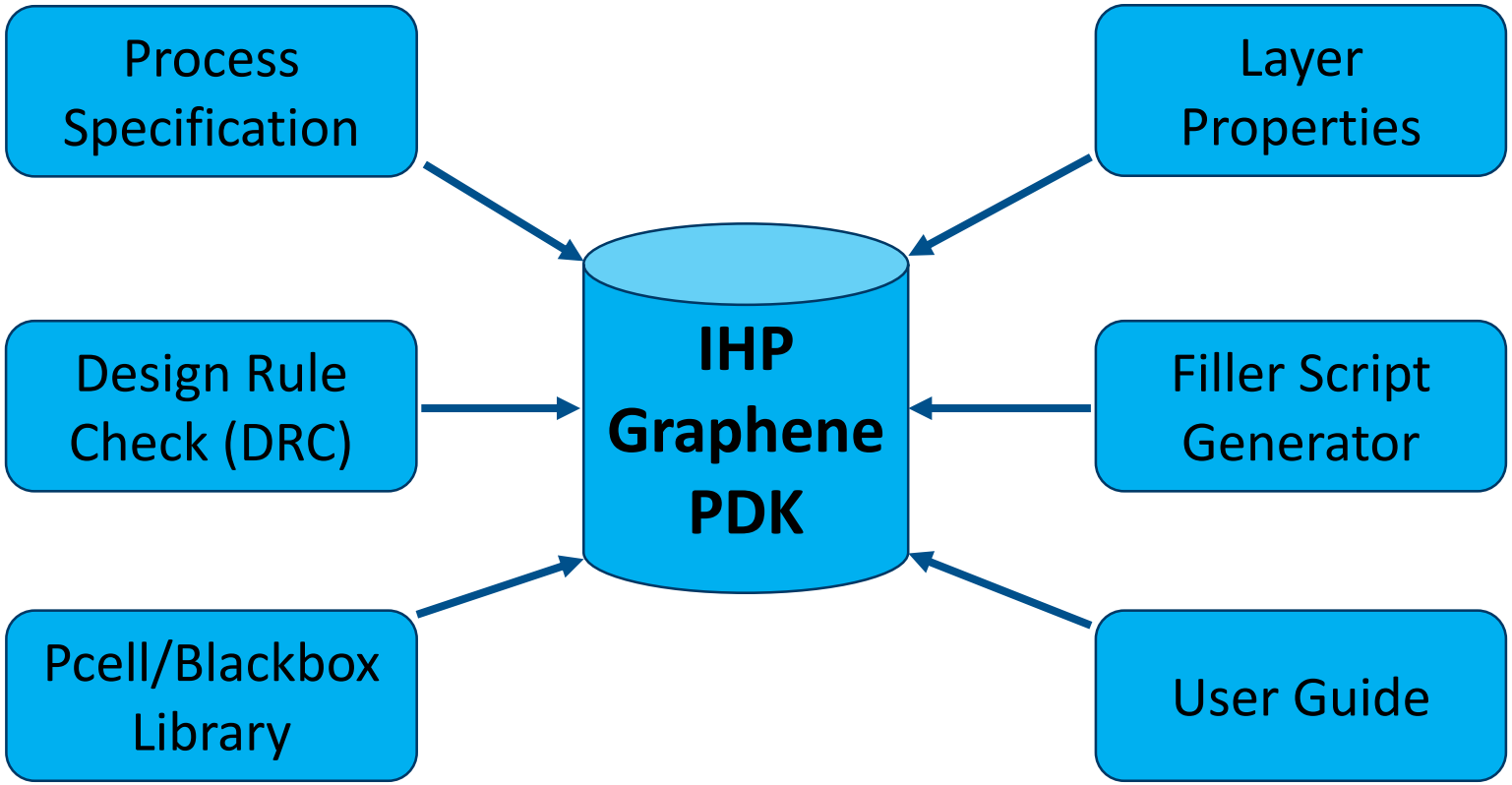
Graphene Phtonic Device Design with IHP Graphene Process Design Kit (PDK)

Ashraful Islam Raju, Daniele Capista, Mindaugas Lukosius

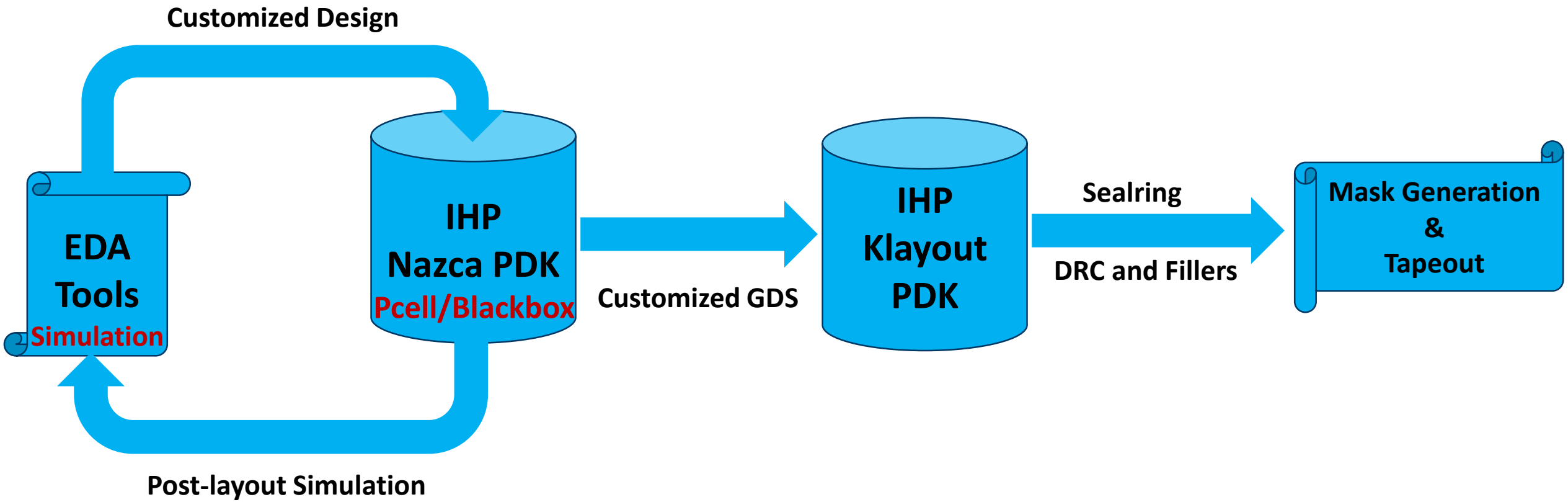
04.11.2026

Leibniz Institute
for high
performance
microelectronics

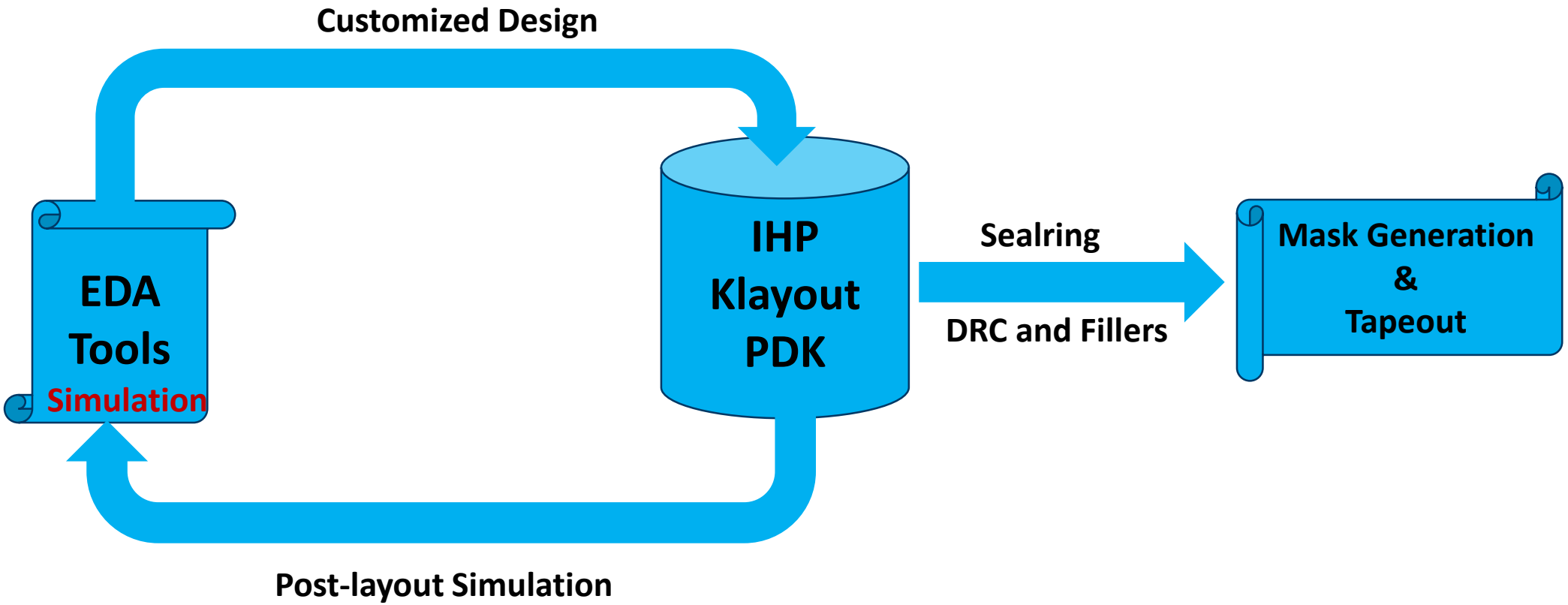
IHP Graphene PDK Overview



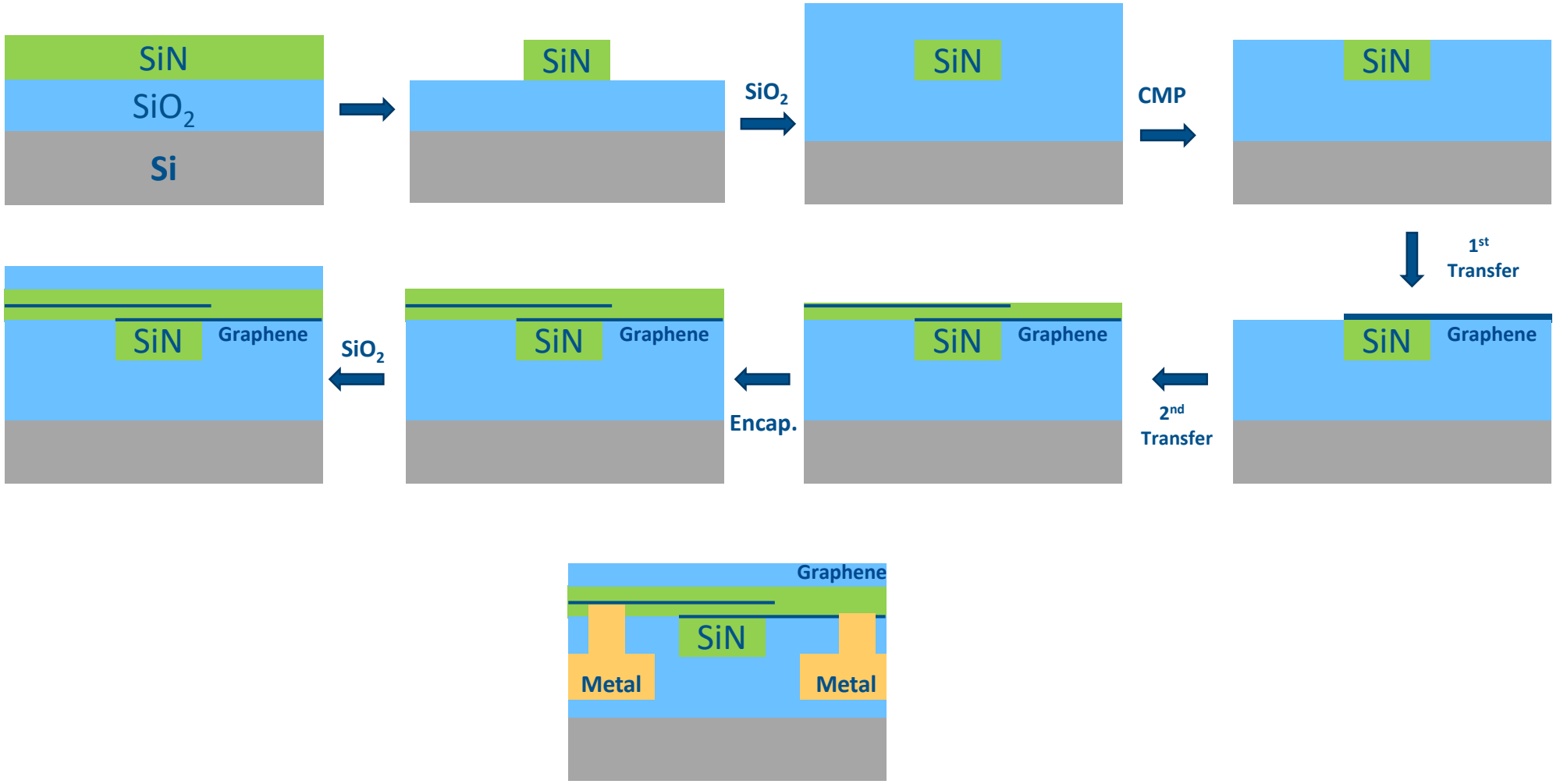
PDK Flow 1


Customized Design
EDA
Tools
Simulation
IHP
Nazca PDK
Pcell/Blackbox
Customized GDS
IHP
Klayout
PDK
Sealring
DRC and Fillers
Mask Generation
&
Tapeout
Post-layout Simulation

PDK Flow 2


Customized Design
EDA
Tools
Simulation
IHP
Klayout
PDK
Sealring
DRC and Fillers
Mask Generation
&
Tapeout
Post-layout Simulation

Process Flow

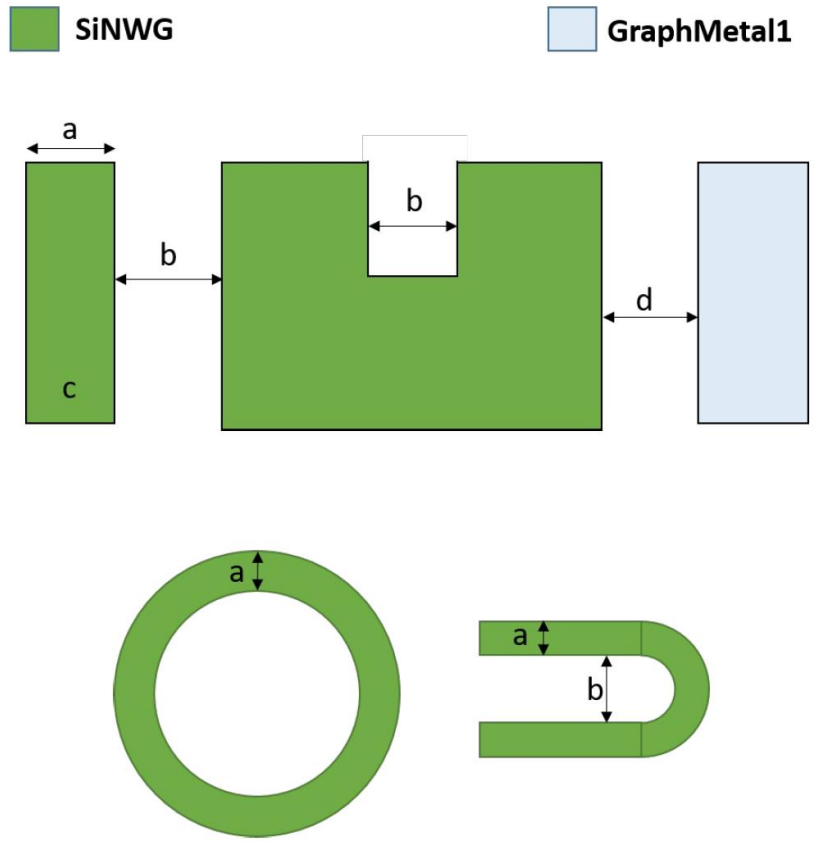


Layer Properties

Layer Name	GDSII No	Mask	Description
GraphBot	78	GRB	patterning 1st Graphene layer
GraphTop	79	GRT	patterning 2nd Graphene layer
GraphGat	118	GRG	patterning Graphene GFET Gate
SiNWG	119	SNW	patterning silicon nitride waveguide
GraphCont	85	GCT	metal contacts to GraphBot, GraphTop, GraphGat
GraphVia	256	GCV	metal contacts to GraphBot, GraphTop, GraphGat
SiNGrating	88	GNC	patterning Silicon Nitride Grating
GraphMetal1	109	GM1	Graphene-metal standard interconnect
GraphMet1L	110	GML	Graphene-metal lift-off interconnect
GraphPass	89	GPS	regions where passivation is needed
GraphPad	97	GPD	regions where passivation open

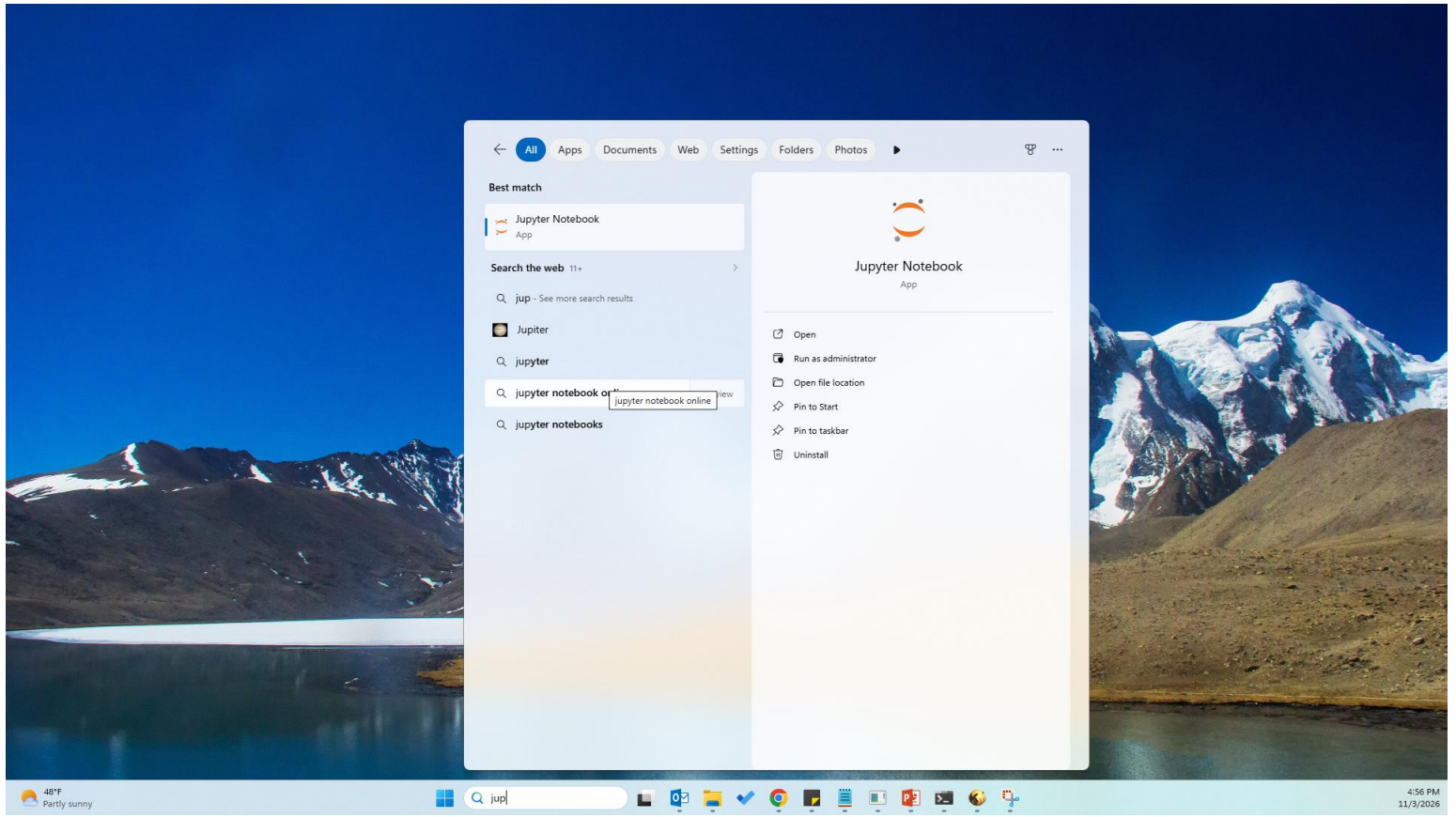
Design Rules

Rule #	Description	Value [µm]
SNW.a	Miminum SiNWG width	0.700
SNW.b	Miminum SiNWG space or notch	0.300
SNW.c	Miminum SiNWG area	5.00
SNW.d	Miminum SiNWG distance to GraphMetal1	1.00



Graphene Electro-Absorption Modulator Design

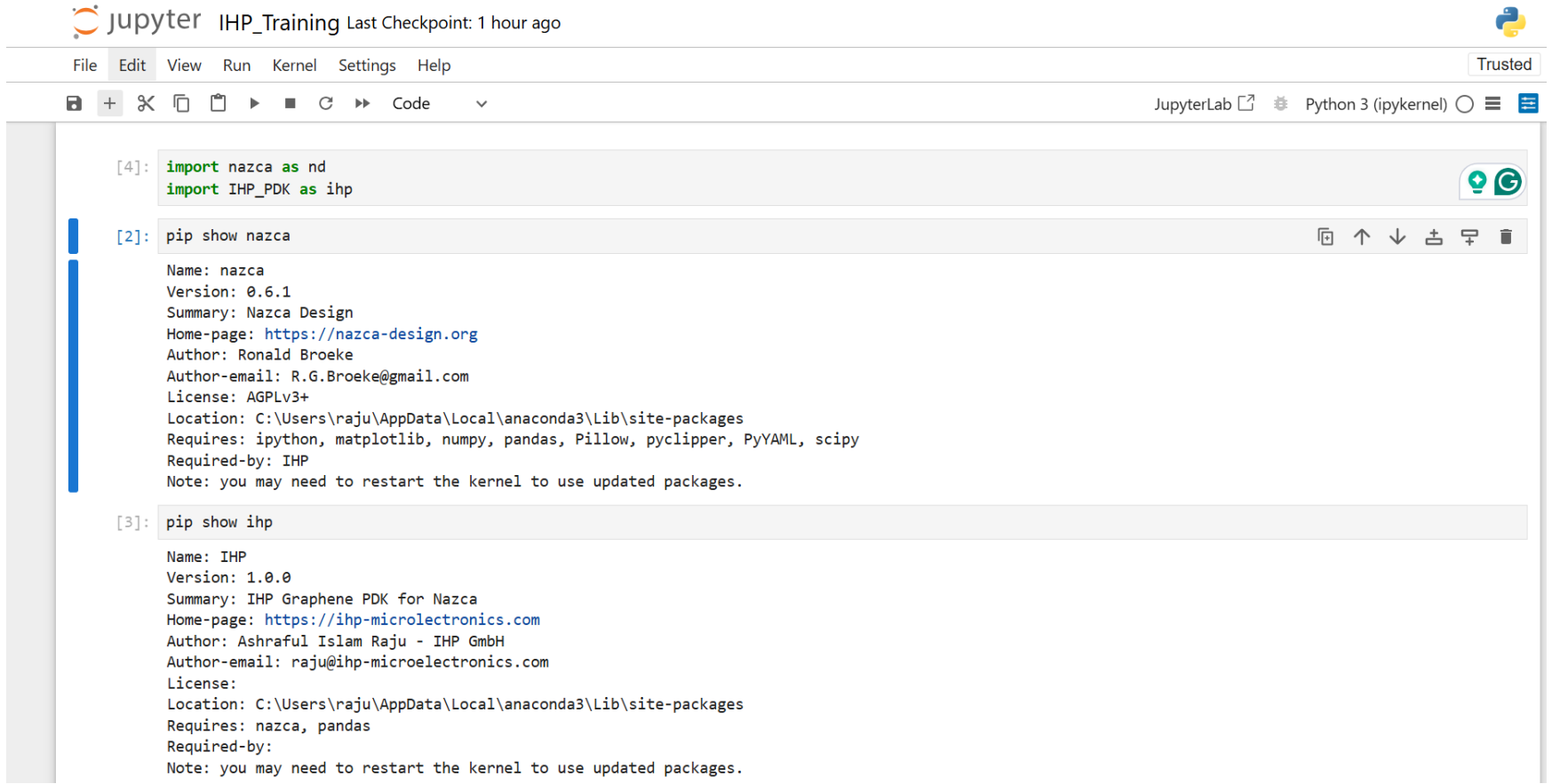
Step 0: Open Jupyther Notebook >



Importing PDK and Nazca

Step 1: Import libraries and PDK

- *Import nazca as nd*
- *Import IHP_PDK as ihp*
- *pip show nazca*
- *pip show ihp*



Example: Electro-absorption Modulator (EAM)

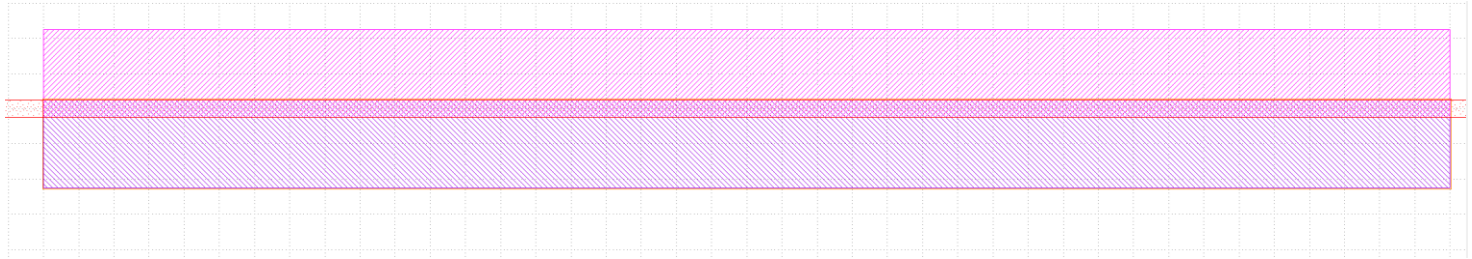
Step 2: Constructing EAM

- *wg_str = ihp.ic.strt(length=100, width=1).put()*
- *grt_str = ihp.ic_grt.strt(length=80, width=5,).put(100/2 - 80/2, -1/2 + 5/2)*
- *grb_str = ihp.ic_grb.strt(length=80, width=5,).put(100/2 - 80/2, 1/2 - 5/2)*
- gps_str = ihp.ic_gps.strt(length=80+0.1, width=5+0.1).put(100/2 - 80/2- 0.1/2, 1/2 - 5/2)
- nd.export_gds(filename='EAM.gds')

Layer Name	GDSII No	Mask
GraphBot	78	GRB
GraphTop	79	GRT
GraphGat	118	GRG
SiNWG	119	SNW
GraphCont	85	GCT
GraphVia	256	GCV
SiNGrating	88	GNC
GraphMetal1	109	GM1
GraphMet1L	110	GML
GraphPass	89	GPS
GraphPad	97	GPD

Klayout

- nd.export_gds(filename='EAM.gds')

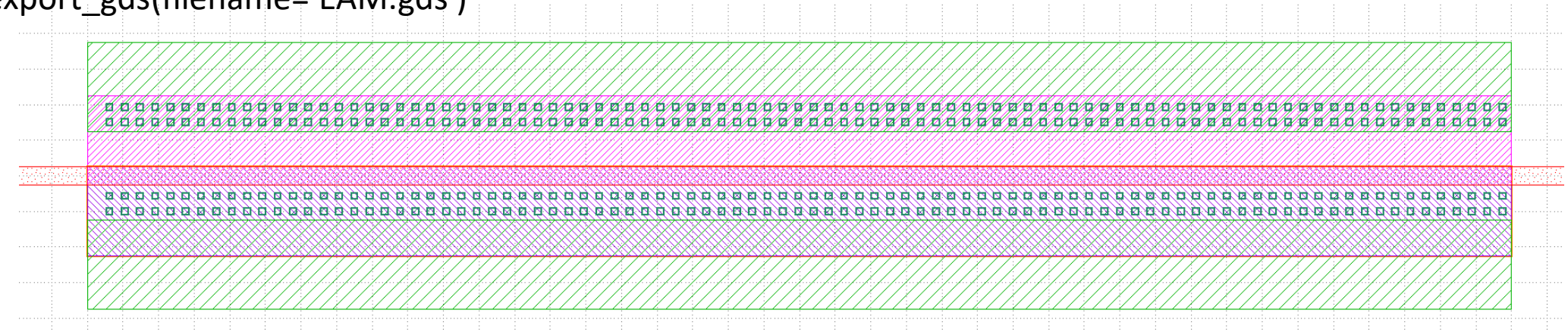


Example: Electro-absorption Modulator (EAM)

Layer Name	GDSII No	Mask
GraphBot	78	GRB
GraphTop	79	GRT
GraphGat	118	GRG
SiNWG	119	SNW
GraphCont	85	GCT
GraphVia	256	GCV
SiNGrating	88	GNC
GraphMetal1	109	GM1
GraphMet1L	110	GML
GraphPass	89	GPS
GraphPad	97	GPD

Step 3: Constructing EAM contact

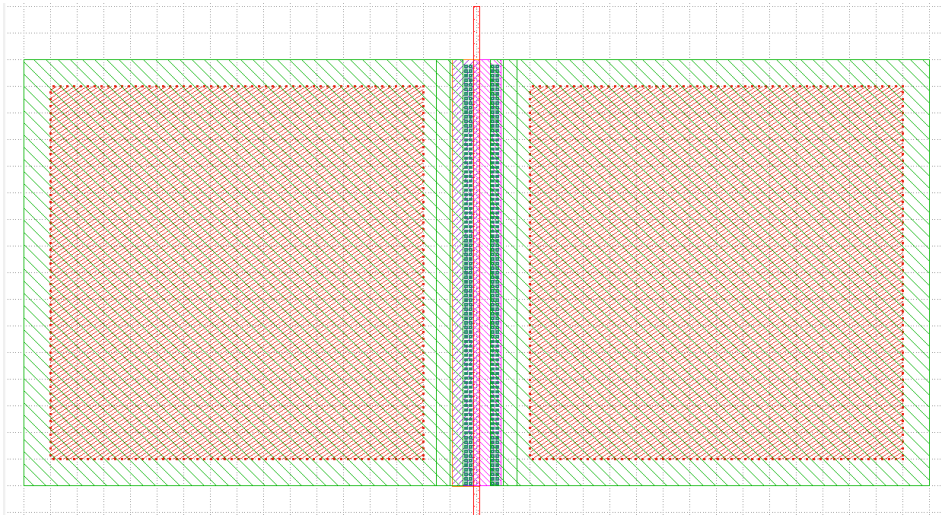
- *gm1_str_top = ihp.ic_gm1.strt(length=80, width=5,).put(100/2 - 80/2, -1/2 + 5/2 + 3)*
- *gm1_str_bot = ihp.ic_gm1.strt(length=80, width=5,).put(100/2 - 80/2, 1/2 - 5/2 - 3)*
- *contact_top = ihp.contact_array(0.36, 0.5, 2.5, 80).put(gm1_str_top.pin['a0'].move(-80/2, 5/2,270), flop=False)*
- *contact_bot = ihp.contact_array(0.36, 0.5, 2.5, 80).put(gm1_str_bot.pin['a0'].move(-80/2, -5/2,270), flop=False)*
- nd.export_gds(filename='EAM.gds')



Example: Electro-absorption Modulator (EAM)

Step 4: Constructing EAM Top Pad

- *toppad_gm1 = ihp.ic_gm1.strt(length=80, width=80,).put(gm1_str_top.pin['b0'].move(0, 80/2, 0), flop=True)*
- *toppad_gpd = ihp.ic_gpd.strt(length=70, width=70, arrow=False).put(gm1_str_bot.pin['b0'].move(-5, 80/2+10, 0), flop=True)*
- *botpad_gm1 = ihp.ic_gm1.strt(length=80, width=80,).put(gm1_str_bot.pin['b0'].move(0, -80/2, 0), flop=True)*
- *botpad_gpd = ihp.ic_gpd.strt(length=70, width=70, arrow=False).put(gm1_str_bot.pin['b0'].move(-5, -80/2, 0), flop=True)*
- nd.export_gds(filename='EAM.gds')



Practice Session

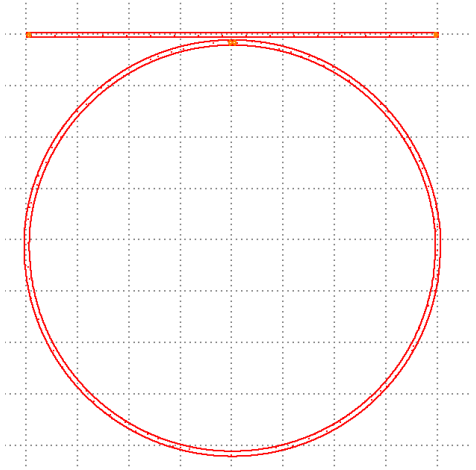
Task 1: Ring Resonator Design

Task 2: Add Drop Ring Resonator Design

Practice Session

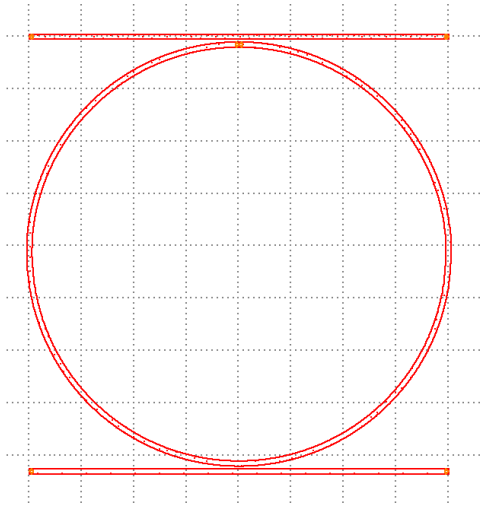
- **Extercise 1:**

- *strt_top = ihp.ic.strt(length=2*40, width=1.0).put(0,0)*
- *ring = ihp.ic.bend(radius=40, angle=-360, width=1.0).put(40, -1-0.5)*
- *nd.export_gds(filename='ring.gds')*

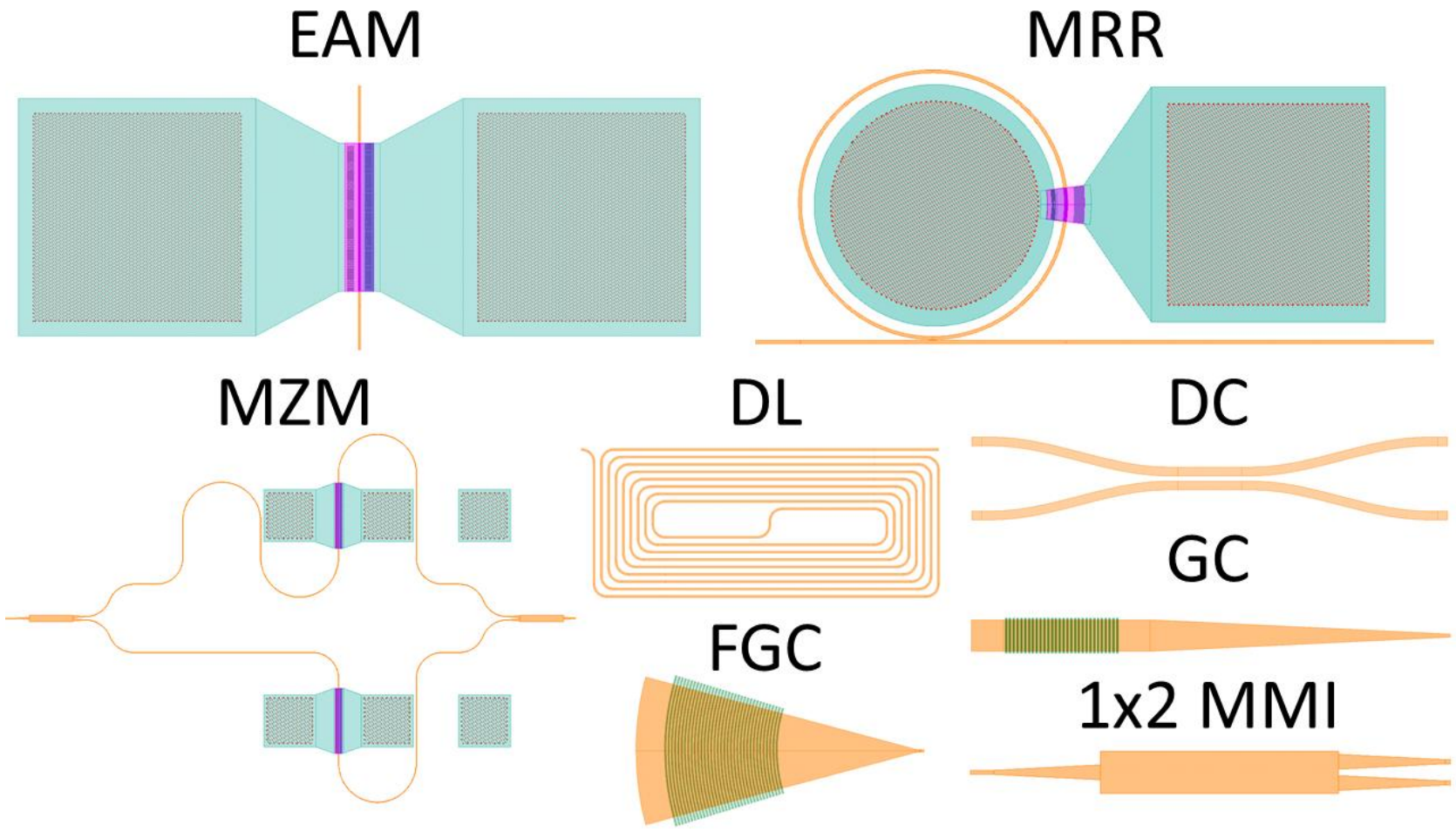


- **Extercise 2:**

- *strt_top = ihp.ic.strt(length=2*40, width=1.0).put(0,0)*
- *ring = ihp.ic.bend(radius=40, angle=-360, width=1.0).put(40, -1-0.5)*
- *strt_bot = ihp.ic.strt(length=2*40, width=1.0).put(0, -(2*40+2*1+2*0.5))*
- *nd.export_gds(filename='ring.gds')*



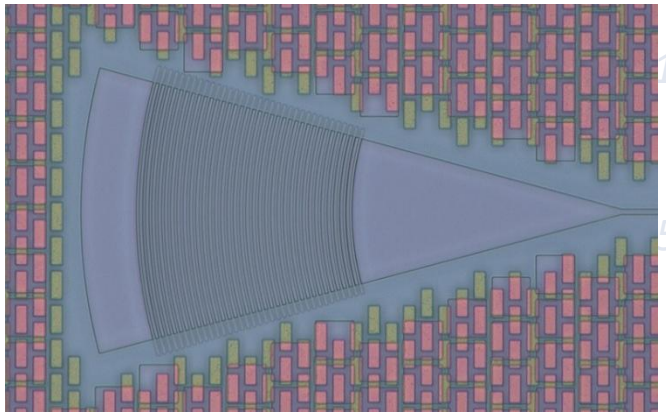
Pcell/Library Components Overview

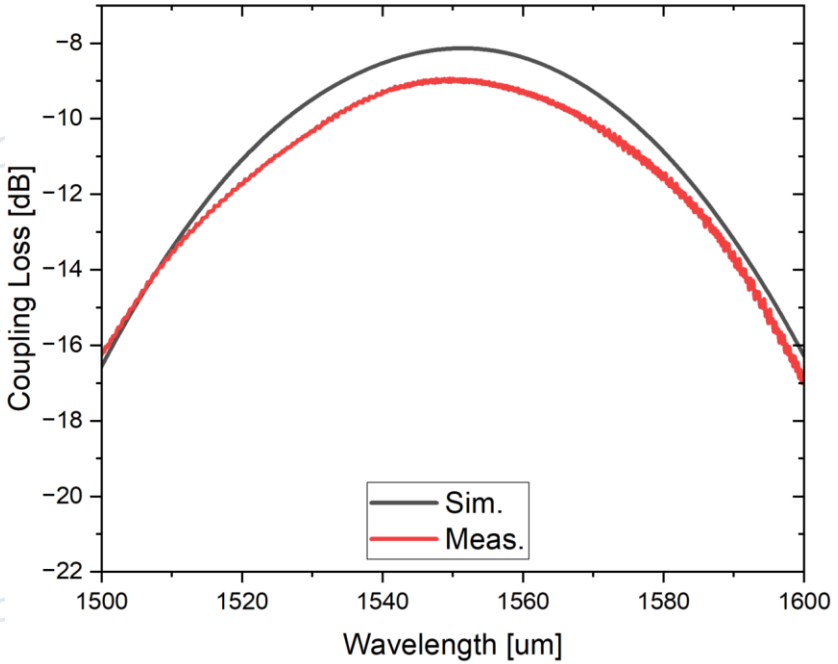


Pcell/Library Components

Standard Grating Coupler for C-Band

GratingCoupler_SiN1550(period=1.114, ff=0.5,





Focussing Grating Coupler for C-Band

FocusingGratingCoupler_SiN1550(wg_width =

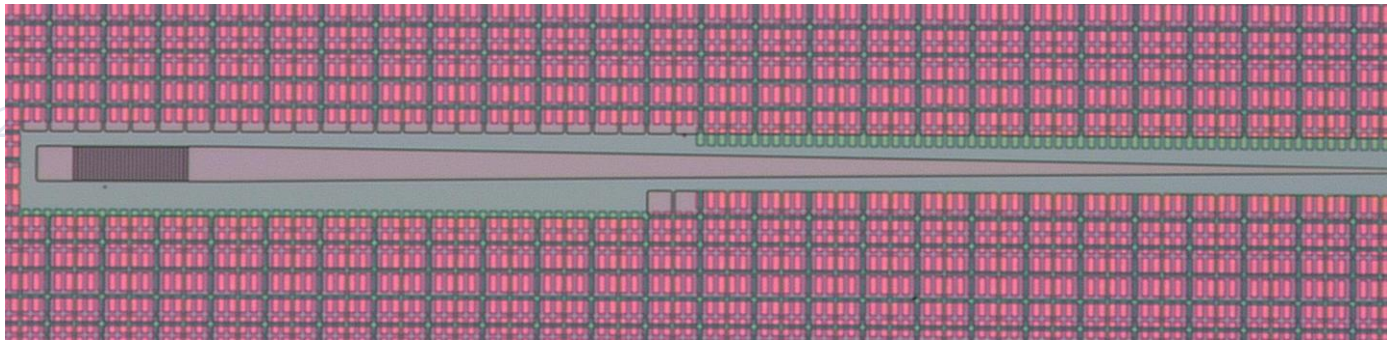
omega = 30, width=100)

Standard Single Ring Resonator

SingleRingResonator_SiN(wg_width=0.9, gap=0.3, radius=40, wg1=None, wg2=N

Standard Add-Drop Ring Resonator

AddDropRR_SiN(wg_width=0.9, gap1=0.6, gap



Pcell/Library Components

50:50 1x2 MMI for C-Band

MMI1x2_SiN1550(length=33.10, width = 3, tp_length=30,

50:50 2x2 MMI for C-Band

MMI2x2_SiN1550(wg_width=0.9, leng 50, tp_width = 3.2,

50:50 DC for C-Band

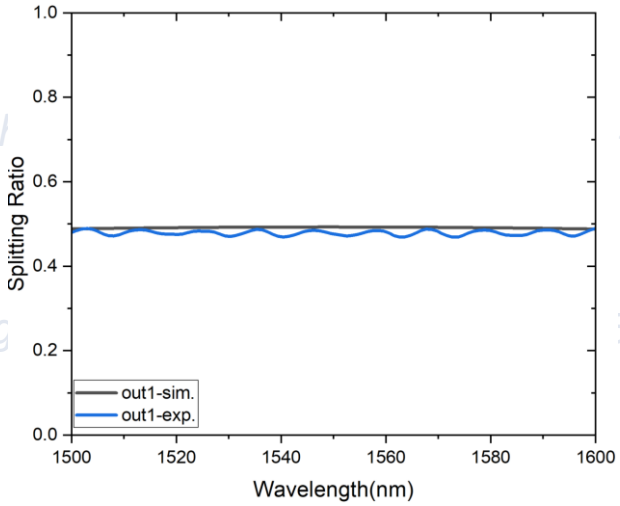
DC_SiN_1550(wg_width= 0.9, gap = 0.5, coupling_length=6.5,

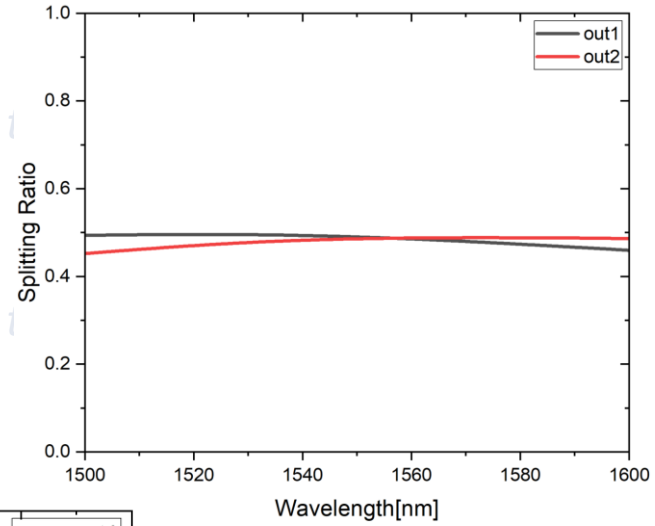
Standard Rectangular PADs

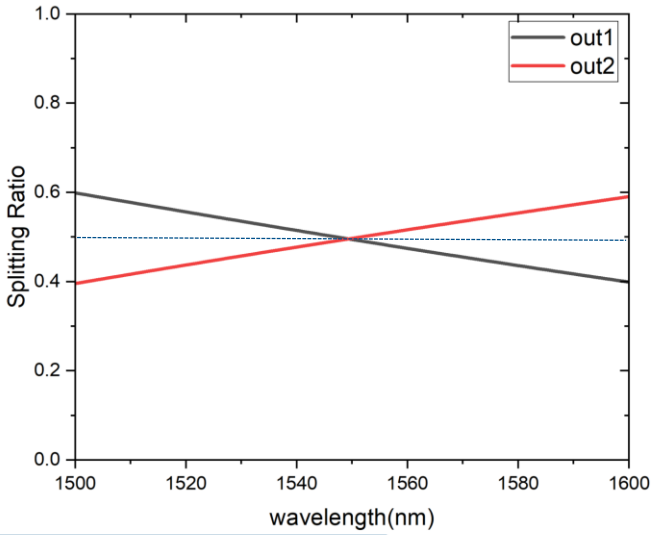
PAD_Rec(pad_length=80, pad_width=80)

Standard Circular PADs

PAD_Cir(pad_radius=80)







Pcell/Library Components

Graphene EAM With Contact for

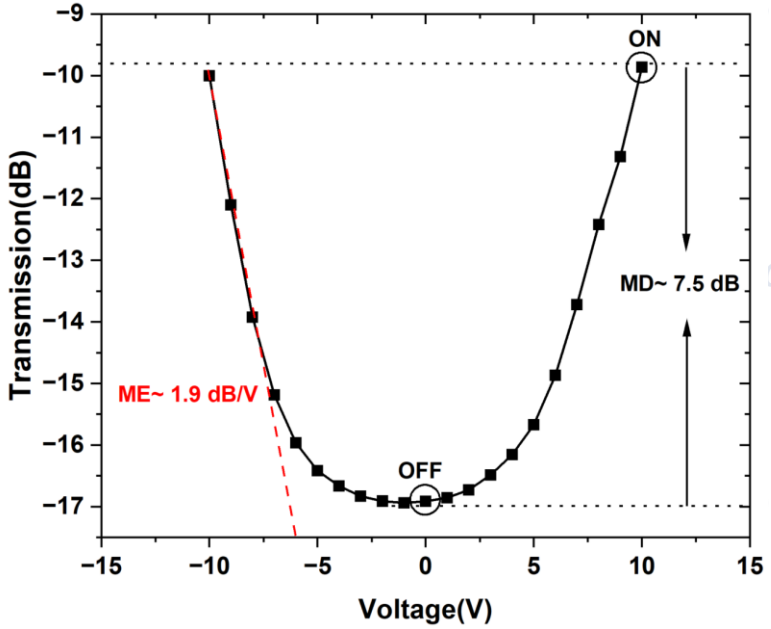
EAMCont_SiN(wg_width = 0.9, gr_len… 6, gc…

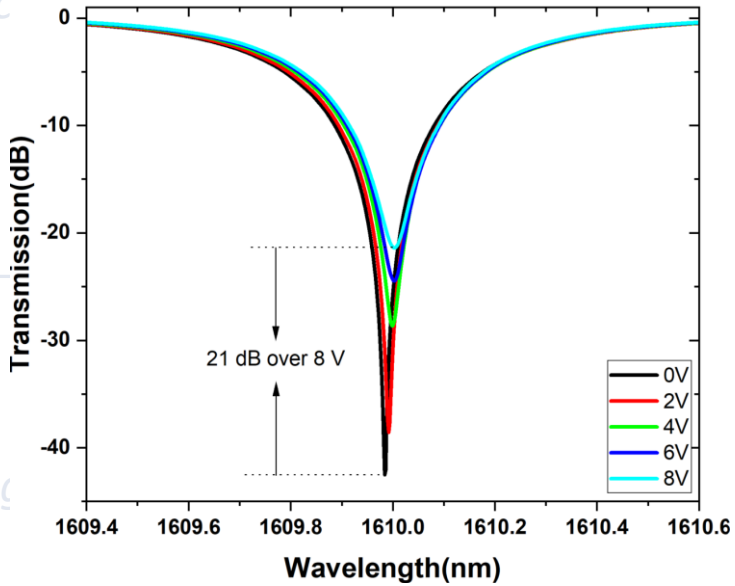
Graphene EAM with Contact and

EAM_All_SiN(wg_width = 0.9, gr_leng… cont_…

Graphene Ring Modulator

Graphene_RM_All(radius=40, gap=0.5… acing…
wg_width =0.9)





Graphene Racetract Modulator

Racetrack_RM_All(radius=40, gap=0.5, via_size=0.36, via_gap=0.36, layer_spacing=0.8, graphene_length = 25, coupling_length = 25, wg_width =0.9)

Pcell/Library Components

Graphene Hallbar

IHP_HALL(L_cha=20,W_cha=6,L_arm=10,W_arm=2,wm=10,wm1=4, d_arm=3, overlap_cha=6, overlap_arm=2, l_pad=80, WP=0.16, DPx=0.1733,DPy=0.1733, C_patt="null",pari="even", gated=False)

Graphene GFET

IHP_GFET(L_cha=2,W_cha=23,overlap=3, wm=10, l_pad=80, WP=0.12, DPx=0.2,DPy=0.2, C_patt="via",pari="even")

Graphene TLM

IHP_TLM_2p(Wg=6,wm1=9,wm2=4,d=([1, 1.5, 2, 3, 3.5, 5, 7, 12, 16]),pari="even",C_patt="null",WP=0.36, DPx=0.2,DPy=0.2,l_pad=80,gated=False)

Graphene Capacitor

GraphCap(graph_overlap = 1, cap_length=40, cont_size = 0.36, cont_space= 0.5, pad_size = 80, pad_pitch = 100)

Example: Electro-absorption Modulator (EAM2)

Step 1: Import libraries and PDK

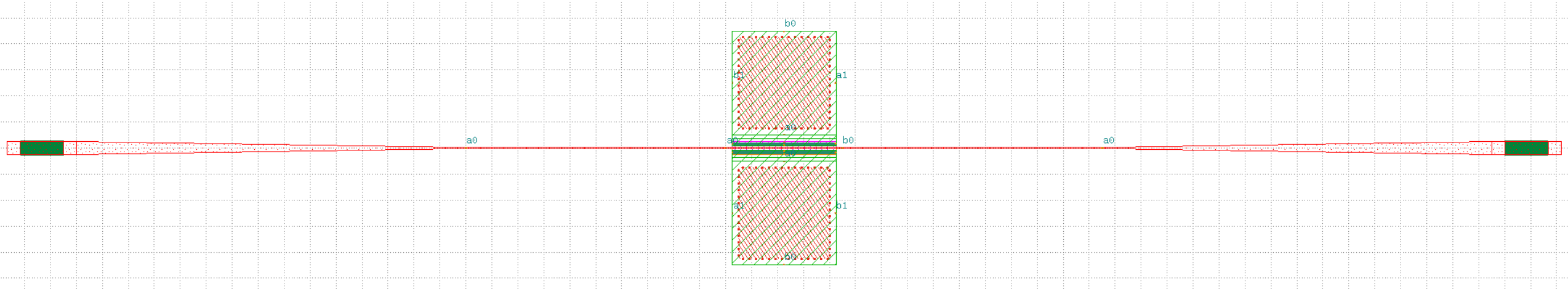
- *Import nazca as nd*
- *Import IHP_PDK as ihp*

Step 2: Import Pcell/Library

- *gc = ihp.GratingCoupler_SiN1550()*
- *pad = ihp.PAD_Rec()*
- *eam = ihp.EAM_All_SiN(wg_width = 1, gr_length=80, pad_size = 80)*

Step 3: Placning the Pcell/Library

- *eam1 = eam.put(0,0,0)*
- *gc_l = gc.put(eam.pin['a0'].move(200, 0, 0))*
- *gc_r = gc.put(eam.pin['b0'].move(200, 0, 0))*
- *ihp.ic.strt_p2p(gc_l.pin['a0'], eam1.pin['a0'], width = wg_width).put()*
- *ihp.ic.strt_p2p(gc_r.pin['a0'], eam1.pin['b0'], width = wg_width).put()*



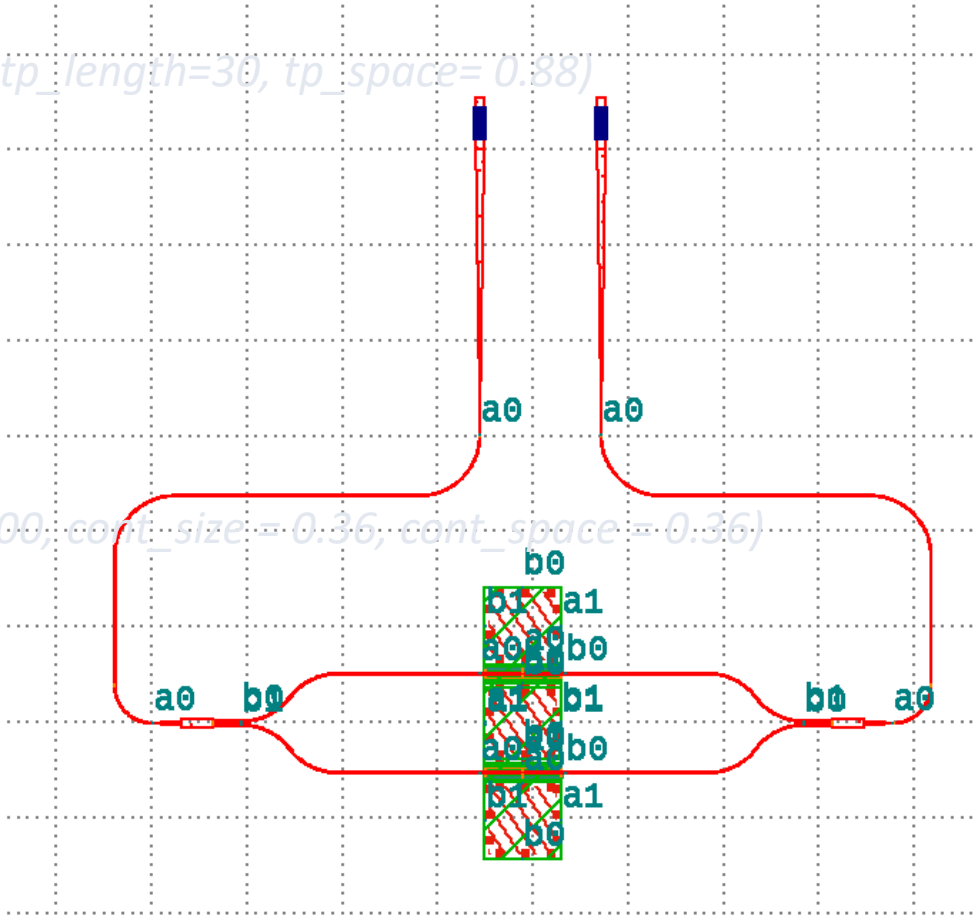
Practice Session

Task: Mach-Zehnder Modulator Design

Practice Session

MMI1x2_SiN1550(length=33.10, width=7.5, wg_width=0.9, tp_width = 3, tp_length=30, tp_space= 0.88)

EAM_All_SiN(wg_width = 0.9, gr_length=80, pad_size = 80, pad_pitch = 100, cont_size = 0.36, cont_space = 0.36)



Practice Session

```python
# Import library components
eam= ihp.EAM_All_SiN()
mmi= ihp.MMI1x2_SiN1550()
gc = ihp.GratingCoupler_SiN1550()

# Placeing Components
left_mmi = mmi.put()
top_eam = eam.put(left_mmi.pin['b0'].move(250, 50))
bot_eam = eam.put(left_mmi.pin['b1'].move(250, -50), flip=True)
right_mmi = mmi.put('b1', top_eam.pin['b0'].move(250, -50))

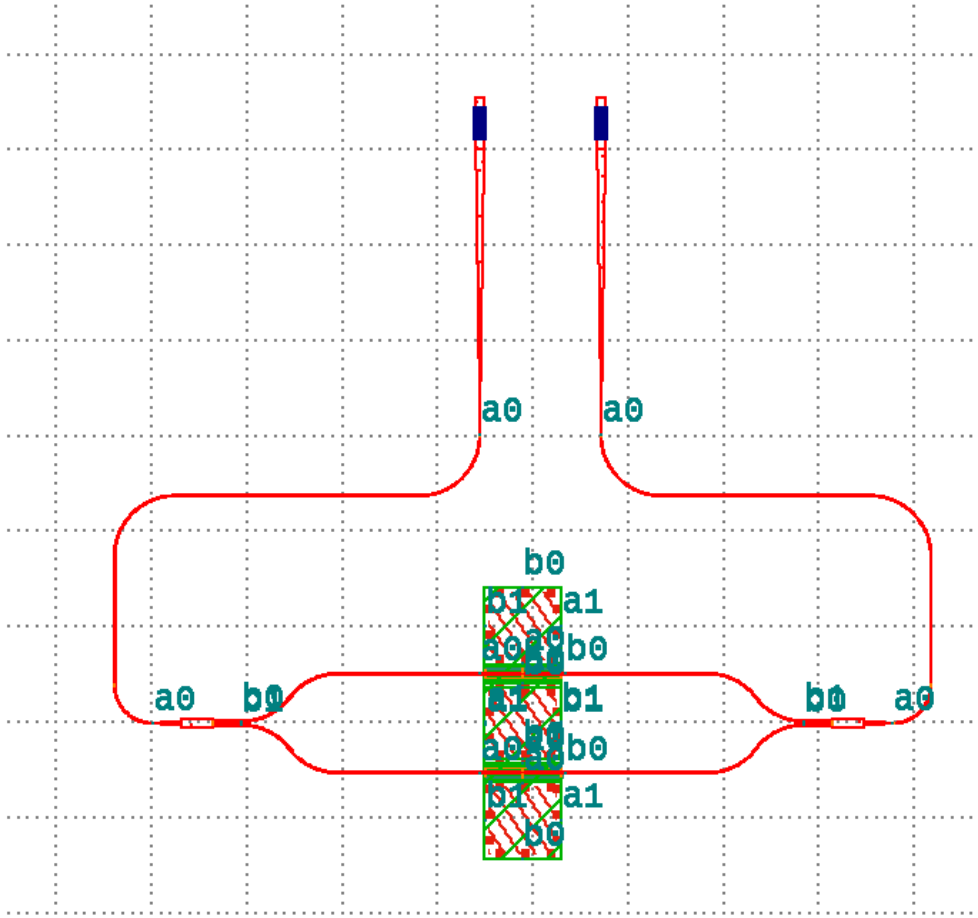
# top Connections
ihp.ic.sbend_p2p(left_mmi.pin['b0'], top_eam.pin['a0'], width = 1).put()
ihp.ic.sbend_p2p(right_mmi.pin['b1'], top_eam.pin['b0'], width = 1, ).put()

# bot Connections
ihp.ic.sbend_p2p(left_mmi.pin['b1'], bot_eam.pin['a0'], width = 1).put()
ihp.ic.sbend_p2p(right_mmi.pin['b0'], bot_eam.pin['b0'], width = 1, ).put()

## Grating Couplers
gc1 = gc.put(left_mmi.pin['b0'].move(250, 300, 90))
gc2 = gc.put(left_mmi.pin['b0'].move(377, 300, 90))

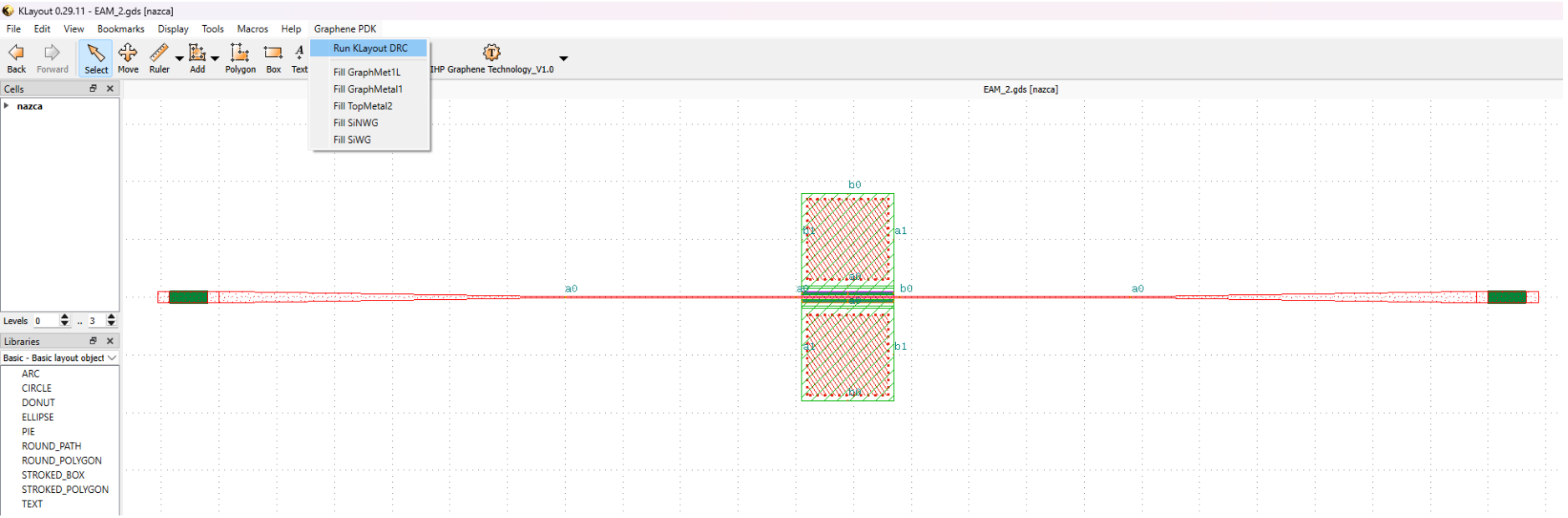
#  Connections
bend1 = ihp.ic.bend(radius=40, angle=-90, width=1.0).put(left_mmi.pin['a0'])
bend2 = ihp.ic.bend(radius=40, angle=90, width=1.0).put(right_mmi.pin['a0'])
ihp.ic.sbend_p2p(gc1.pin['a0'], bend1.pin['b0'], width = 1).put()
ihp.ic.sbend_p2p(gc2.pin['a0'], bend2.pin['b0'], width = 1).put()

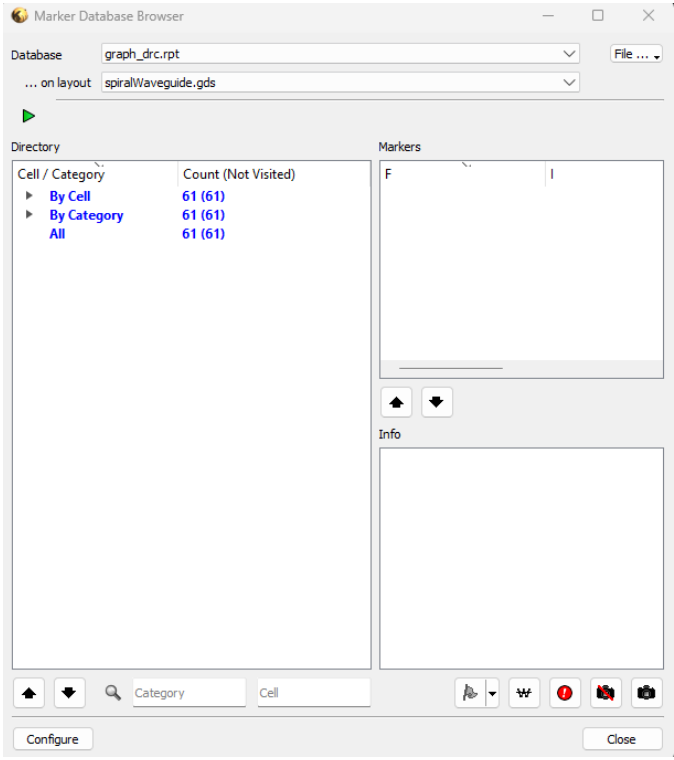
nd.export_gds(filename='MZM.gds')
```



IHP Graphene Klayout PDK: Design Rule Check

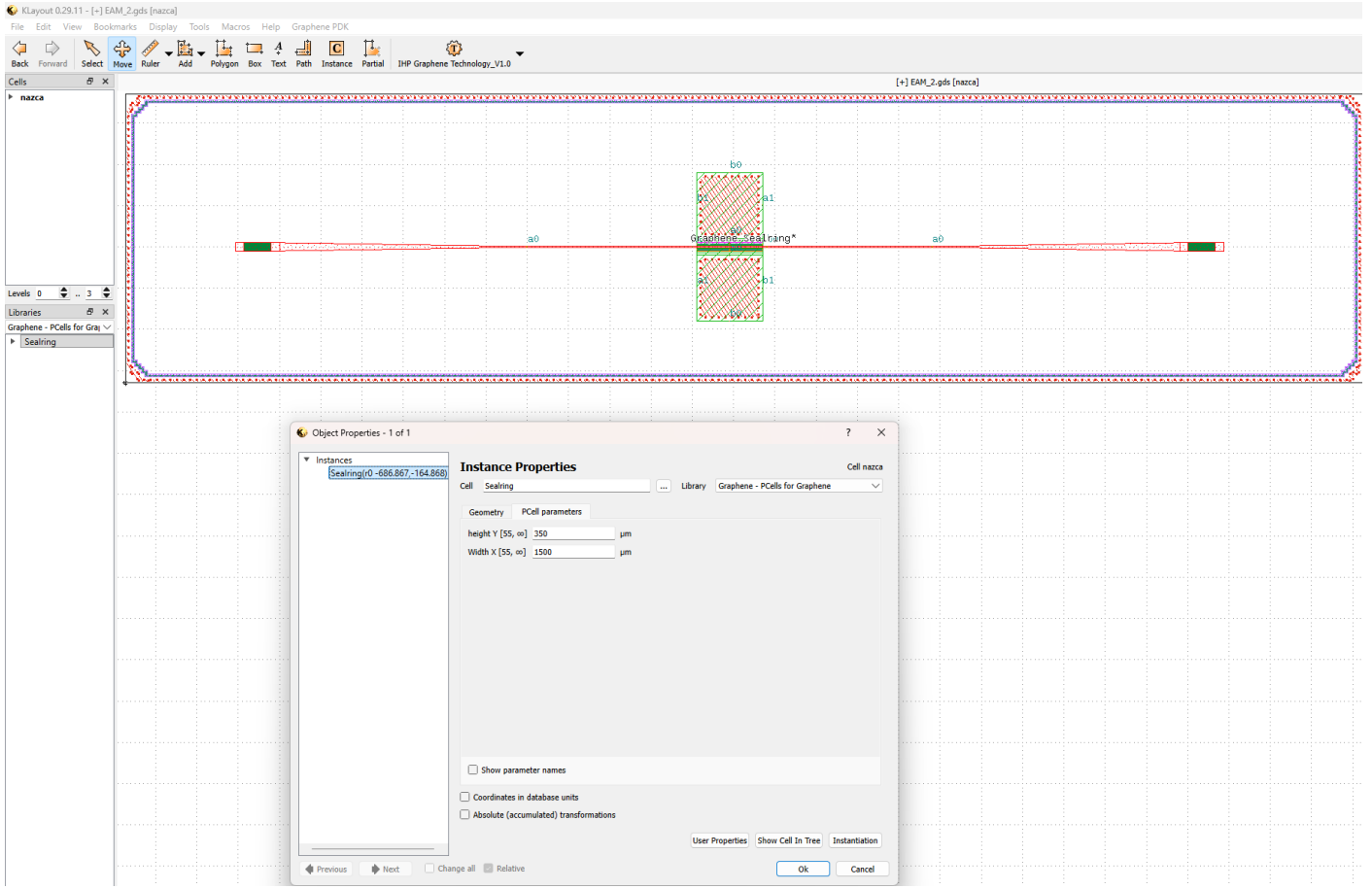
- Click ``Graphene PDK`` and Select ``Run Klayout DRC`` . A DRC script will compile and report the violation of the design rules in layout (shown in the Right Figure) and can be navigate the DRC error by clicking in cell/Category





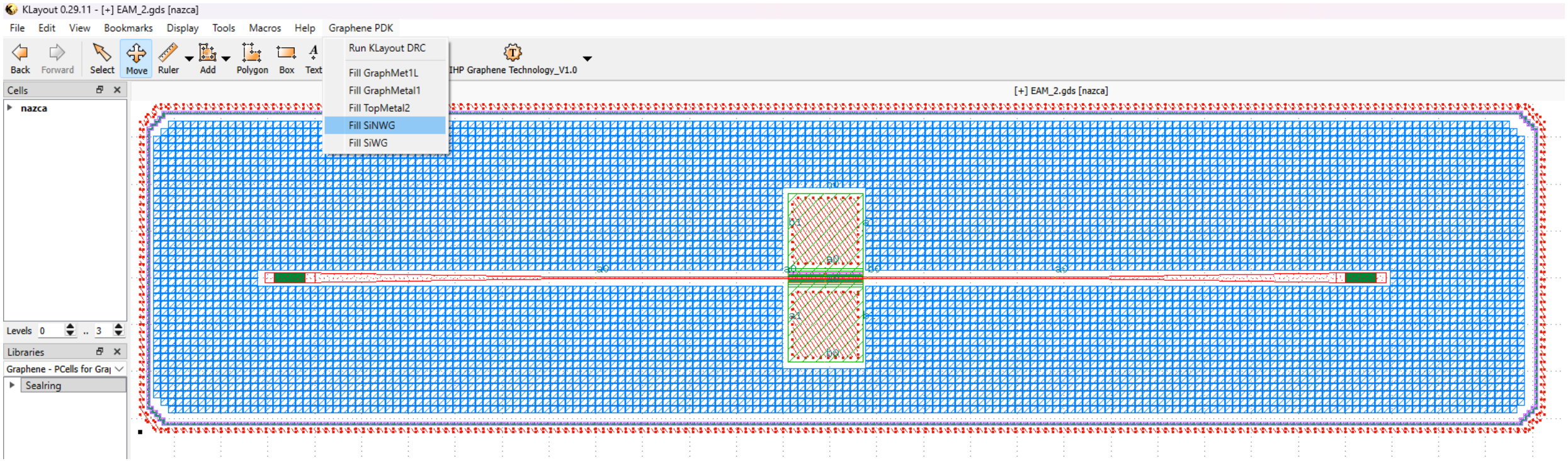
IHP Graphene Klayout PDK: SealRing

- Select ``Sealring`` and drag into the layout window
- A sealring ``Editor Options`` will open on the left bottom
- Select ``Pcell`` and give sealring height and width according to your layout size (please see the Layout Manual for Sealring DRC)
- A Sealring object will show around the layout.
- Place the sealring in the centre of the layout



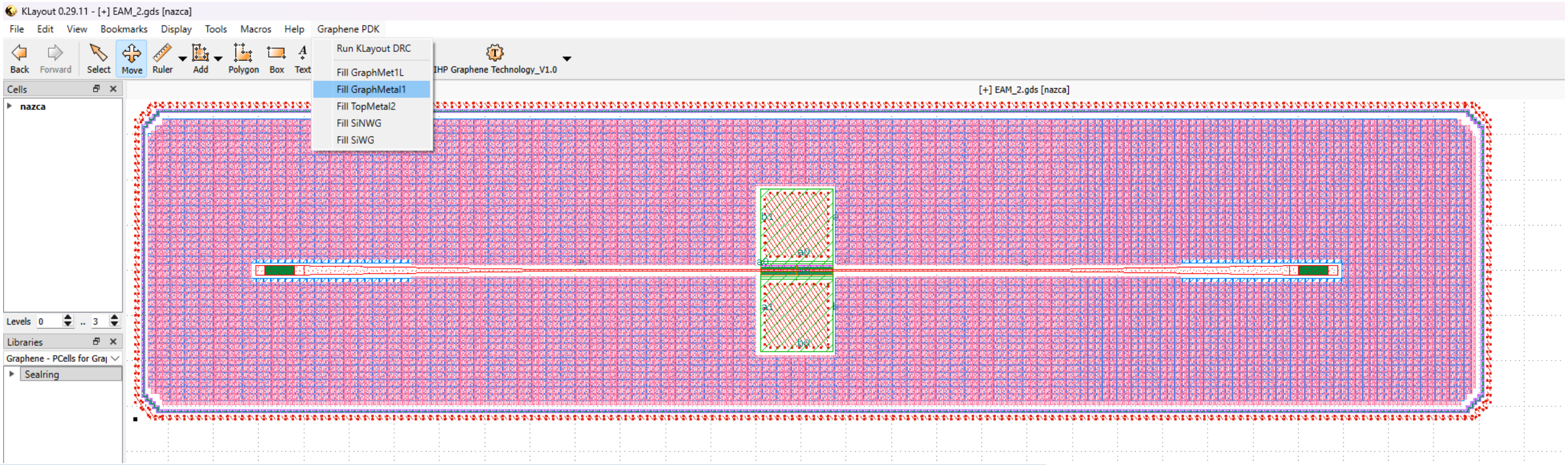
IHP Graphene Klayout PDK: Waveguide Filler

- Click ``Graphene PDK`` and Select ``Fill SiNWG`` . A Filler script will compile and generate the Silicon nitride waveguide Filler within the sealring.



IHP Graphene Klayout PDK: Metal Filler

- Click ``Graphene PDK`` and Select ``Fill Graphmetal1`` . A Filler script will compile and generate the Graphene meteal1 Filler within the sealring.



Thank you for your attention!

IHP GmbH – Innovations for High Performance Microelectronics
Im Technologiepark 25
15236 Frankfurt (Oder)
Tel.: +49 (0) 335 5625 241
E-Mail: raju@ihp-microelectronics.com

