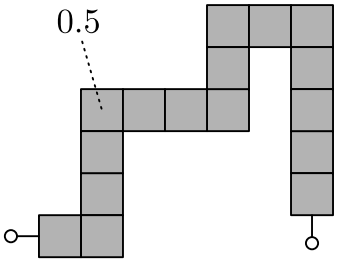




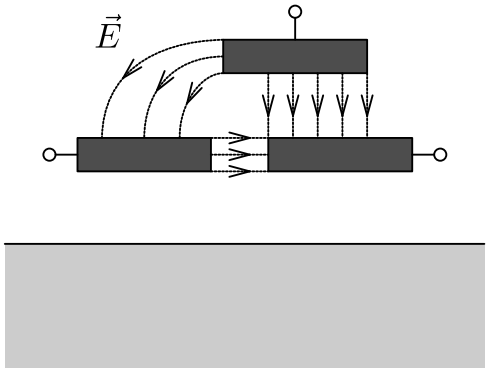
Motivation

- **Planar** technology parasitics
- **R** and **L** require node **splitting**!

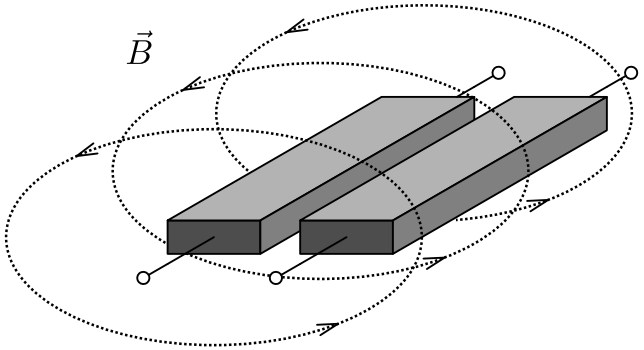
(2D layout view)

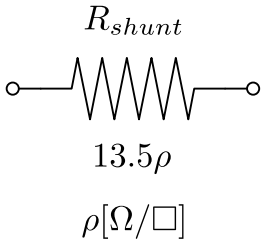


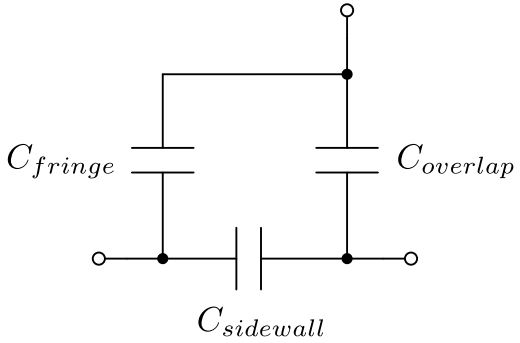
(2D cross section view)

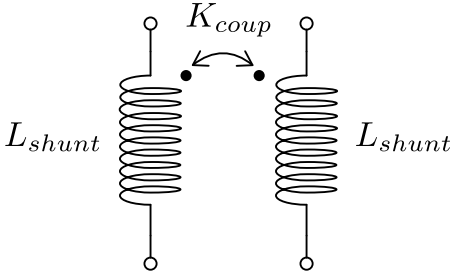


(3D view)









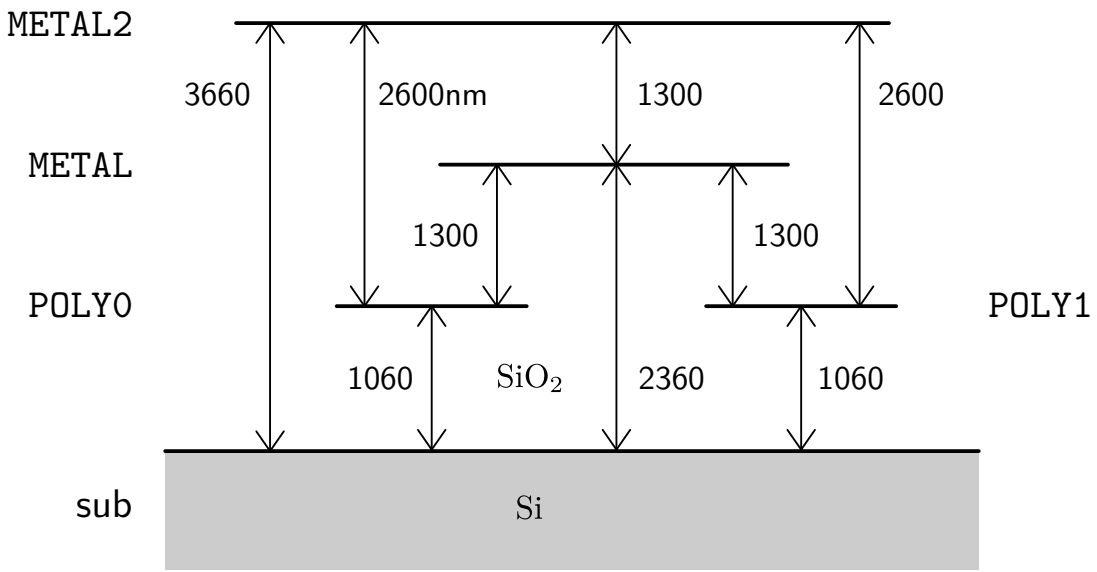
Extraction Tools

- **Programming** a simple rules set for parasitic overlap capacitance only...

cnm25xtr.py

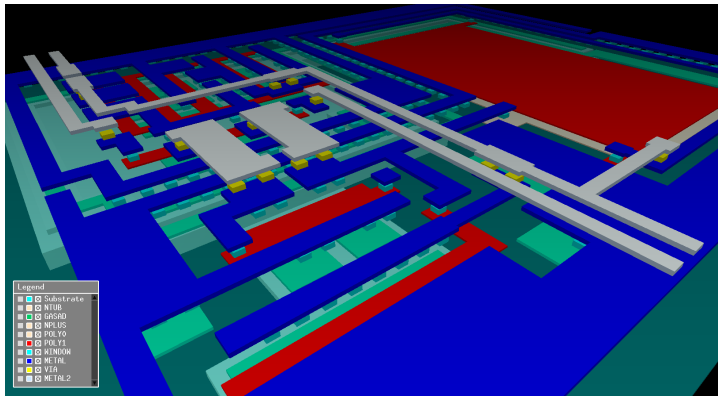
```
...
geomLabel(polygate, "POLY1", "pin", 1)
geomLabel(polygate, "POLY1", "net", 0)
geomConnect([
  [cont, ndiff, pdiff, polygate, polycap, metal1],
  [via12, metal1, metal2]... ] )
extractMOS("cnm25modn", ngate, polygate, ndiff, pwell)
extractParasitic3(pdiff, metal2, cmetal2diff, 0,
  [metal1, polygate, polycap])
...
```

e.g. 2.5um 2P2M CMOS (**CNM25**)



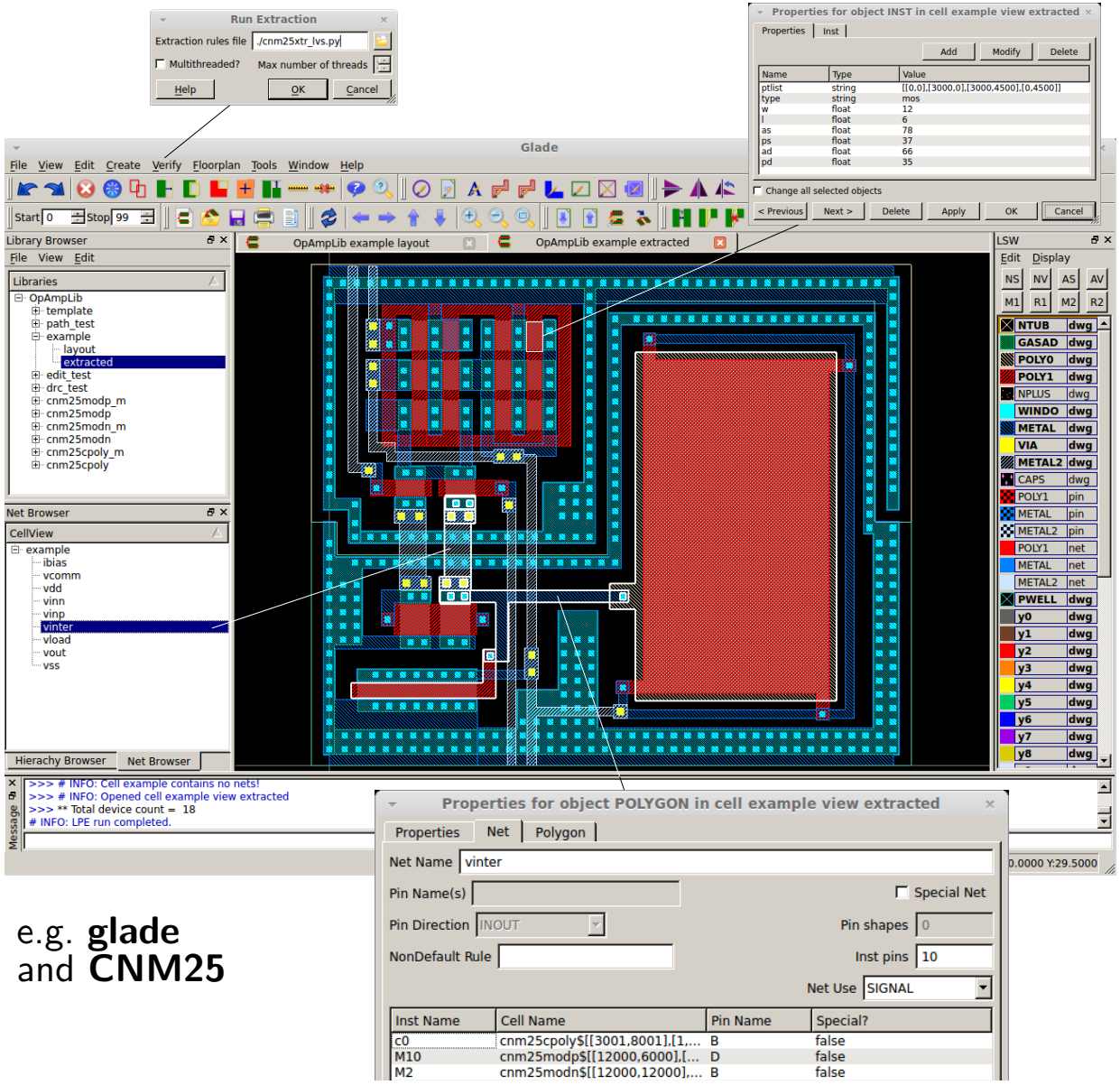
Extraction Tools

- **Programming** a simple rules set for parasitic overlap capacitance only...
- Using assisted **extraction tools**...



opamp_par.sub

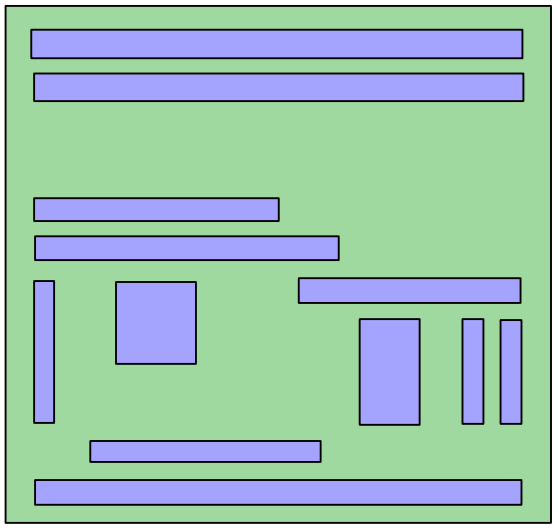
```
.SUBCKT opamp vinn vinp vout vdd vss ibias
MM0 vdd ibias vdd vdd cnm25modp w=1.2e-05 l=6e-06 as=...
MM1 vdd ibias vout vdd cnm25modp w=1.2e-05 l=6e-06 as=...
Cc0 vinter vout cnm25cpoly w=6.42928e-05 l=0.000156207
MM8 vout ibias vdd vdd cnm25modp w=1.2e-05 l=6e-06 as=...
...
CP1 vinter vss  C=3.8582e-13
CP2 vout ibias  C=3.33692e-15
CP3 vinp vss  C=1.85938e-15
CP4 vout vcomm  C=2.0918e-15
...
.ENDS
```



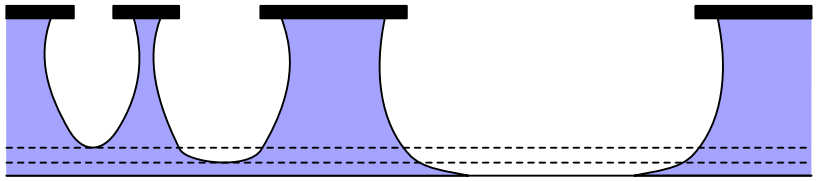
e.g. **glade** and **CNM25**

Going Into Production

- From few IC **prototypes** to small and medium **series**
- Design for manufacturing (**DFM**) layout rules to increase manufacturing **yield**
- Some **examples**:
  - Dummy **filling**

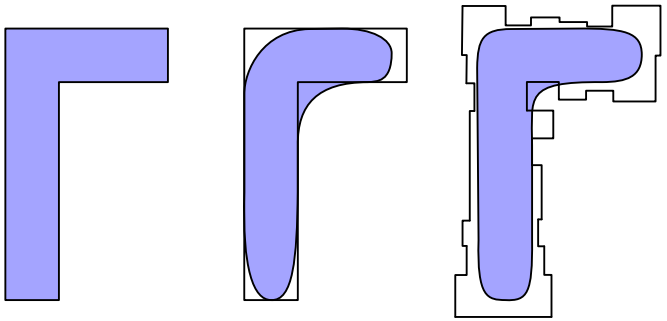


non-uniform layer area = **etching** issues

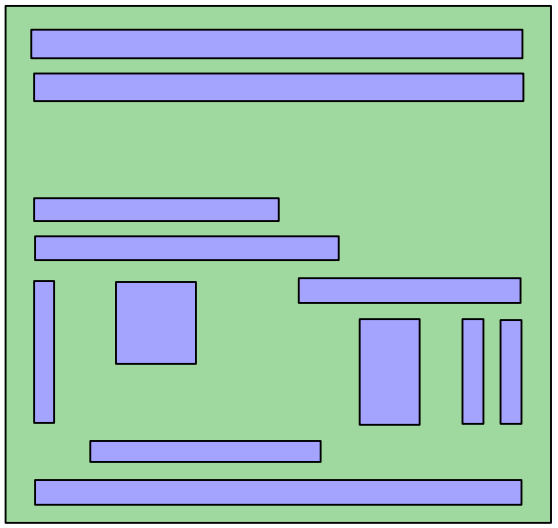


Going Into Production

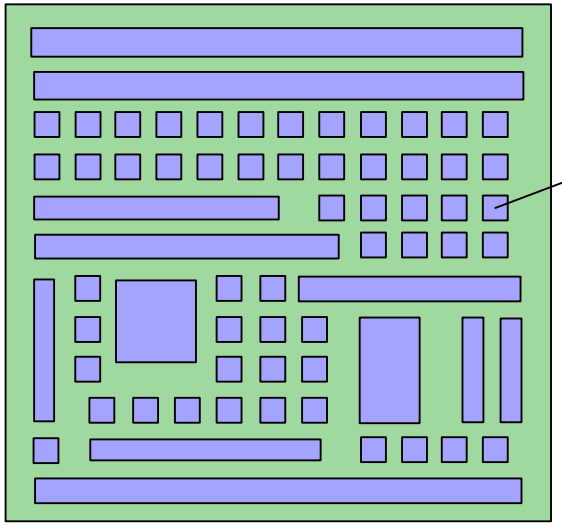
- From few IC **prototypes** to small and medium **series**
- Design for manufacturing (**DFM**) layout rules to increase manufacturing **yield**
- Some **examples**:
  - Dummy **filling**



Using few shapes also simplifies optical proximity correction (**OPC**)



non-uniform layer area = **etching** issues



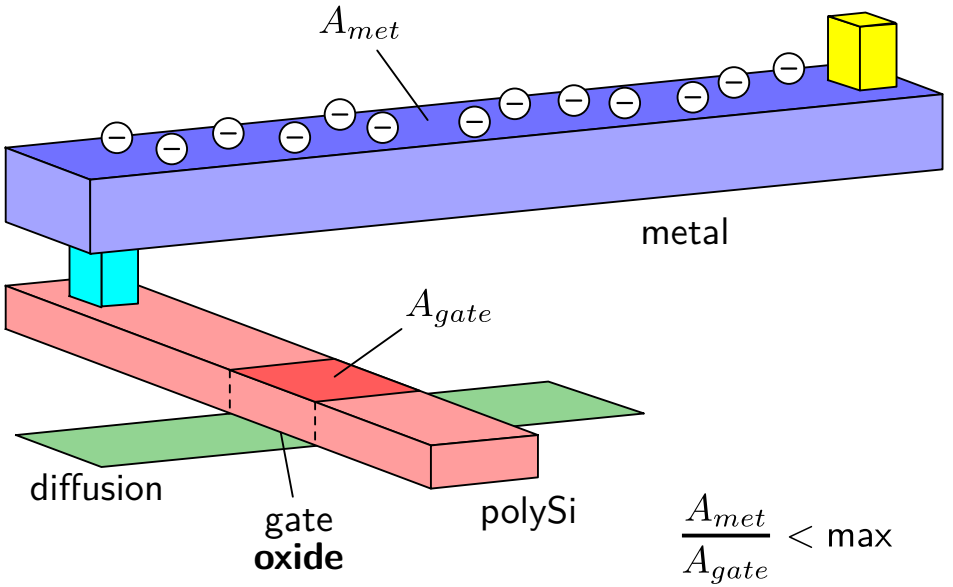
floating **dummy** structures

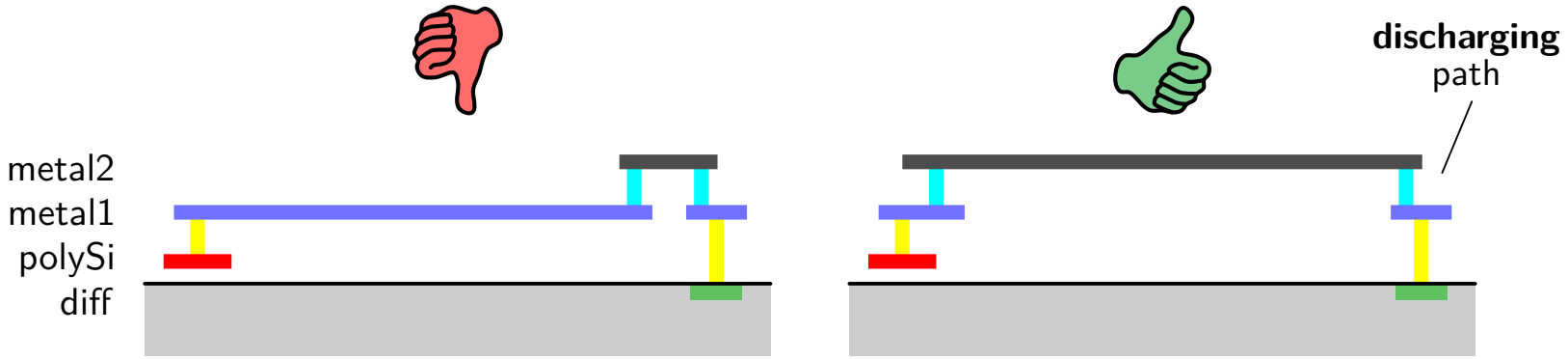
automatic filling for a target **% coverage**

Going Into Production

- From few IC **prototypes** to small and medium **series**
- Design for manufacturing (**DFM**) layout rules to increase manufacturing **yield**
- Some **examples**:
  - Dummy **filling**
  - **Antenna** reduction

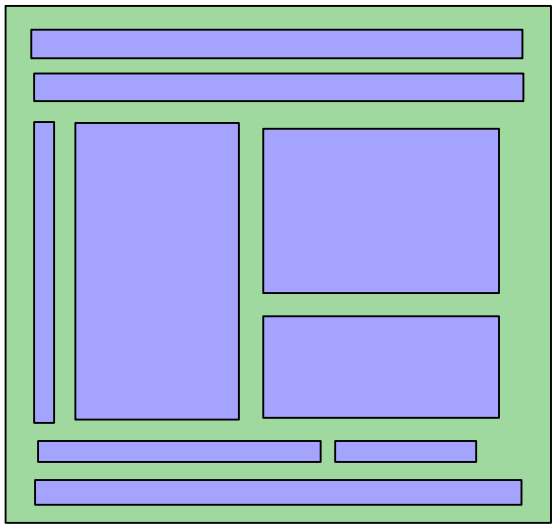
charge stored during metal patterning can **break** thin gate oxide:



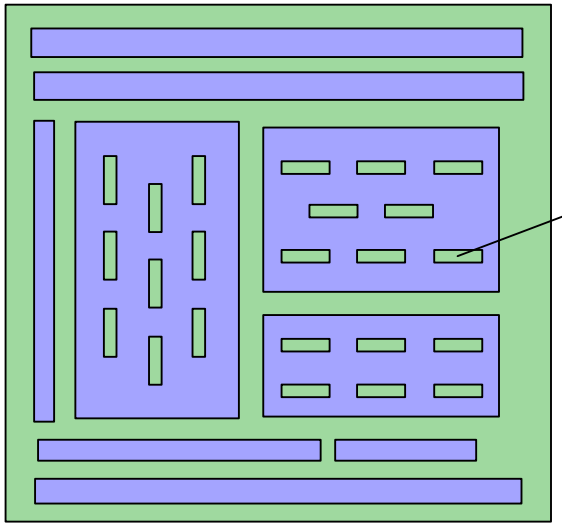


Going Into Production

- From few IC **prototypes** to small and medium **series**
- Design for manufacturing (**DFM**) layout rules to increase manufacturing **yield**
- Some **examples**:
  - Dummy **filling**
  - **Antenna** reduction
  - Metal **slotting**
  - **Multiple** contacts
  - Extra **guard rings**



large metal area = **mechanical** stress issues



stress-relief **holes**