CAS
IEEE CIRCUITS AND SYSTEMS SOCIETY

UNIC-CASS
Mock Tapeout
“250 MHz *Quadrature Delay Locked Loop* ”

27.01.2026

Fundación
FULGOR

UNS
UNIVERSIDAD
NACIONAL DEL SUR

UTN
Facultad Regional Córdoba

UNC
Universidad
Nacional
de Córdoba

Content

1. **Fundamentals**
2. **QDLL Project:** Block Diagram and Specifications
3. **Schematics**: TOP and Sub-blocks: Symbols & Schematics & Layouts
4. **Characterization Testbenches**
5. **TOP Simulation:** Testbench with the IO-Cells. Characterization Testbenches.
6. **QDLL Layout:** Top Layout and layout within the pad ring.
7. **QDLL:** Pins and IO Cells.
8. **Project Status**

1- QDLL Fundamentals

1 - QDLL Fundamentals

The **Quadrature Delay Locked Loop (QDLL)** is a phase control system designed to delay a differential CMOS input signal by *90°*, generating a differential quadrature-phase output.

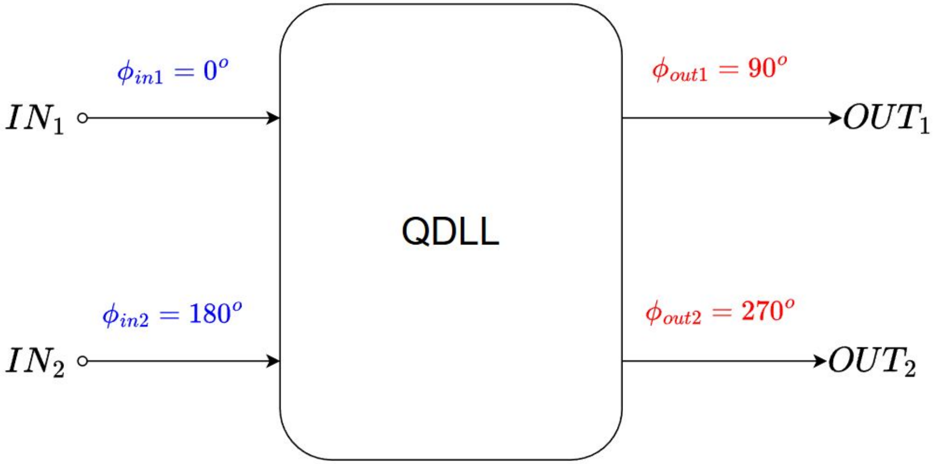
The QDLL operates through three key components:

1. **Phase Detector (PD) + Charge Pump (CP):**: Measures the phase difference between the input signal **ϕin** and the delayed output **ϕout** , and generates a proportional voltage **Vcont**.

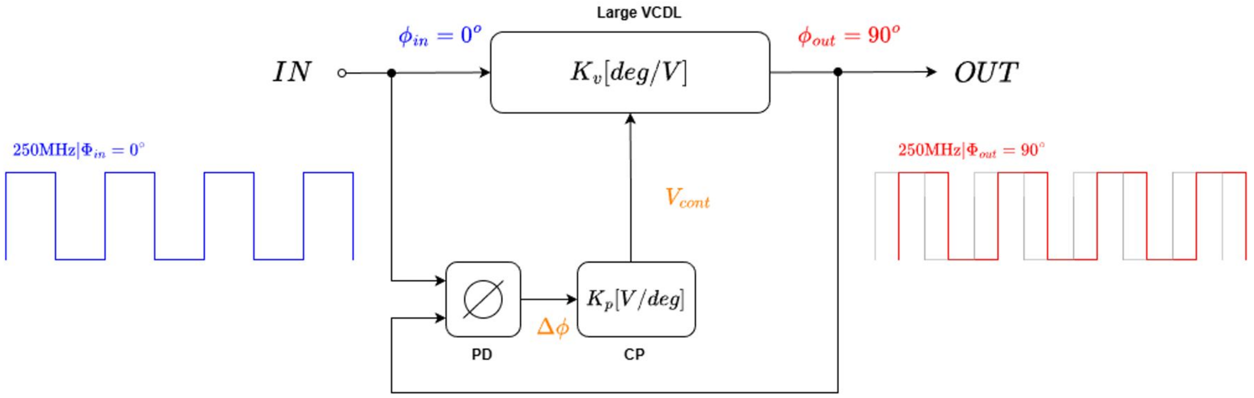
2. **Large Voltage-Controlled Delay Line (Large VCDL)**: Introduces a controllable delay to the input signal, adjusted by the control voltage **Vcont**.

3. **Feedback Loop**: Processes the phase error signal from the phase detector and adapts the VCDL delay until the phase difference is locked at *90°*.

The main goal of the QDLL is to achieve and maintain a quadrature (90°) phase shift between the input and output signals.



QDLL block diagram



Single QDLL Operation

1 - QDLL Fundamentals

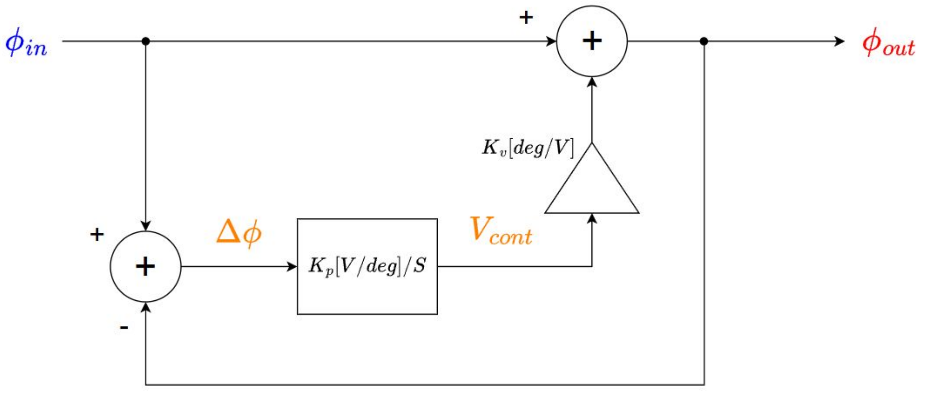
If we model the diagram of the single QDLL operation, we observe the phase detector compares the input and output phases (**ϕin** & **ϕout**), and any phase error is converted into a control voltage via the charge pump or loop filter. This voltage, **Vcont**, tunes the delay element (Large VCDL) with gain **KvKp**, adjusting the phase of the output signal **ϕout**.

On the other hand, using a single pole (1/S) in the feedback path guarantees inherent stability and ensures smooth convergence. Therefore, a simple RC network can be an effective choice for implementation.

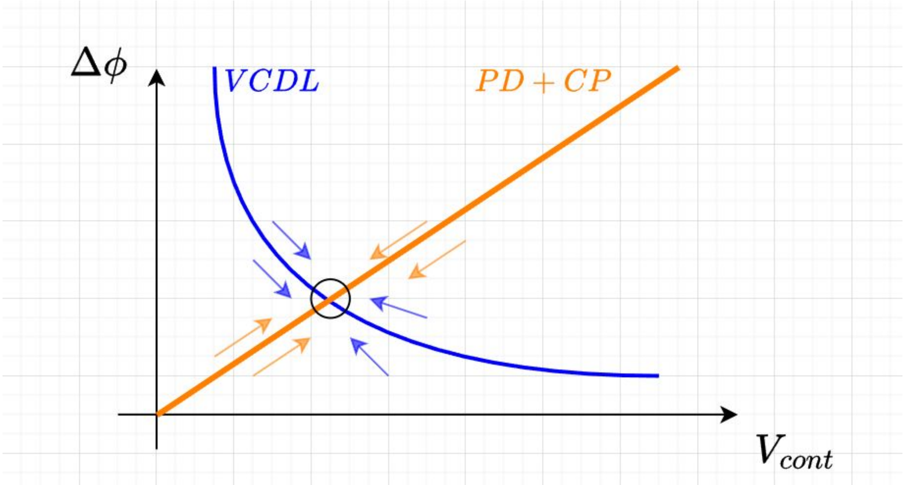
This feedback mechanism ensures that the system locks the output phase at a fixed offset (e.g., 90° for QDLL) from the input phase eliminating steady-state phase error.

The Phase Converge plot shows that:

- A stable locking point exists where the two characteristics intersect. (90° is the desired)
- The negative slope of the VCDL ensures that the loop has a stable operating point.



QDLL System Operation

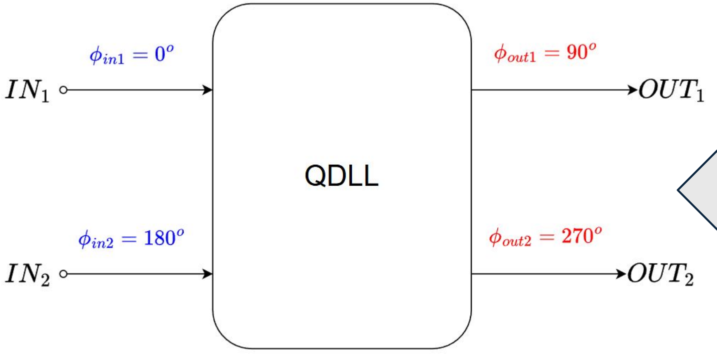


Phase Convergence

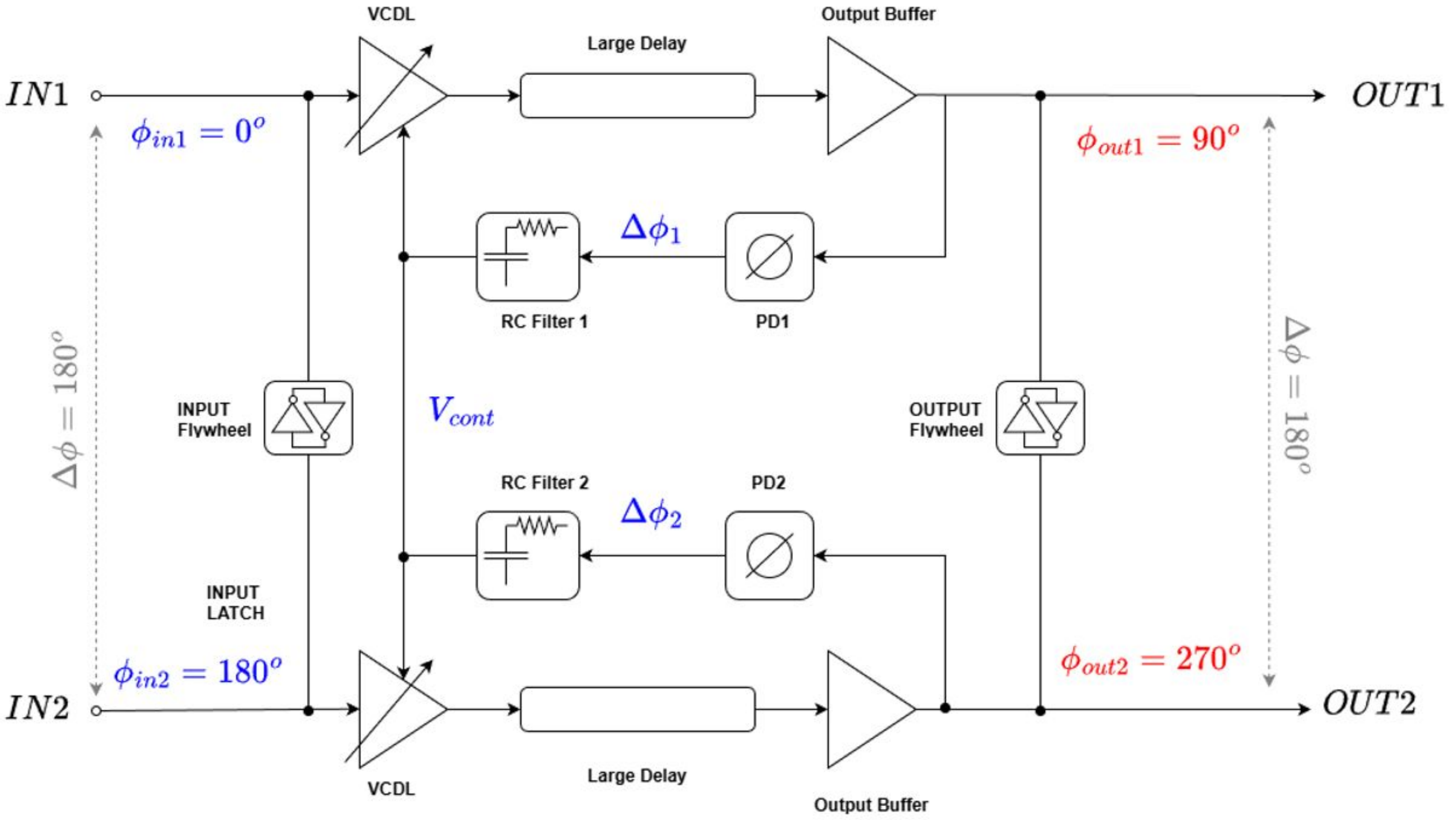
2- QDLL Project Diagram & Specifications

2 - QDLL Project Diagram

For this project, two single-ended DLLs with differential outputs are used to generate two differential signals with 90° and 270° phase shifts. The RC filters and the large VCDLs are designed to stabilize each branch at a **90°± 5°** .phase difference at **250 MHz**. To ensure synchronization between the signal pairs, flywheel cells are added.



QDLL block diagram

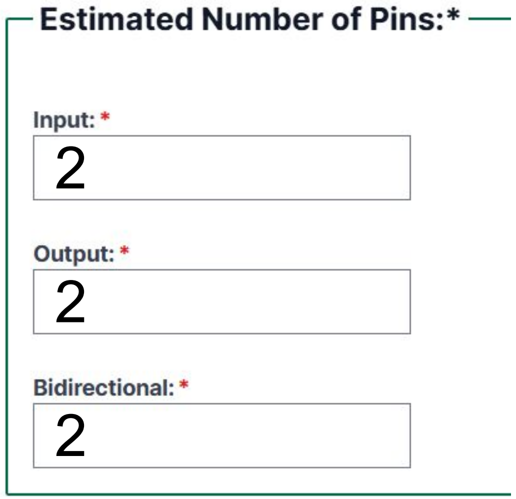


2 - QDLL Project Specifications

The design specifications were modified due to the high-frequency behavior of the I/O cells. As a result, the performance requirements were relaxed, along with adjustments to the project focus and the number of input/output pins.

Pins:

- 2 Bi-direcctional: VDD, VSS
- 2 inputs: inpad1, inpad2
- 2 outputs: outpad1, outpad2



Estimated Number of Pins:*

Input: *
2

Output: *
2

Bidirectional: *
2

ihp 130 nm

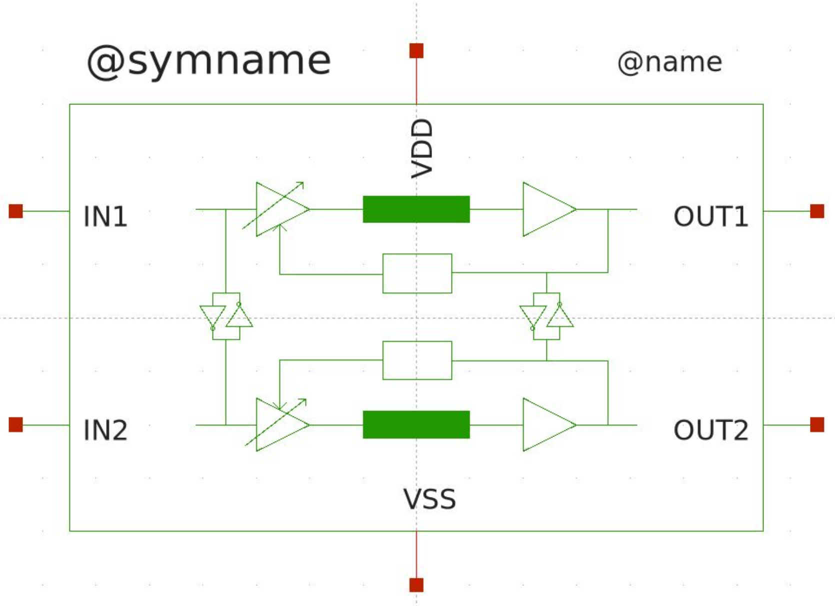
Design Type:* *
Analog

QDLL Specifications	
Boundary Conditions	**Requirement**
Techn.	CMOS
Node	130 nm
Supply Voltage	1.2 ± 5% V
Temperature	[0, 65, 125] °C
Input Amplitude	[0 , 1.2] V (CMOS)
Input Frequency	250±25 MHz
Input/Output Type	Fully Differential
Specifications	**Value**
Phase Difference	90 ± 5 degrees
Output Swing	[0 , 1.2] V (CMOS)
t_rise/t_fall output	<
Vcont_ripple	≥ 9 bits
Output Jitter	≥ 7.5 bits

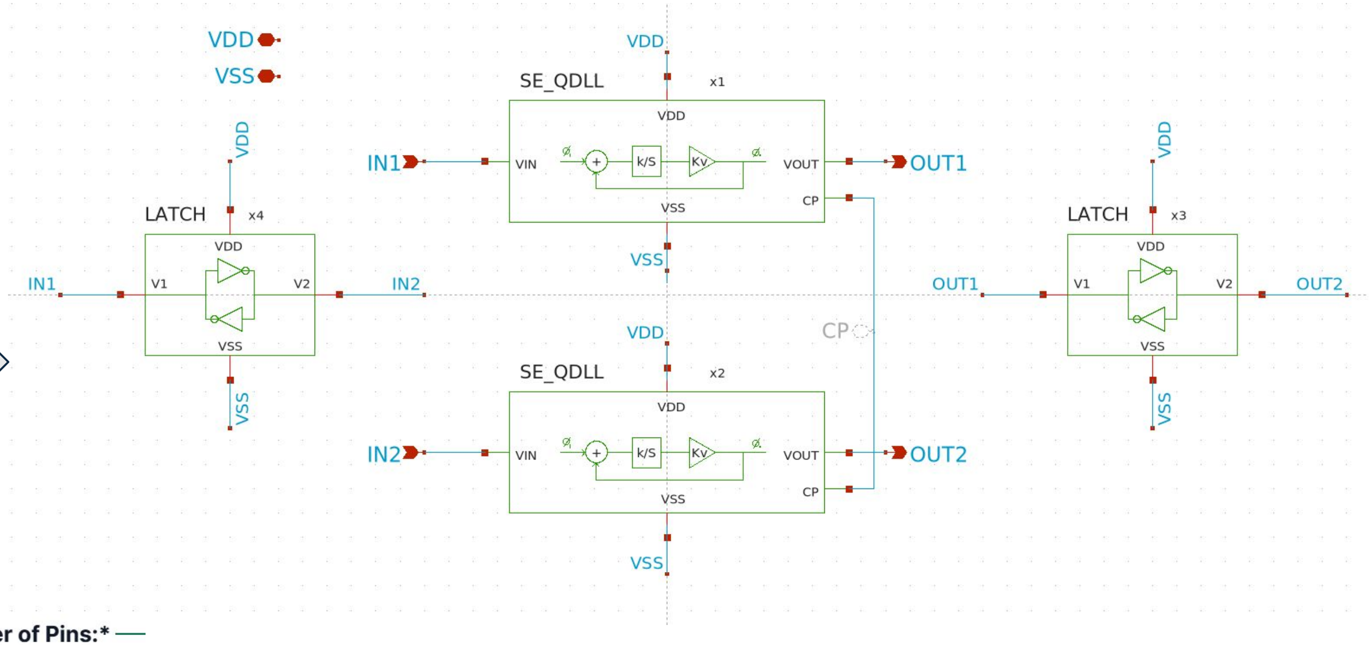
3- QDLL Project Schematics & Layouts

3 - QDLL Schematics (TOP)

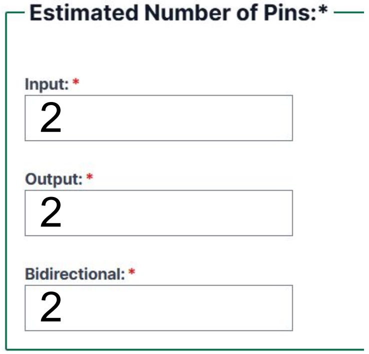
- Differential QDLL TOP:



QDLL_TOP.sym



QDLL_TOP.sch



Estimated Number of Pins:*

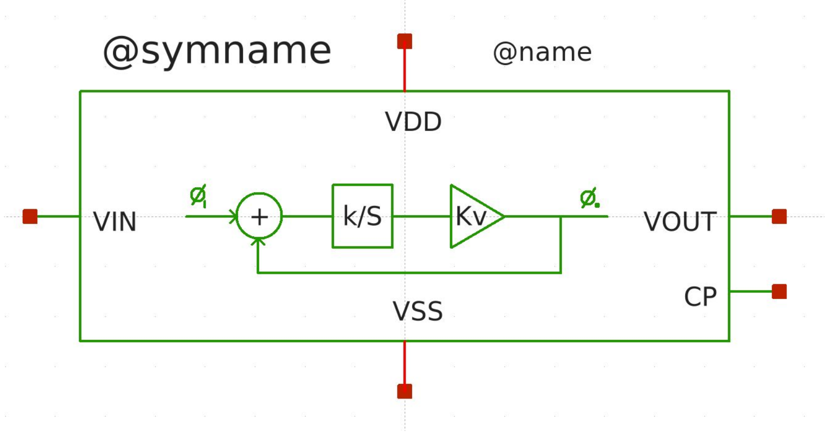
Input: *
2

Output: *
2

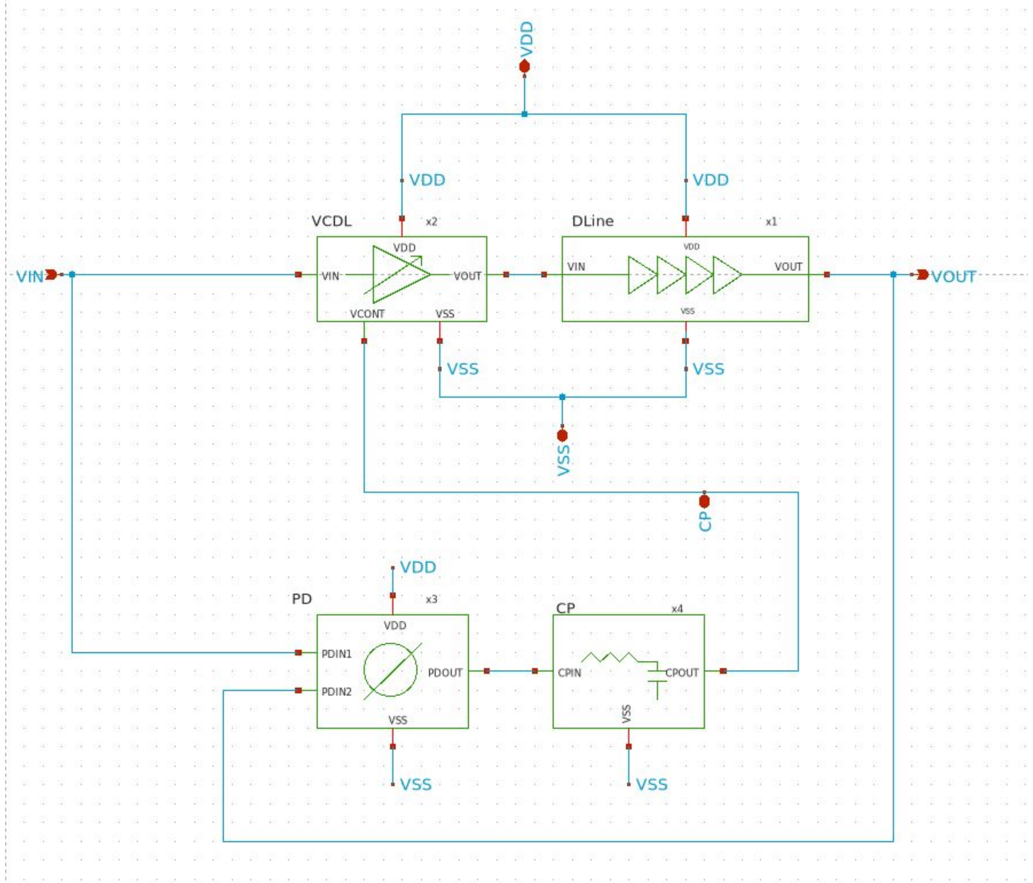
Bidirectional: *
2

3 - QDLL Schematics

- Single-Ended QDLL:



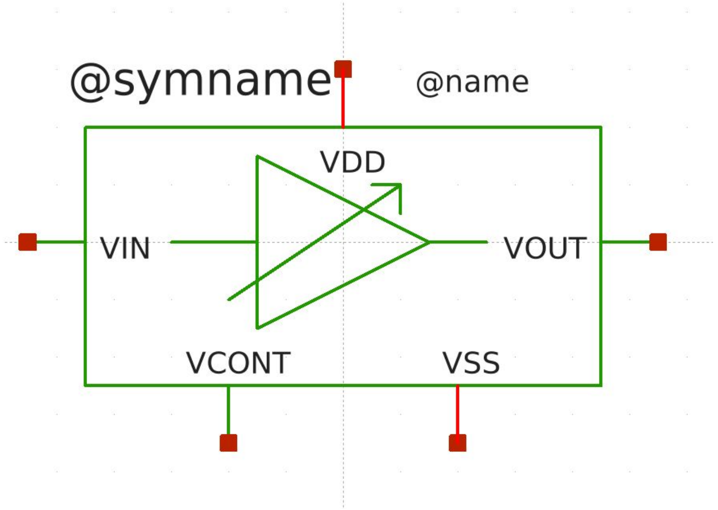
SE_QDLL.sym



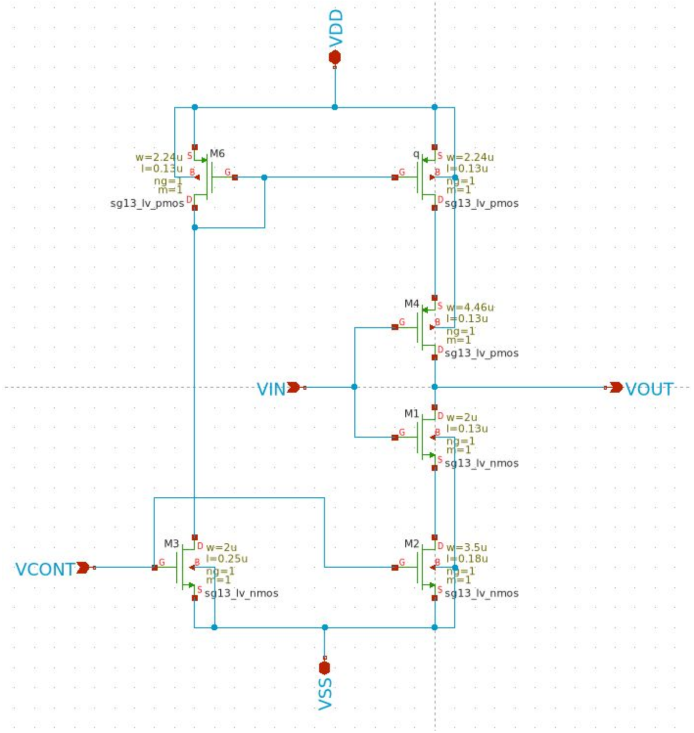
SE_QDLL.sch

3 - QDLL Schematics & Layouts

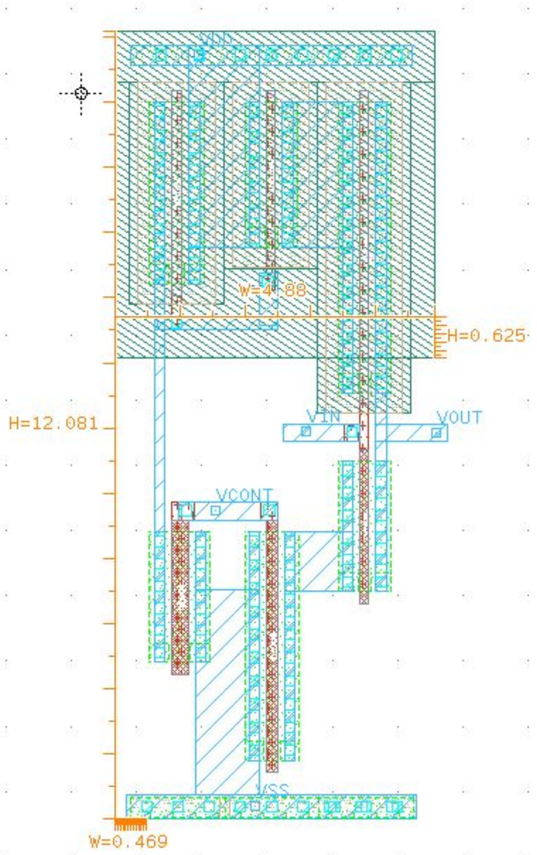
- Voltage-Controlled Delay Line (VCDL) :



VCDL.sym



VCDL.sch

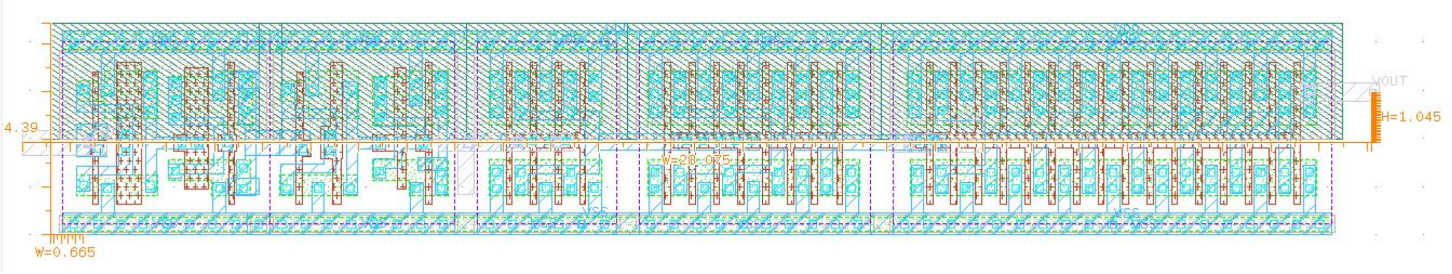


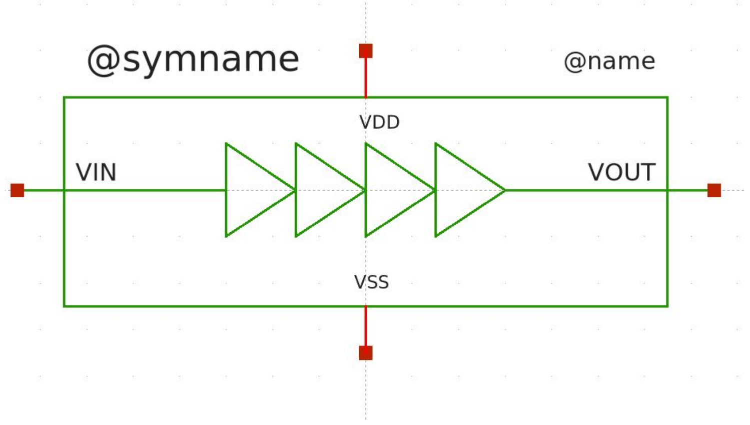
VCDL Layout

3 - QDLL Schematics & Layouts

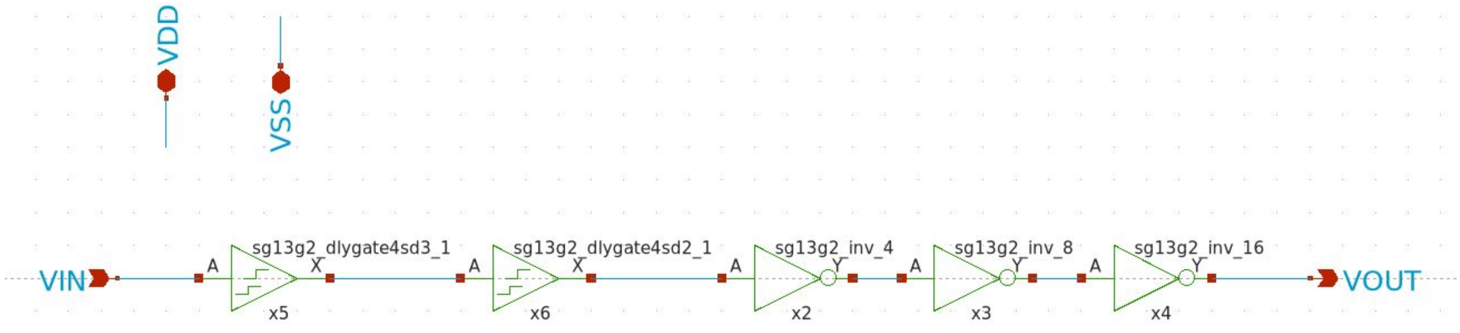
- Large delay Line + Buffer Stage (DLine) :

DLine + Buffer Layout





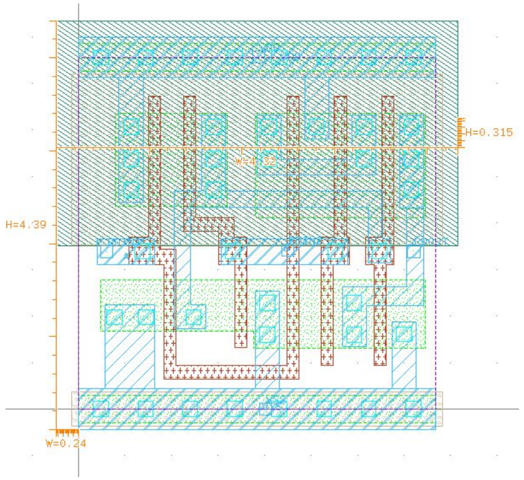
DLine.sym



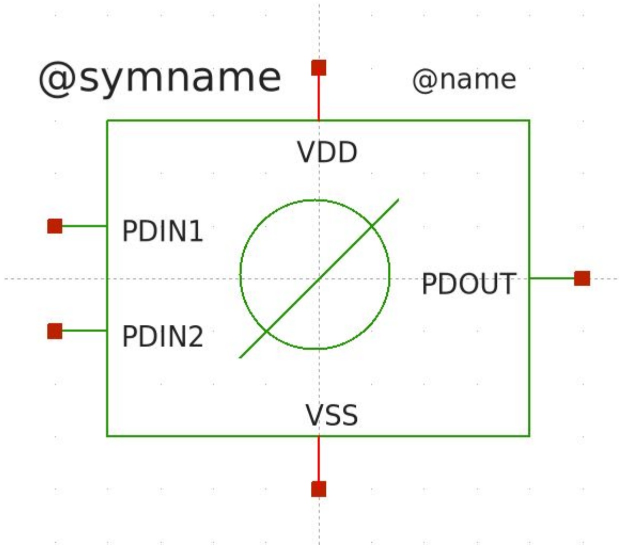
DLine.sch

3 - QDLL Schematics & Layouts

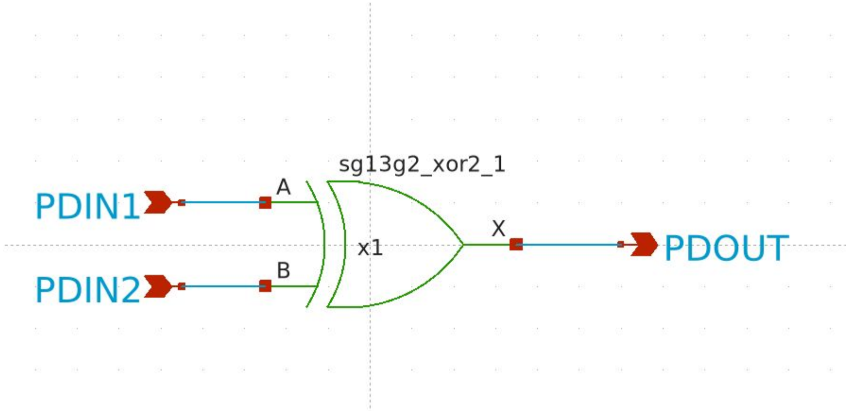
- Phase Detector (XOR cell) :



PD Layout



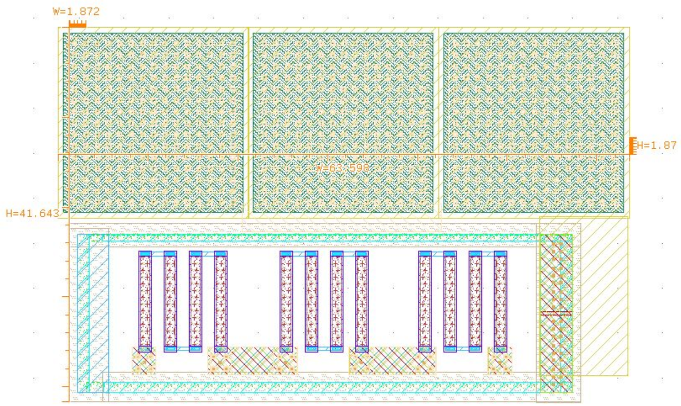
PD.sym



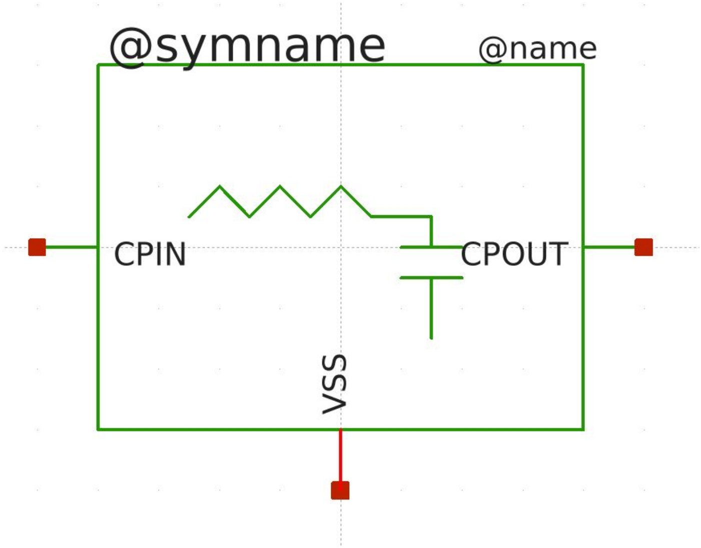
PD.sch

3 - QDLL Schematics & Layouts

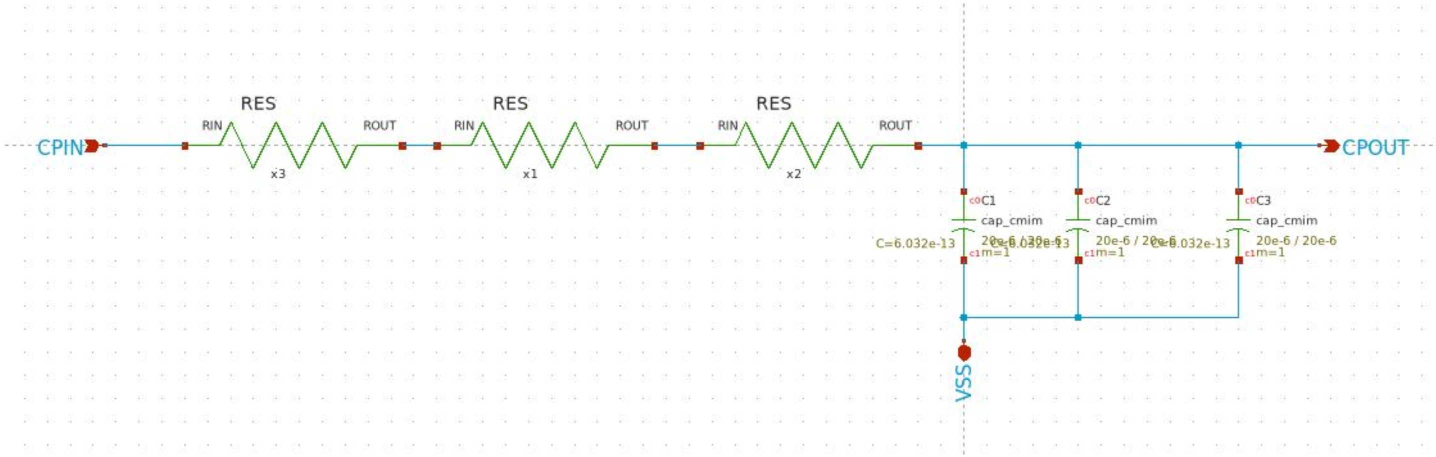
- Charge-Pump/ RC Loop Filter:



CP Layout



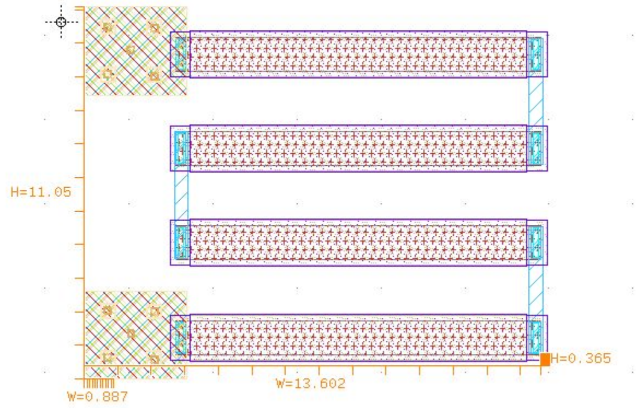
CP.sym



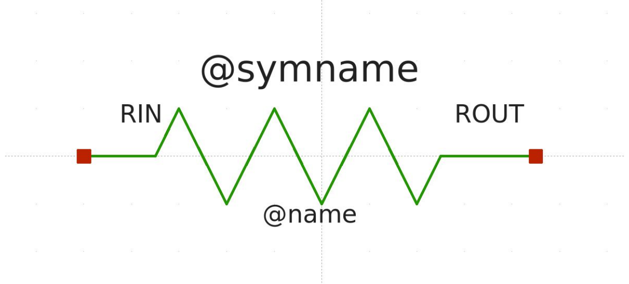
CP.sch

3 - QDLL Schematics & Layouts

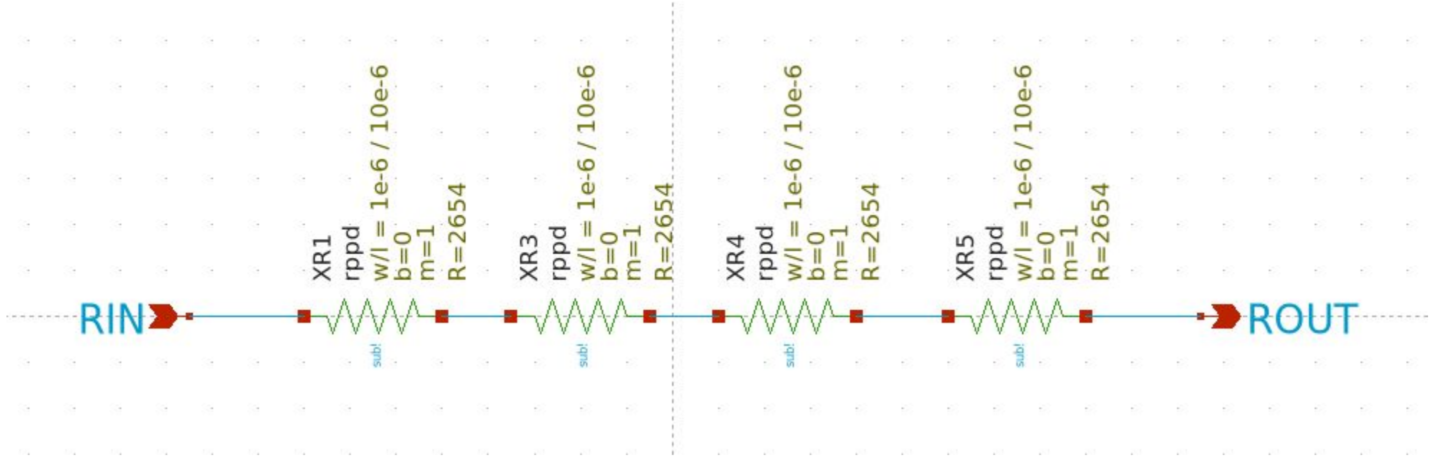
- Series Resistors:



RES Layout



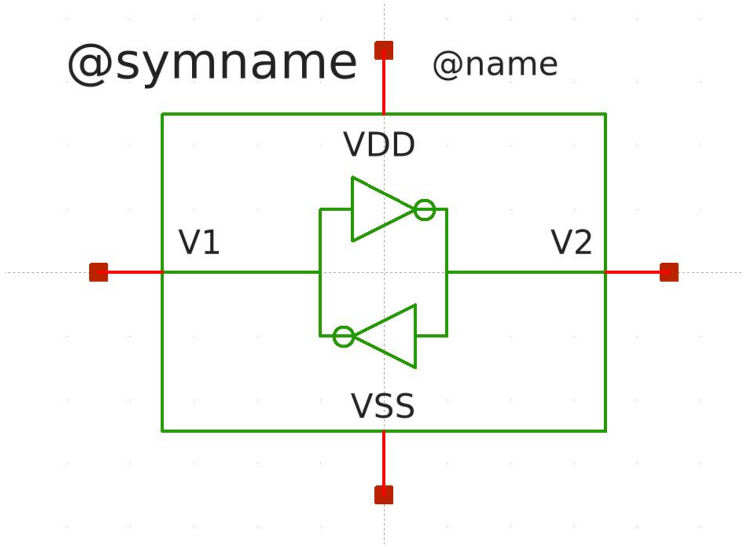
RES.sym



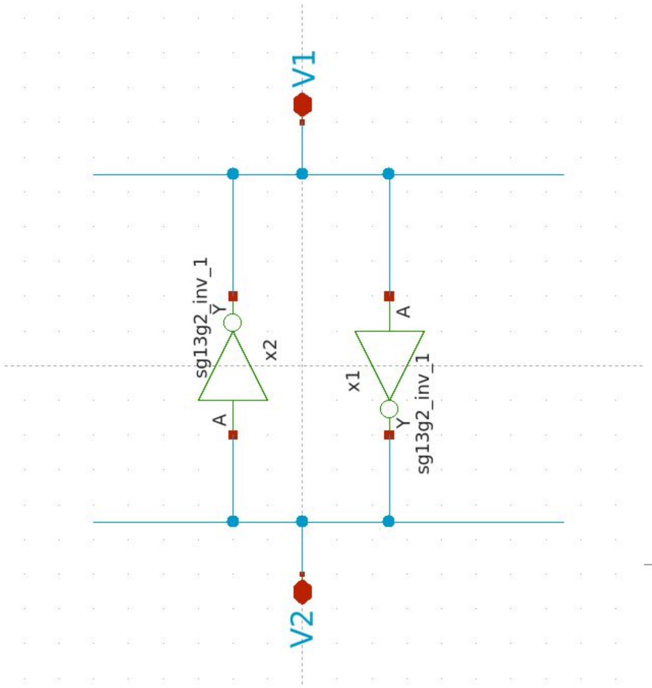
RES.sch

3 - QDLL Schematics

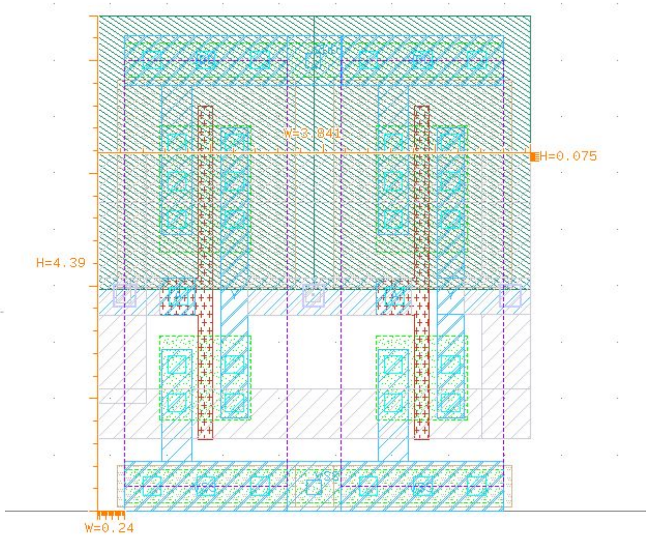
- Flywheel Latchs:



LATCH.sym



LATCH.sch

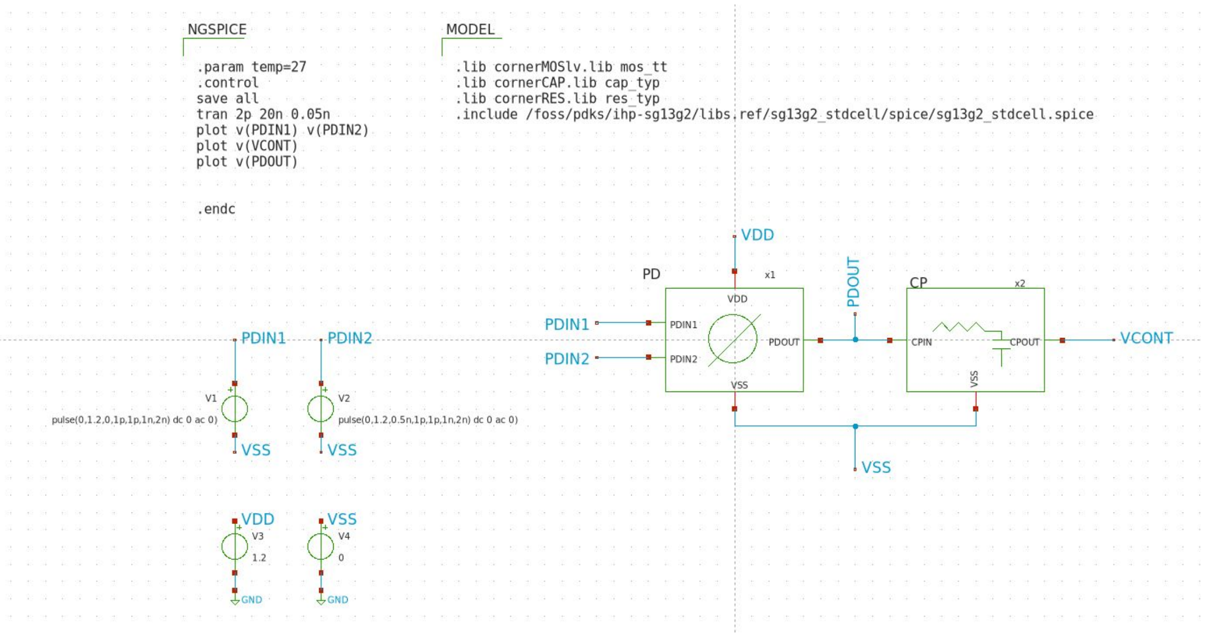


LATCH Layout

4- QDLL Characterization Testbenchs

4 - QDLL Testbenchs

- PD + CP Characterization Testbench
    - For Vcont vs Phase Difff Curves



NGSPICE

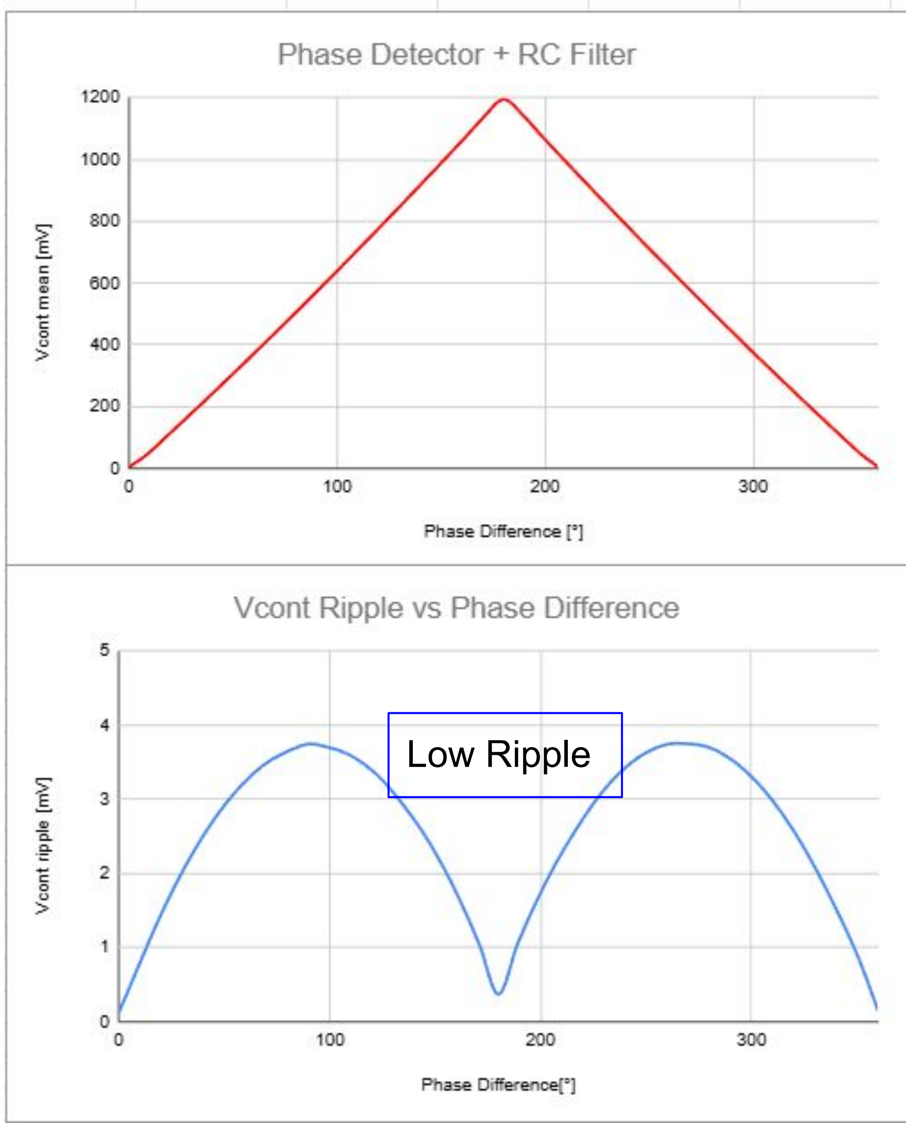
```
.param temp=27
.control
save all
tran 2p 20n 0.05n
plot v(PDIN1) v(PDIN2)
plot v(VCONT)
plot v(PDOUT)

.endc
```

MODEL

```
.lib cornerMOSlv.lib mos_tt
.lib cornerCAP.lib cap_typ
.lib cornerRES.lib res_typ
.include /foss/pdks/ihp-sg13g2/libs.ref/sg13g2_stdcell/spice/sg13g2_stdcell.spice
```

tb_TB_CP.sch

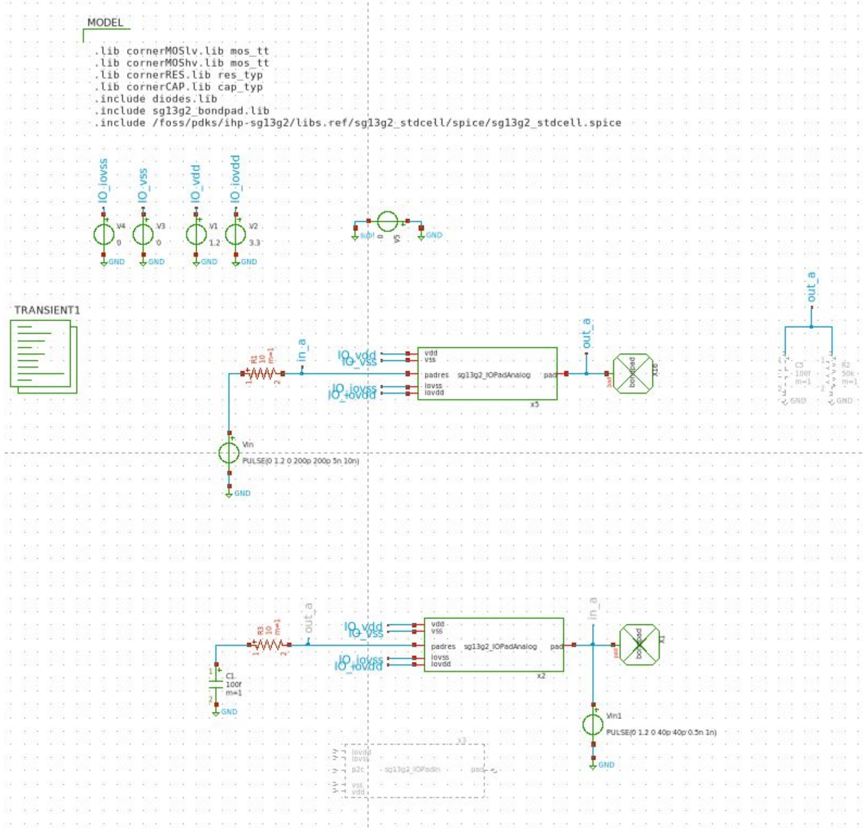
We only use the PD+CP up to 180°



Low Ripple

4 - QDLL Testbenches

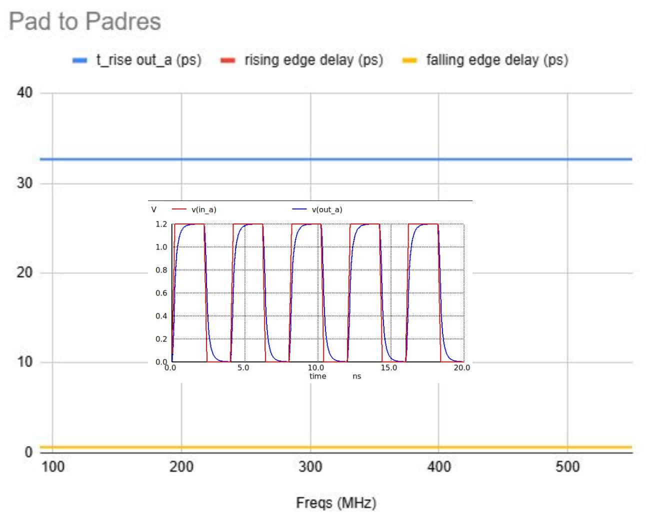
- IO CELLS Characterization Testbench
    - For BW delimitation and delay responses



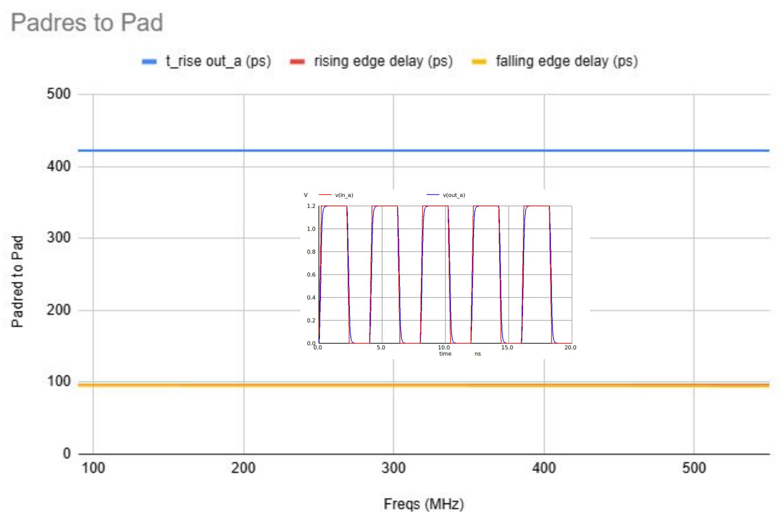
MODEL

```
.lib cornerMOSlv.lib mos_tt
.lib cornerMOShv.lib mos_tt
.lib cornerRES.lib res_typ
.lib cornerCAP.lib cap_typ
.include diodes.lib
.include sg13g2_bondpad.lib
.include /foss/pdks/ihp-sg13g2/libs.ref/sg13g2_stdcell/spice/sg13g2_stdcell.spice
```

tb_sg13g2_IOPadAnalog.sch



Pad to Padres

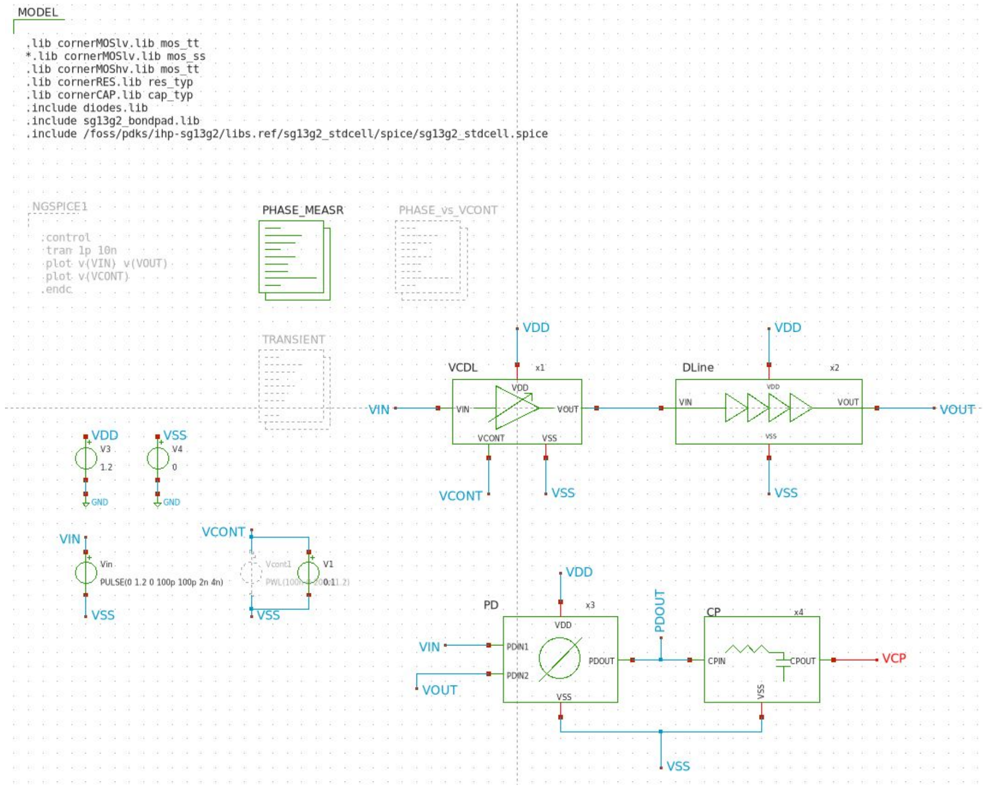


Padres to Pad

At 250 MHz the Analog IOCell presents low delay: padres-to-pad = 37°; pad -to-padres = 3°

4 - QDLL Testbenchs

- Open Loop for Large Delay Line Characterization Testbench
  - For Phase Diff vs Vcont Curves



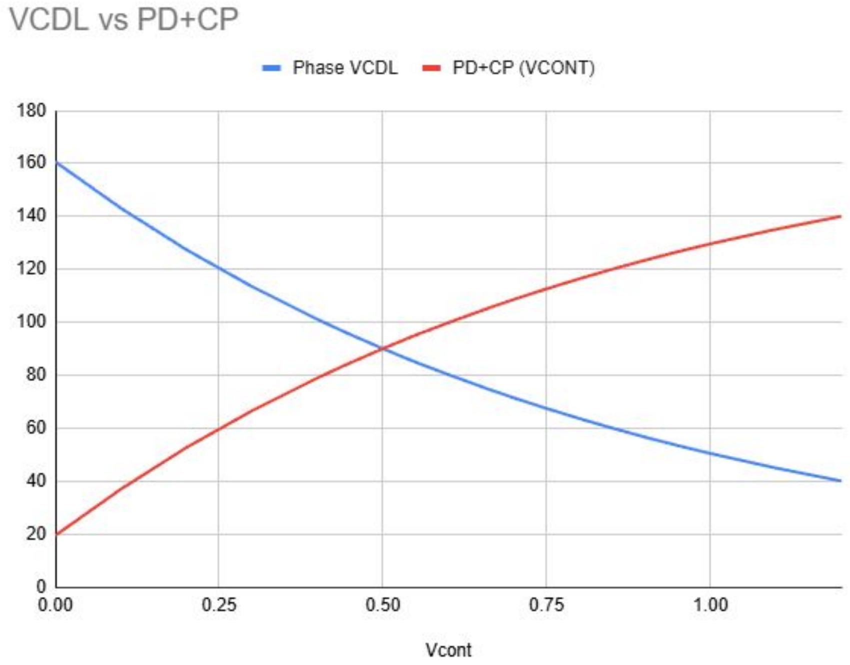
MODEL

```
.lib cornerMOSlv.lib mos_tt
*.lib cornerMOSlv.lib mos_ss
.lib cornerMOShv.lib mos_tt
.lib cornerRES.lib res_typ
.lib cornerCAP.lib cap_typ
.include diodes.lib
.include sg13g2_bondpad.lib
.include /foss/pdks/ihp-sg13g2/libs.ref/sg13g2_stdcell/spice/sg13g2_stdcell.spice
```

NGSPICE1

```
.control
tran 1p 10n
plot v(VIN) v(VOUT)
plot v(VCONT)
.endc
```

tb_open_loop.sch

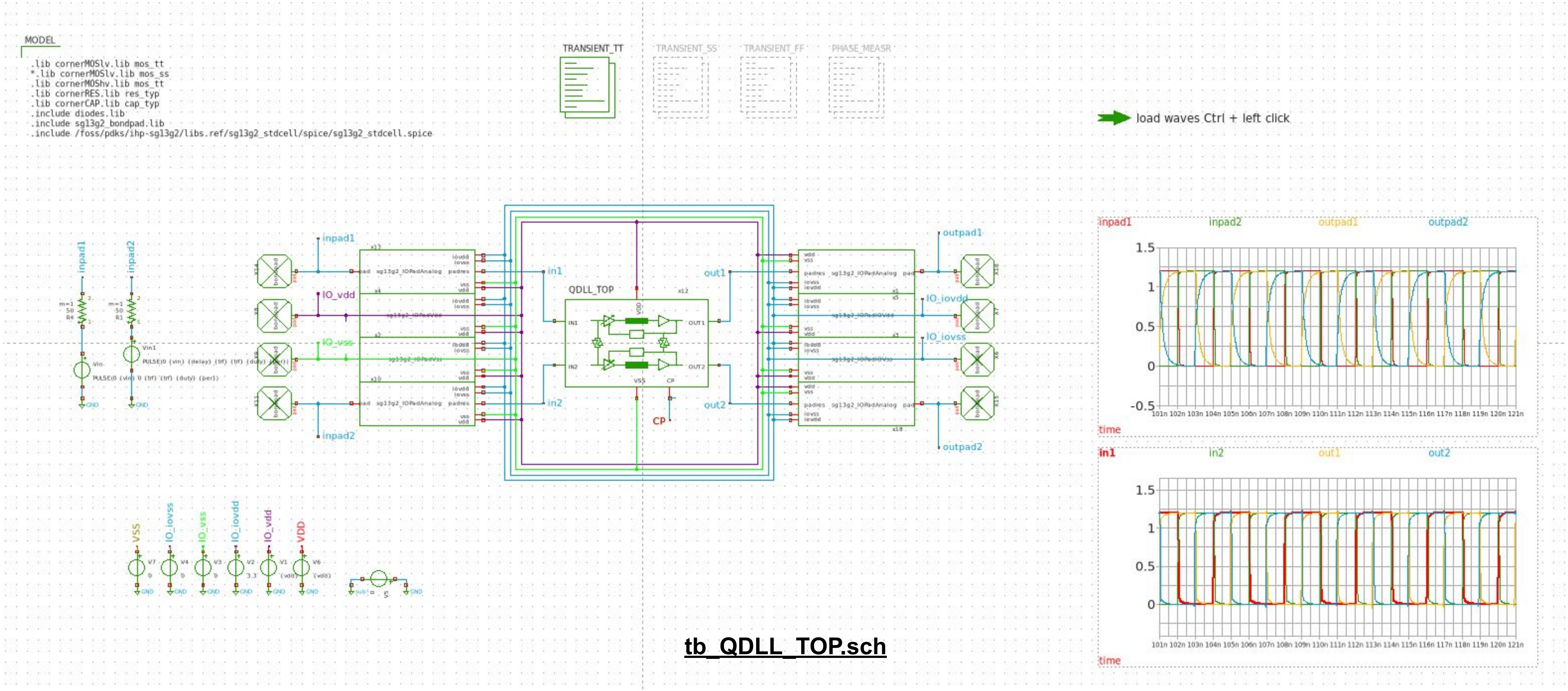


We can adjust the circuit in a way to obtain the desired intercept point → Fixed Phase Difference

5- QDLL TOP Testbench

5 - QDLL TOP Testbench (TT)

- TOP Testbench with the IOCELLS



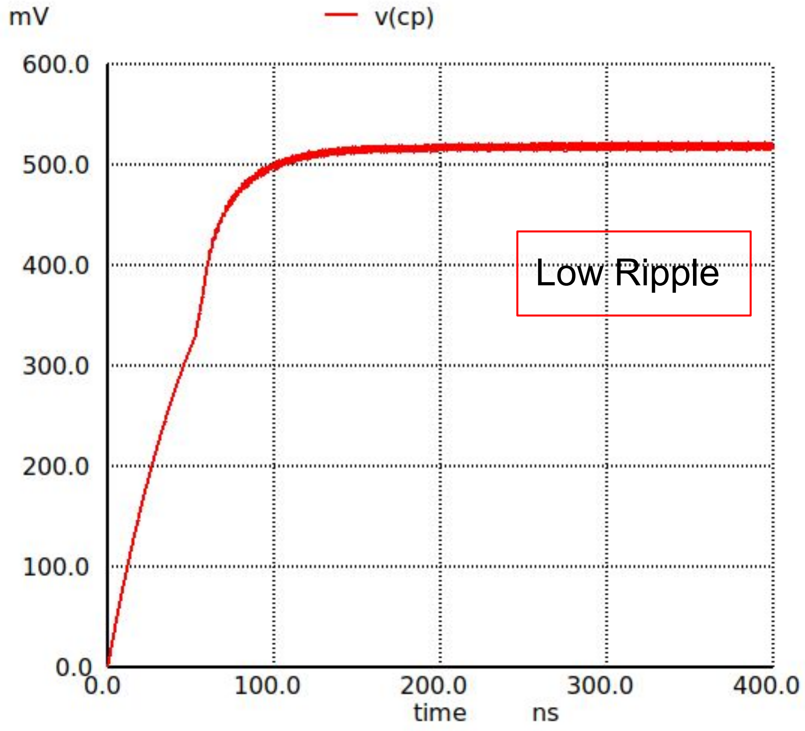
MODEL

```
.lib cornerMOSlv.lib mos_tt
*.lib cornerMOSlv.lib mos_ss
.lib cornerMOShv.lib mos_tt
.lib cornerRES.lib res_typ
.lib cornerCAP.lib cap_typ
.include diodes.lib
.include sg13g2_bondpad.lib
.include /foss/pdks/ihp-sg13g2/libs.ref/sg13g2_stdcell/spice/sg13g2_stdcell.spice
```

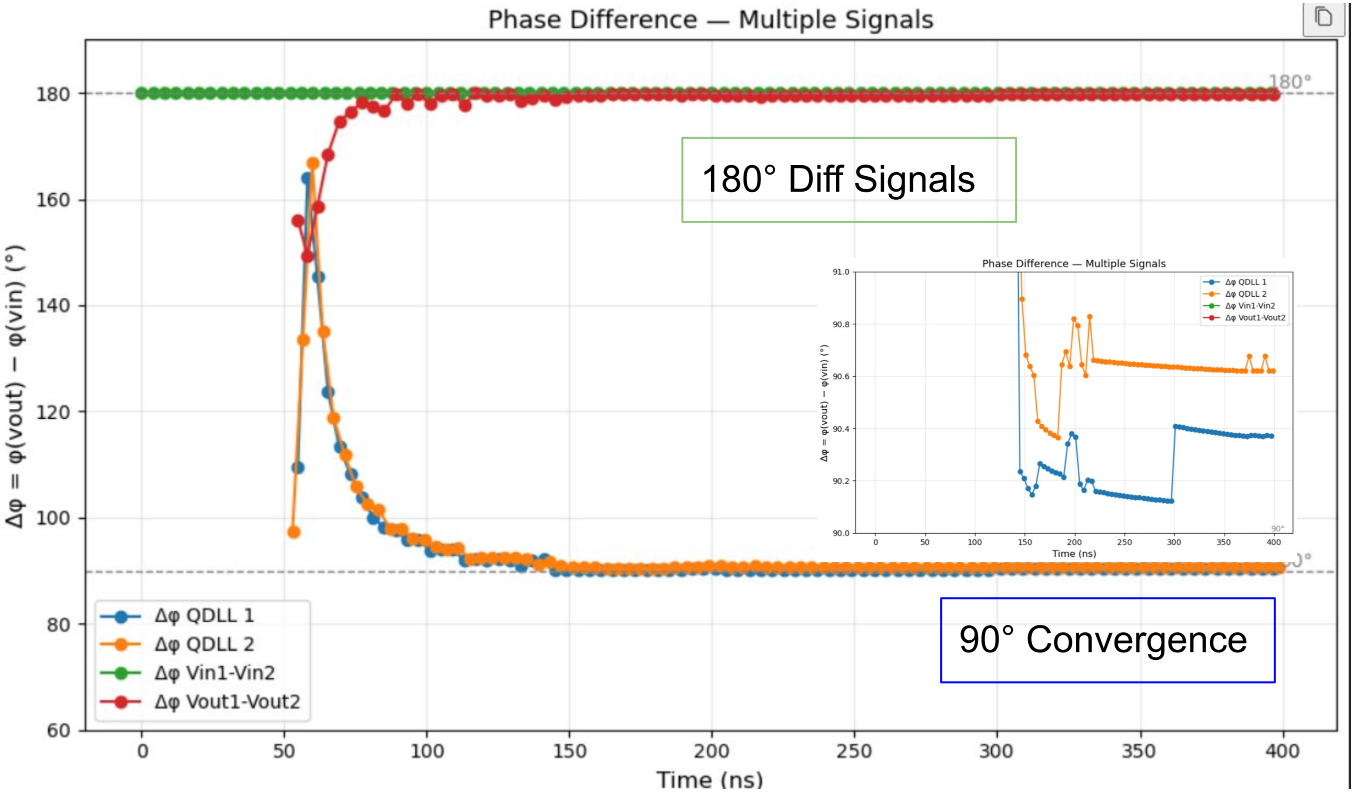
tb_QDLL_TOP.sch

5 - QDLL TOP Testbench (TT)

- Setup Time



Low Ripple



180° Diff Signals

90° Convergence

Vcont

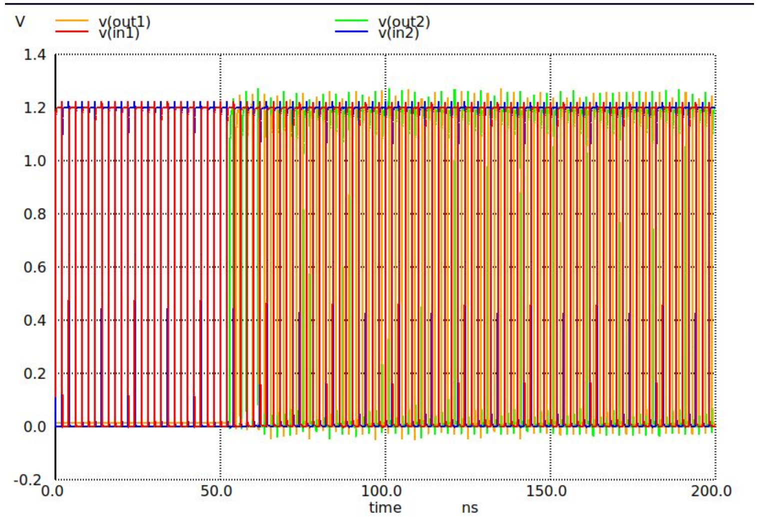
stabilises at approximately 100 ns in 515 mV mean

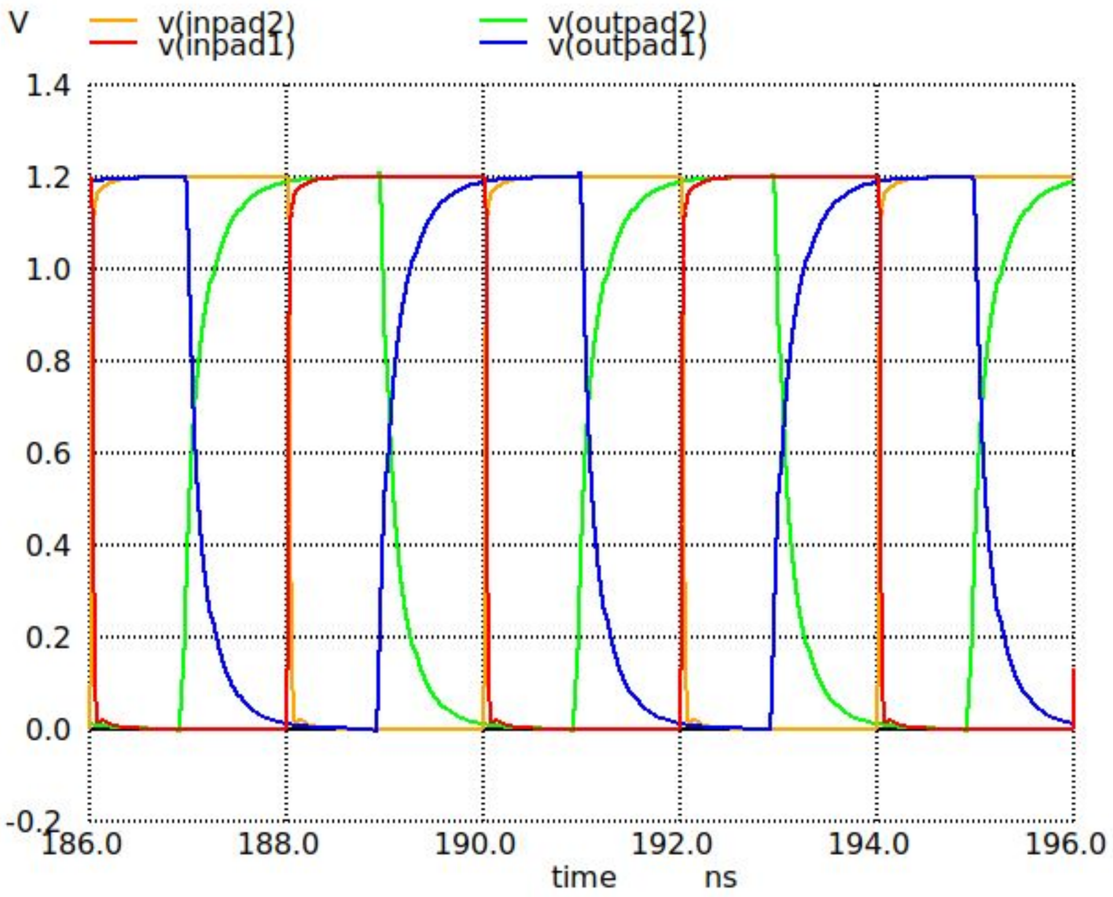
Python Plot

(You can use phase_analysis_example.ipynb)

5 - QDLL TOP Testbench (TT)

- Input/Output Signals





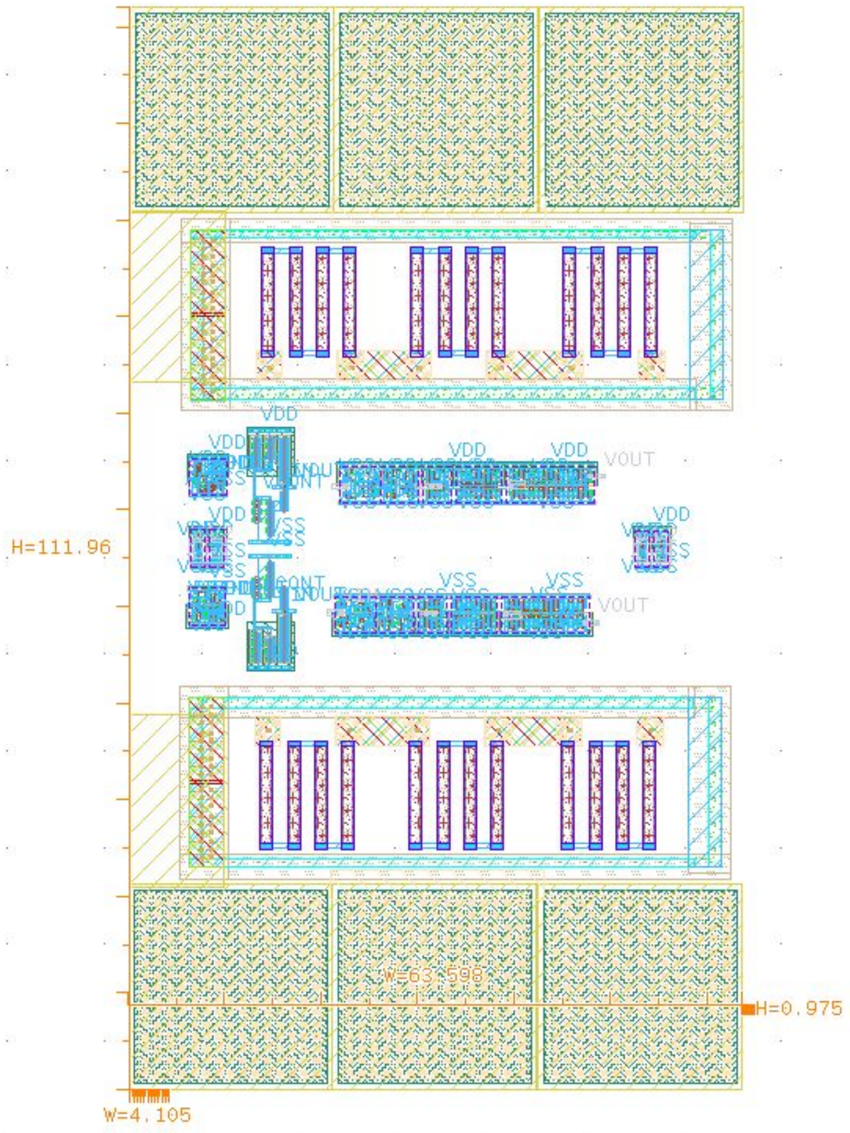
Phase DIff	v(inpad1)	v(inpad2)	v(outpad1)	v(outpad2)
v(inpad1)	-	180°	90.04°	270.04°
v(inpad2)	180°	-	270.06°	90.06°
v(outpad1)	90.04°	270.04°	-	180°
v(outpad2)	270.06°	90.06°	180°	-

Phase DIff	Trise	Tfall
v(outpad1)	400 ps	600 ps
v(outpad2)	400 ps	600 ps

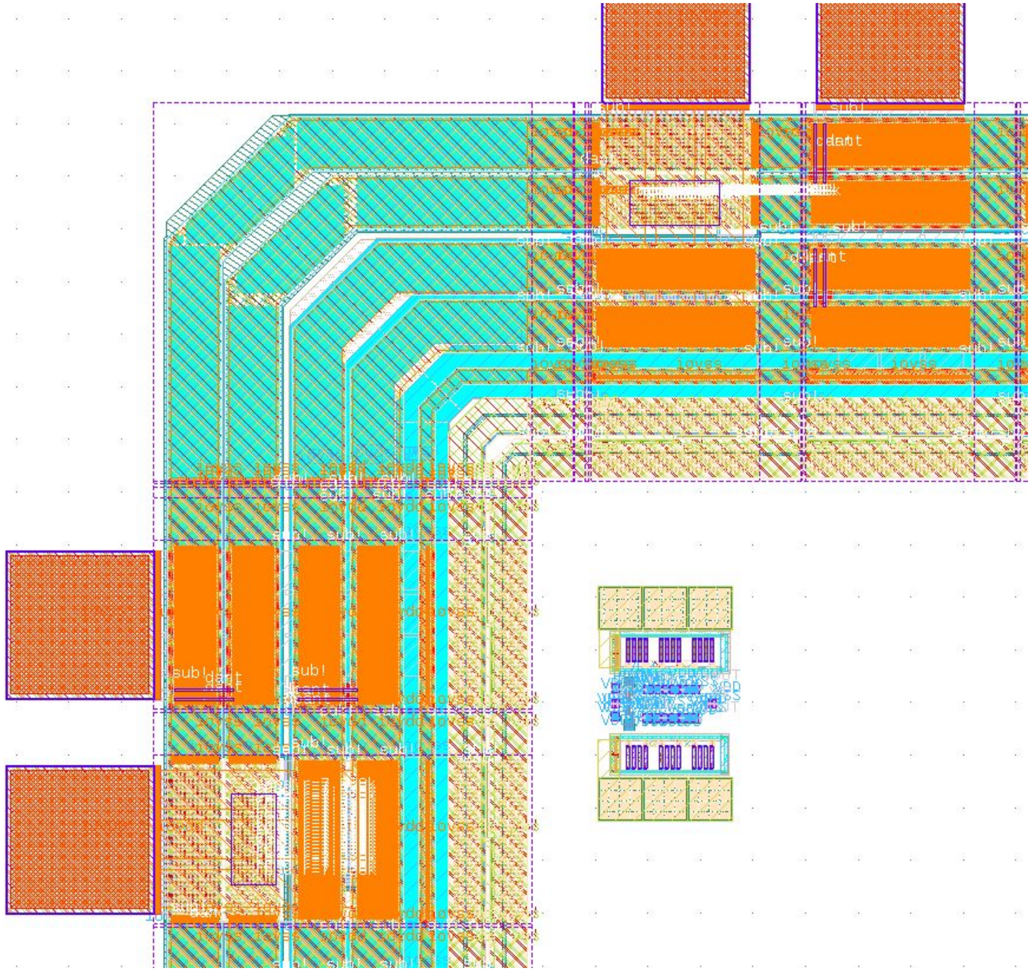
5- QDLL Layout TOP

QDLL Top Layout Placement

h = 111.96um



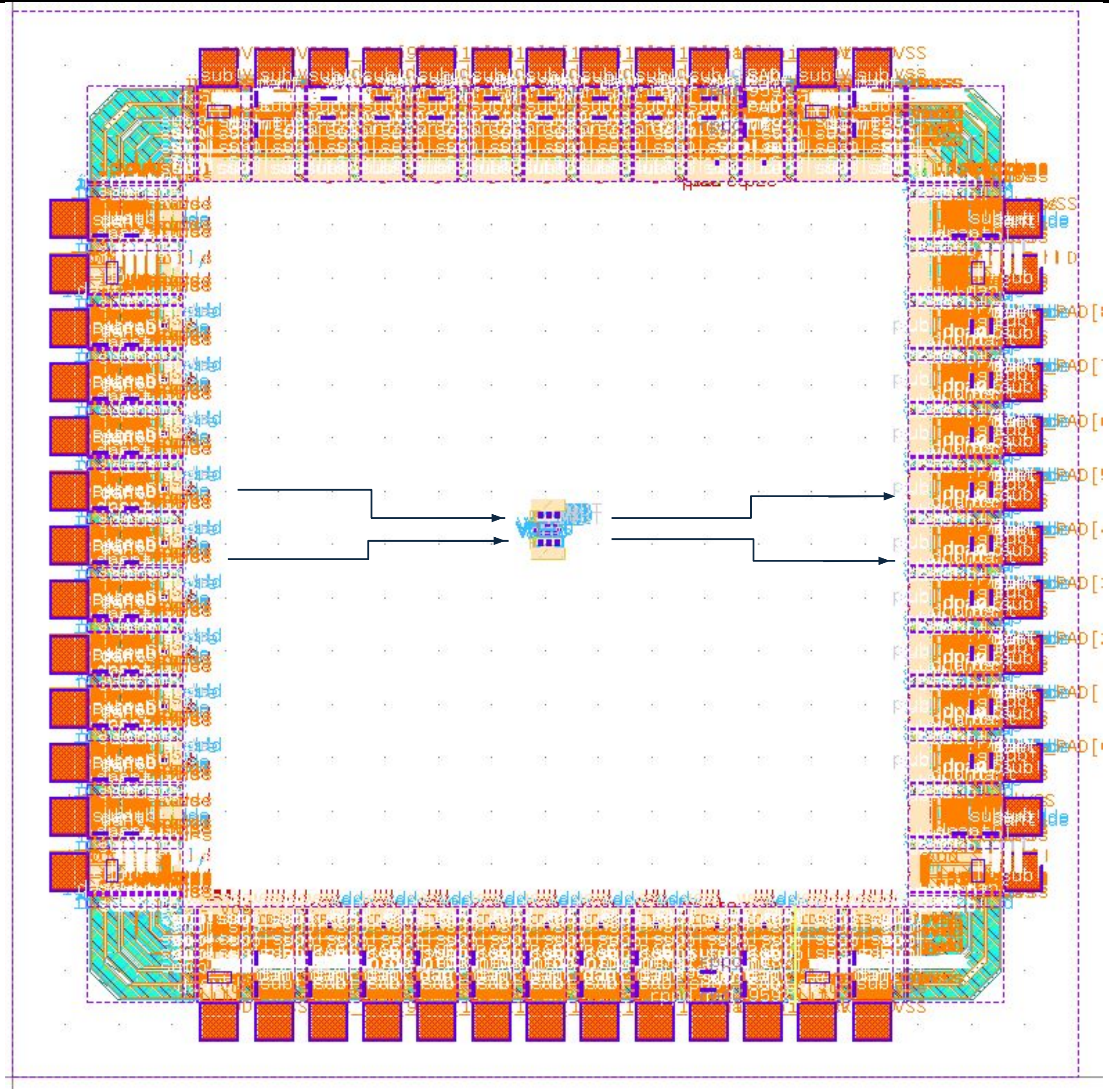
63.598um



Vertical View

Area: 111.96um x 63.598 um = 7120 um^2 =

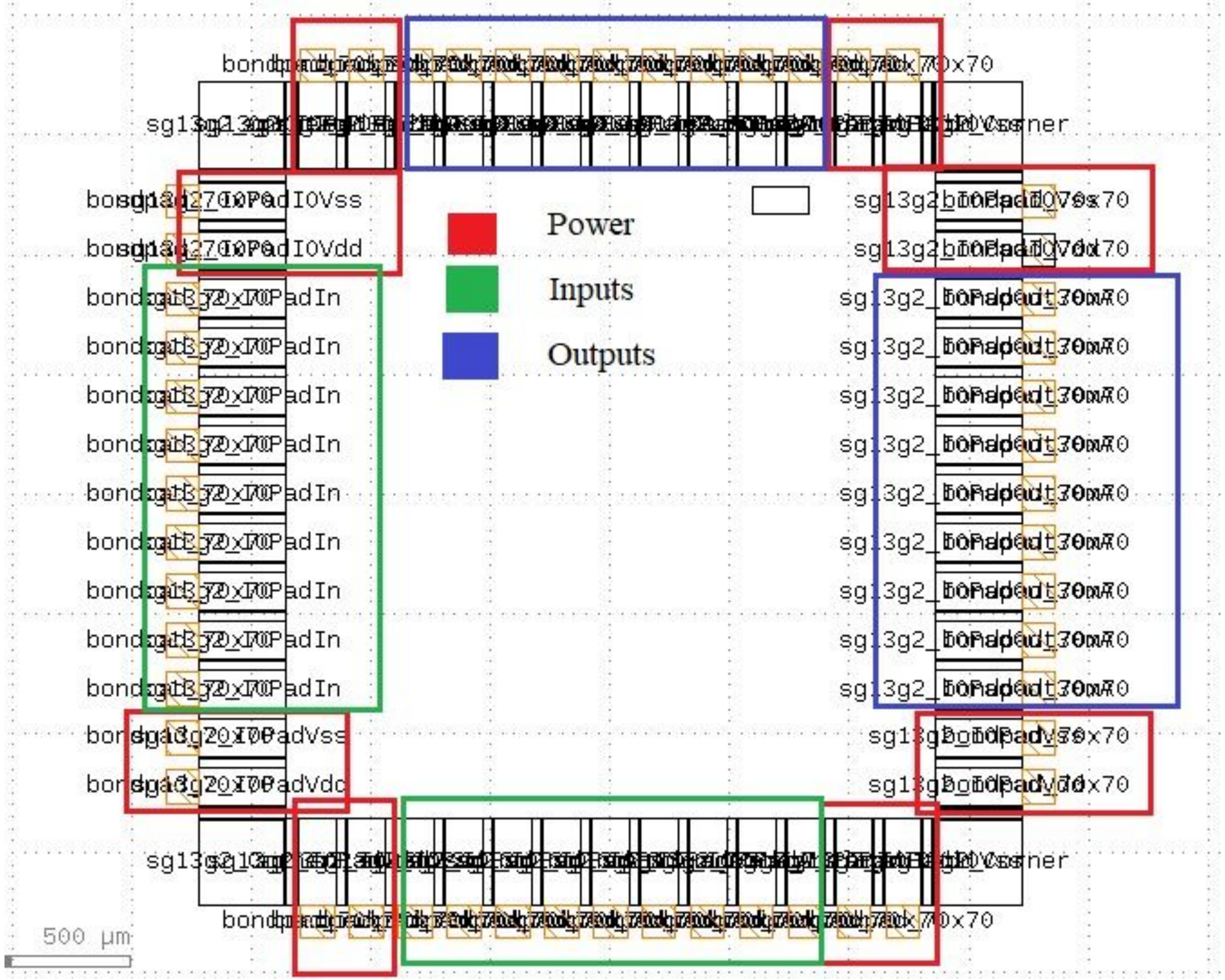
QDLL Top Layout Placement v1



One possible goal would be to place the QDLL vertically and connect the inputs on the left and the outputs on the right with high metal top routing.

For a 250 MHz signal, standard IO cells have been tested and found to perform well. However, if available, we prefer to interface with 50 Ω systems using IO cells with matched impedance.

QDLL layout within the pad ring



7 - QDLL Pins and IO CELLS

Pins and IO Cells

QDLL Pins

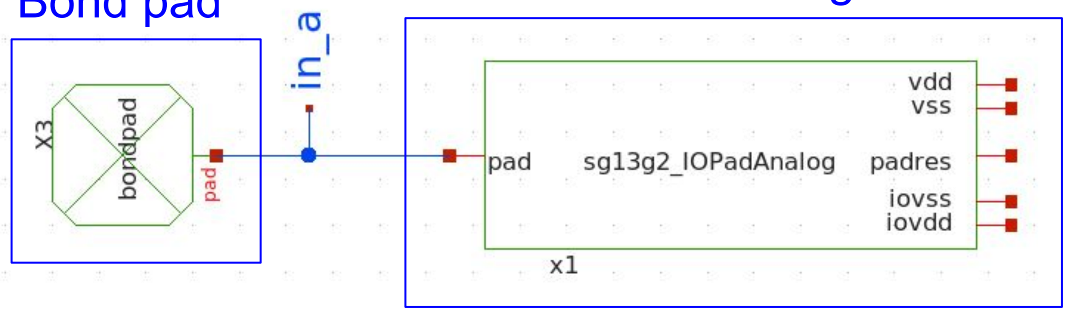
<u>N° Pins:</u> 4

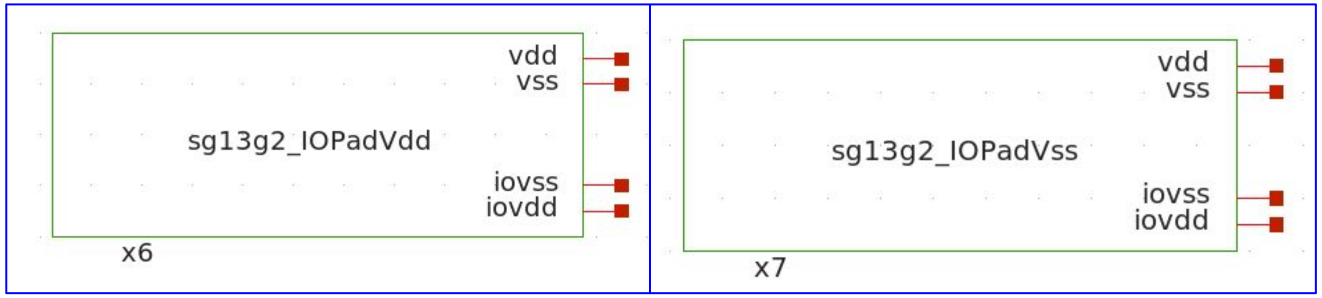
- 2 Input Pins
- 2 Output Pin
- 2 Power Supply Pins (VDD , VSS)

Pin Type	Name	IOCELL
Input	inpad1	sg13g2_IOPADAnalog
Input	inpad2	sg13g2_IOPADAnalog
Output	outpad1	sg13g2_IOPADAnalog
Output	outpad1	sg13g2_IOPADAnalog
Power Supply	VDD	sg13g2_IOPadVdd
Power Supply	VSS	sg13g2_IOPadVss

Bond pad

IOPadAnalog





Power Supply IOPads

8- QDLL Project Status

Project Status

Project Status by Block:

TOP	Sub-top	Sub-Block	Symbols & Schematics	Testbench	Prelayout results	IOCELLS results (TT)	IOCELLS results (SS-FF)	Layout (DRC + LVS)	Postlayout results
QDLL TOP	-	-	Done	Done	Done	Done	To do	WIP	To do
-	SE_QDLL	-	Done	Done	Done	Done	To do	WIP	To do
-	-	VCDL	Done	Done	Done	Done	W/P	WIP	To do
-	-	DLine	Done	Done	Done	Done	WIP	Done	To do
-	-	CP	Done	Done	Done	Done	WIP	WIP	To do
-	-	PD	Done	Done	Done	Done	WIP	Done	To do
-	LATCH	-	Done	Done	Done	Done	WIP	Done	To do

We are currently working on the DRC (minimum and maximum) verification of several cells to achieve the desired LVS verification. Next, we plan to compare the extracted views using Magic in order to obtain more realistic results.

On the other hand, multi-corner simulation results are desirable for our project, and we are actively working toward that goal.