inst.eecs.berkeley.edu/~eecs251b

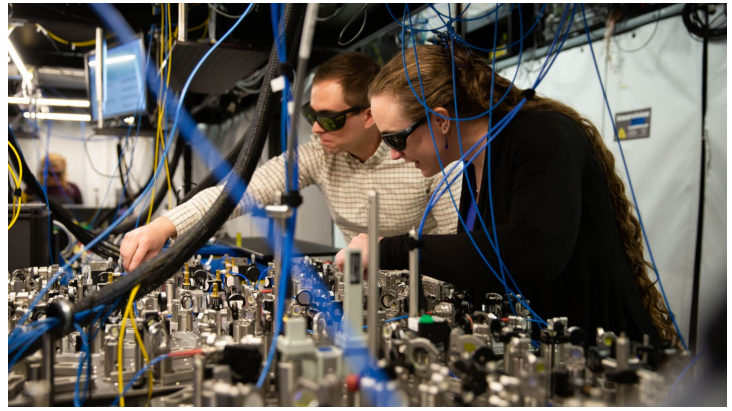
EECS251B : Advanced Digital Circuits and Systems

Lecture 21 – Low-Power Design

Borivoje Nikolić

Advancing science: Microsoft and Quantinuum demonstrate the most reliable logical qubits on record with an error rate 800x better than physical qubits

April 3, 2024. Today signifies a major achievement for the entire quantum ecosystem: Microsoft and Quantinuum demonstrated the most reliable logical qubits on record. By applying Microsoft's breakthrough qubit-virtualization system, with error diagnostics and correction, to Quantinuum's ion-trap hardware, we ran more than 14,000 individual experiments without a single error. Furthermore, we demonstrated more reliable quantum computation by performing error diagnostics and corrections on logical qubits without destroying them. This finally moves us out of the current noisy intermediate-scale quantum (NISQ) level to Level 2 Resilient quantum computing.



Quantinuum scientists making adjustments to a beam line array used to deliver laser pulses in H-Series quantum computers. Photo courtesy of Quantinuum.

Announcements

- Quiz 3 is today
- Homework 4 released
- Project
  - Preliminary design review next Tuesday
  - Starting at 9am, so everyone can present
  - 7 minutes per team
- Lab 5 due this week

Architectural Power-Performance Tradeoffs

Optimal Processors

- Processors used to be optimized for performance
  - Optimal logic depth was found to be 8-11 FO4 delays in superscalar processors
  - 1.8-3 FO4 in sequentials, rest in combinatorial
    - Kunkel, Smith, ISCA'86
    - Hriskesh, Jouppi, Farkas, Burger, Keckler, Shivakumar, ISCA'02
    - Harstein, Puzak, ISCA'02
    - Sprangle, Carmean, ISCA'02
- But those designs have very high power dissipation
  - Need to optimize for both performance and power/energy

From System View: What is the Optimum?

- How do sensitivities relate to more traditional metrics:
  - Power per operation (MIPS/W, GOPS/W, TOPS/W)
  - Energy per operation (Joules per op)
  - Energy-delay product
- Can be reformatted as a goal of optimizing power x delay^n
  - n = 0 – minimize power per operation
  - n = 1 – minimize energy per operation
  - n = 2 – minimize energy-delay product
  - n = 3 – minimize energy-(delay)^2 product

Optimization Problem

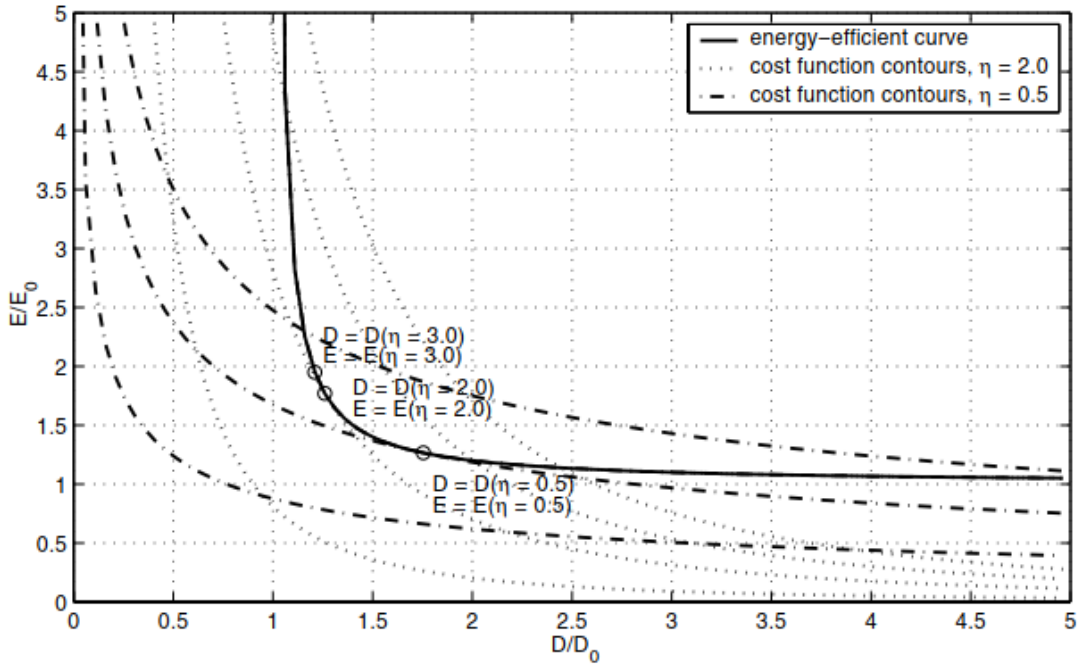
- Set up optimization problem:
  - Maximize performance under energy constraints
  - Minimize energy under performance constraints
- Or minimize a composite function of E^mD^n
  - What are the right m and n?
- $m = 1, n = 1$ is EDP – improves at lower V_{DD}
- $m = 1, n = 2$ is invariant to V_{DD}
  - $E \sim CV_{DD}^2$
  - $D \sim 1/V_{DD}$

Hardware Intensity

- Introduced by Zyuban and Strenski in 2002.
- Measures where is the design on the Energy-Delay curve
- Parameter in cost function optimization

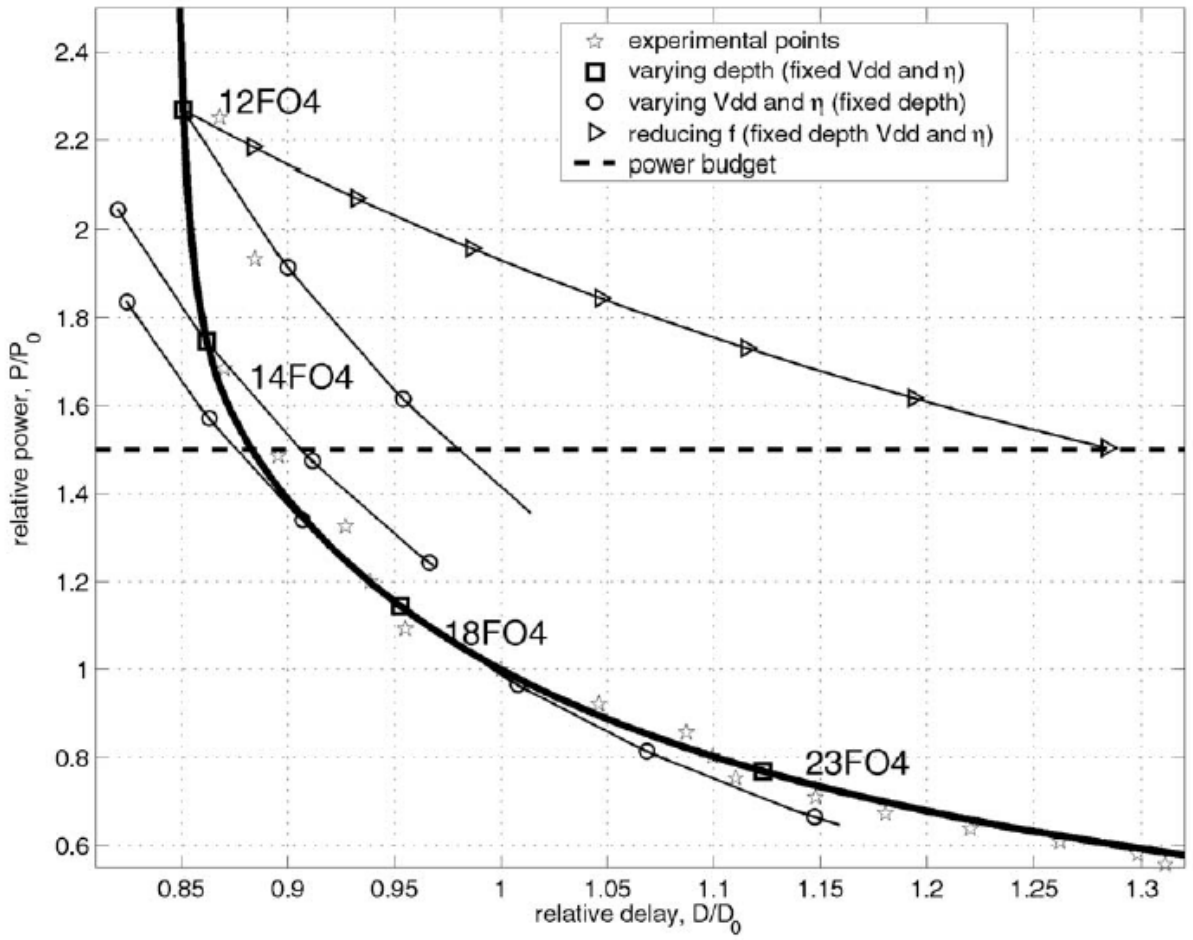
$$F_c = (E/E_0)(D/D_0)^{\eta} \qquad 0 \le \eta < +\infty,$$

$$\eta = -\left.\frac{D\partial E}{E\partial D}\right|_v$$



Slope of the optimal E-D curve at the chosen design point

Optimum Across Hierarchy Layers



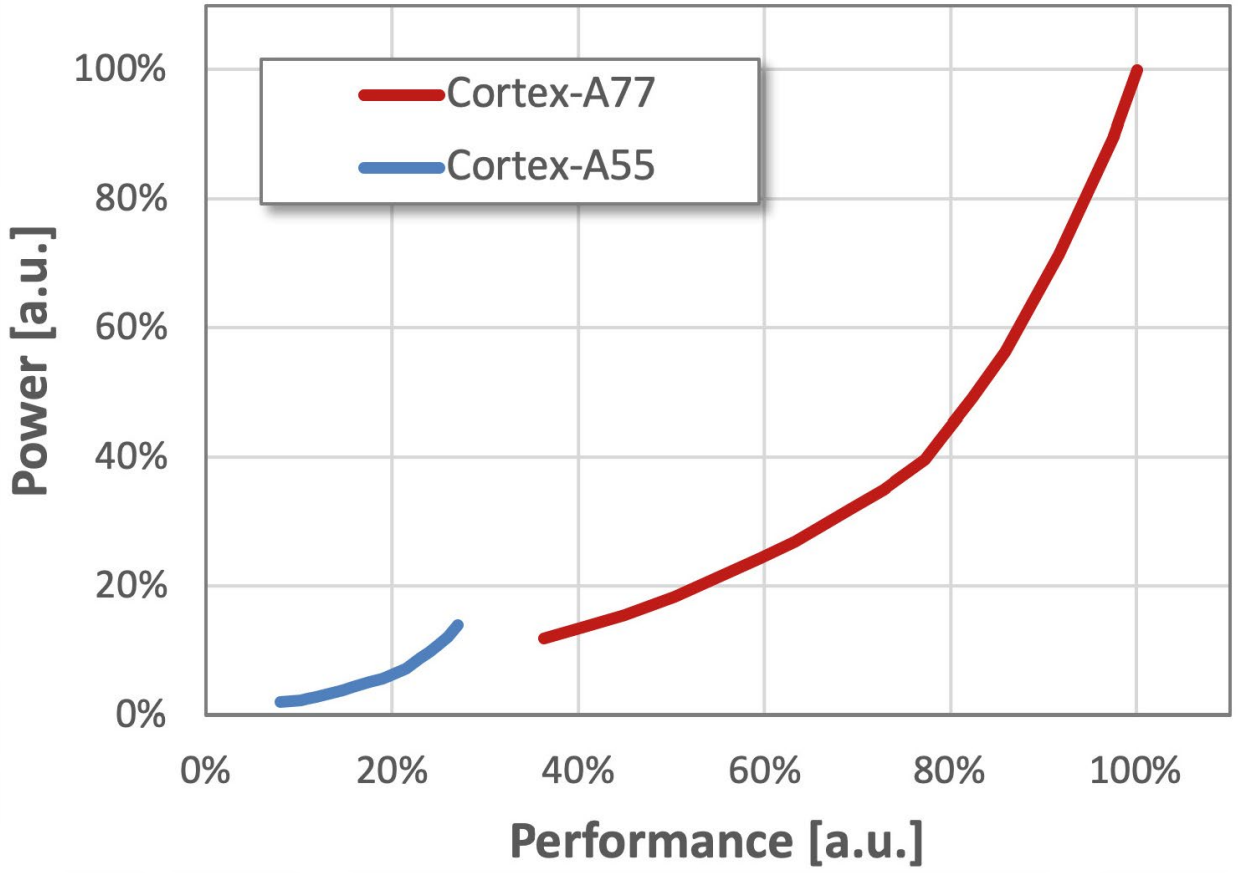
Zyuban et al, TComp'04

Optimal logic depth in pipelined processors is ~18FO4

Relatively flat in the 16-22FO4 range

Architectural Tradeoffs

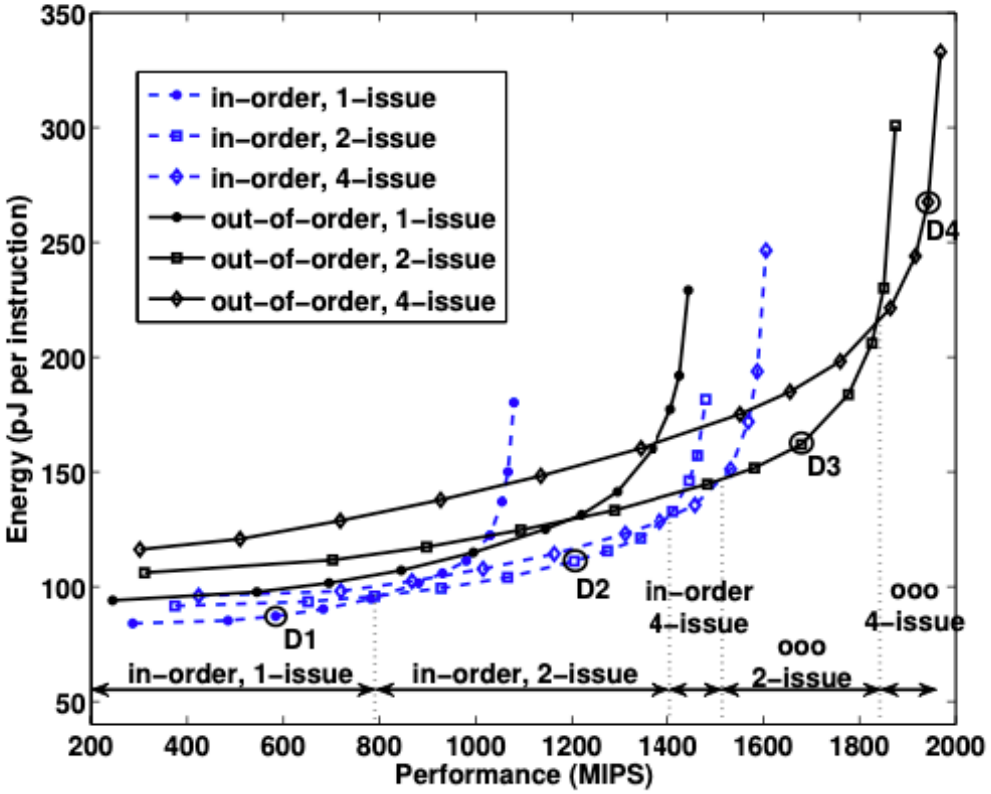
- H, Mair, ISSCC'20



Energy-Delay Tradeoff of Modern Processors

Table 3: *Design Configuration Details For Selected Design Points.*

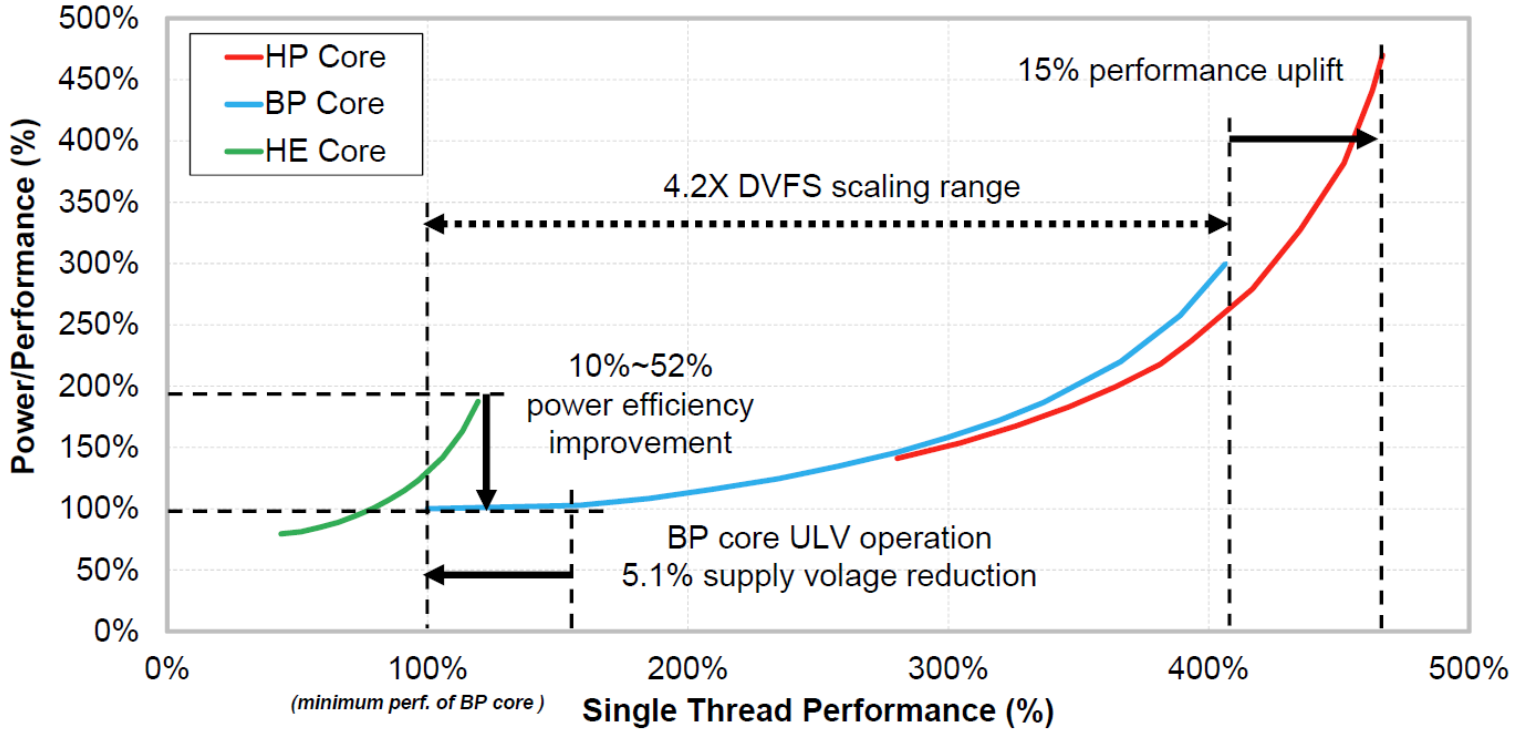
	D1	D2	D3	D4
In-order vs out-of-order	in-order	in-order	out-of-order	out-of-order
Issue width	1-issue	2-issue	2-issue	4-issue
Cycle time (FO4)	27.5	16.9	17.2	16.3
Branch pred size (entries)	264	600	1024	870
BTB size (entries)	64	90	554	1024
I-cache size (KB)	21	32	32	32
D-cache size (KB)	8	11	14	42
Fetch latency	1.0	1.6	2.2	2.1
Decode/Rename latency	1.0	1.7	2.4	3.0
Retire latency	N/A	N/A	2.0	2.2
Integer ALU latency	1.0	1.0	1.0	1.0
FP ALU latency	3.0	4.0	3.9	4.1
L1 D-cache latency	1.0	1.1	1.1	1.1
ROB size	N/A	N/A	22	32
IW size	N/A	N/A	11	9
LSQ size	N/A	N/A	16	16



Azizi, ISCA'10

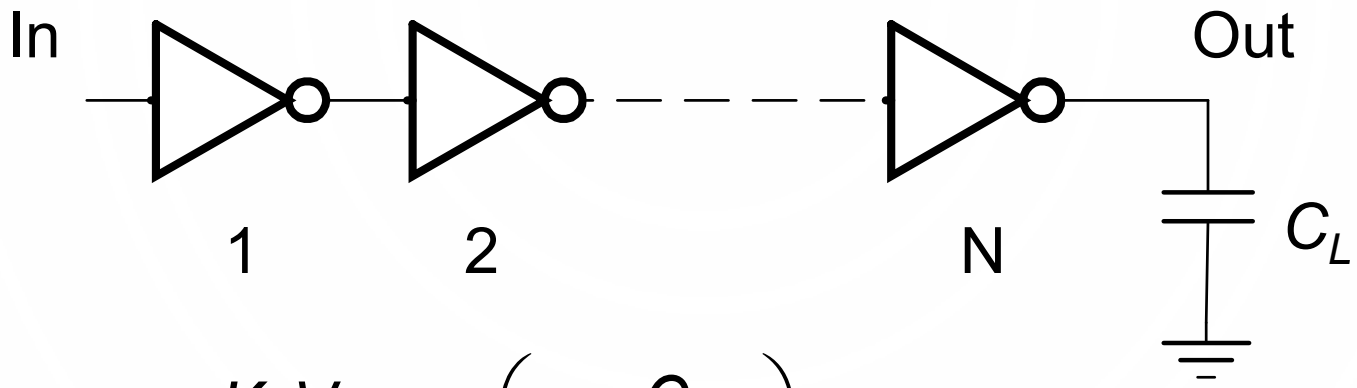
Architectural Tradeoffs: Tri-Gear

- HP: High performance (ARM Cortex A78, optimized for speed, 3.0GHz)
- BP: Balanced performance (ARM Cortex A78, optimized for power, 2.6GHz))
- HE: High efficiency (ARM A55, 2.0GHz)



Circuit-Level Tradeoffs

Alpha-Power Based Delay Model



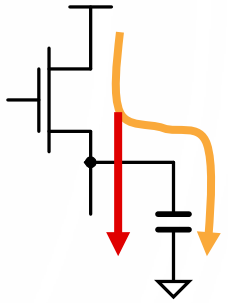
$$t_{pi} = \frac{K_d V_{DD}}{(V_{DD} - V_{Th})^{\alpha}} \left(1 + \frac{C_{L,i}}{C_{in,i}}\right)$$

$$D = \sum t_{pi} = \sum \frac{K_d V_{DD}}{(V_{DD} - V_{Th})^{\alpha}} \left(1 + \frac{W_{L,i}}{W_{in,i}}\right)$$

Energy Models

- **Switching**

$$E_{Sw} = \alpha_{0\to1}\left(C_{L,i} + C_{\text{int},i}\right)V_{DD}^{\,2}$$



- **Leakage**

$$E_{Lk} = W_{In} I_0 e^{\frac{-(V_{Th}-\gamma V_{DD})}{nV_t}} V_{DD} D$$

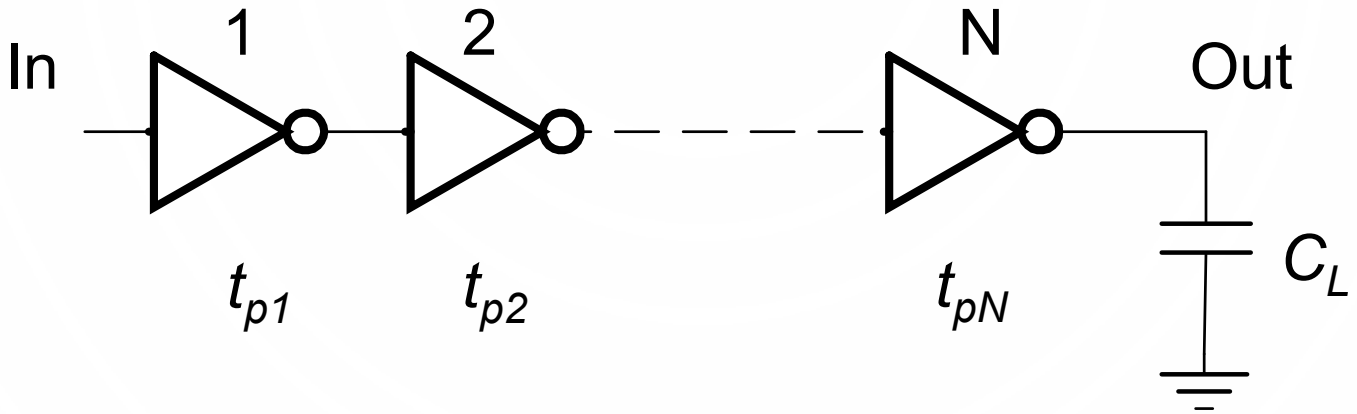
Sizing, Supply, Threshold Optimization

- Transistor sizing can yield large power savings with small delay penalties
  - Gate sizing
  - Beta-ratio adjustments
  - (Stack resizing)

$$\beta = Wp/Wn$$

- Supply voltage affects both active and leakage energy
- Threshold voltage affects primarily the leakage

Apply to Sizing of an Inverter Chain



Unconstrained energy: find min D = Σt_{pi}

$$C_{gin,j} = \sqrt{C_{gin,j-1} C_{gin,j+1}} \qquad W_j = \sqrt{W_{j-1} W_{j+1}}$$

Constrained energy: find min D, under $E < E_{max}$

Where $E = \Sigma E_i$

Constrained Optimization

- Find min(D) subject to $E = E_{max}$
  - *Constrained function minimization*
- E.g. Lagrange multipliers

$$\Lambda(x) = D(x) + \lambda(E(x) - E_{\max})$$

$$\frac{\partial \Lambda}{\partial x} = 0$$

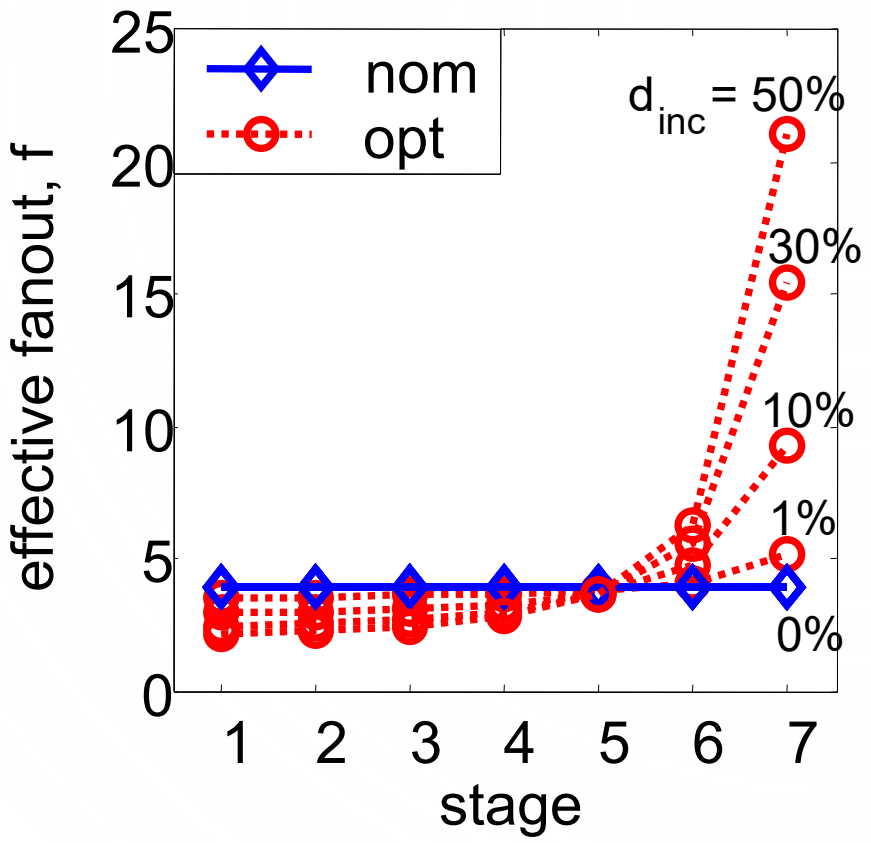
Or dual:

$$\mathrm{K}(x) = E(x) + \lambda(D - D_{\max})$$

- Can solve analytically for $x = W_i$, V_{DD}, V_{Th}

Inverter Chain: Sizing Optimization

Inverter Chain: Sizing Optimization



$$W_j = \sqrt{\frac{W_{j-1}W_{j+1}}{1+\lambda W_{j-1}}}$$

[Ma, Franzon, *IEEE JSSC*, 9/94]

$$\lambda = -\frac{2KV_{DD}^{\ 2}}{\tau_{nom}S_W}$$

$$S_W \propto \frac{e_j}{f_j - f_{j-1}}$$

e_j – energy per stage
f_j – fanout per stage

Stojanovic, ICCAD'02

- Variable taper achieves minimum energy
- Reduce number of stages at large d_{inc}

Sensitivity to Sizing and Supply

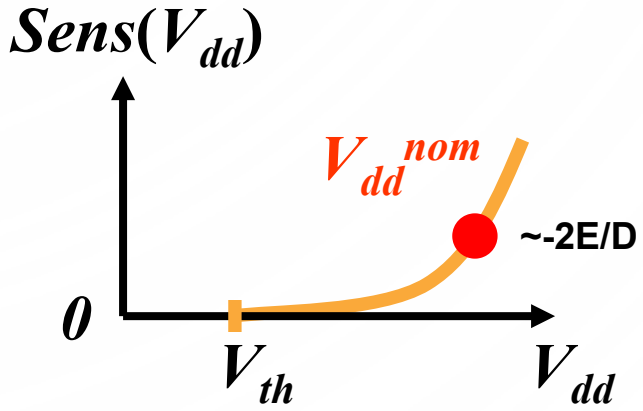
- Gate sizing (W_i)

$$-\frac{\partial E_{sw} / \partial W_j}{\partial D / \partial W_j} = \frac{e_j}{\tau_{nom}\left(f_j - f_{j-1}\right)}$$

∞ for equal f_{eff} (D_{min})

- Supply voltage (V_{dd})

$$-\frac{\partial E_{sw} / \partial V_{DD}}{\partial D / \partial V_{DD}} = \frac{E_{sw}}{D} 2 \frac{1 - x_v}{\alpha - 1 + x_v}$$



$x_v = (V_{Th} + \Delta V_{Th})/V_{dd}$

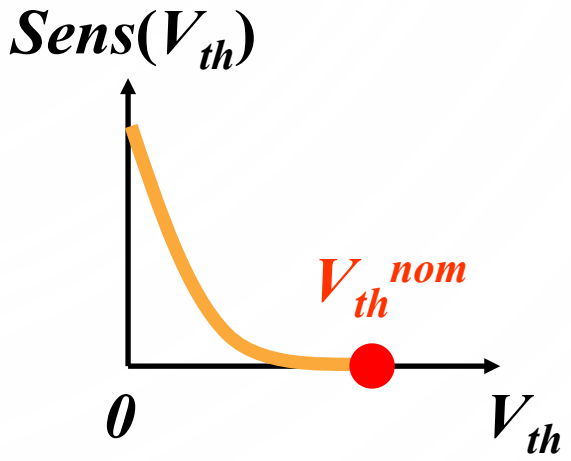
Sensitivity to V_{th}

- Threshold voltage (V_{th})

$$-\frac{\partial E / \partial \Delta V_{Th}}{\partial D / \partial \Delta V_{th}} = P_{Lk}\left(\frac{V_{DD} - V_{Th} - \Delta V_{Th}}{\alpha n V_t} - 1\right)$$

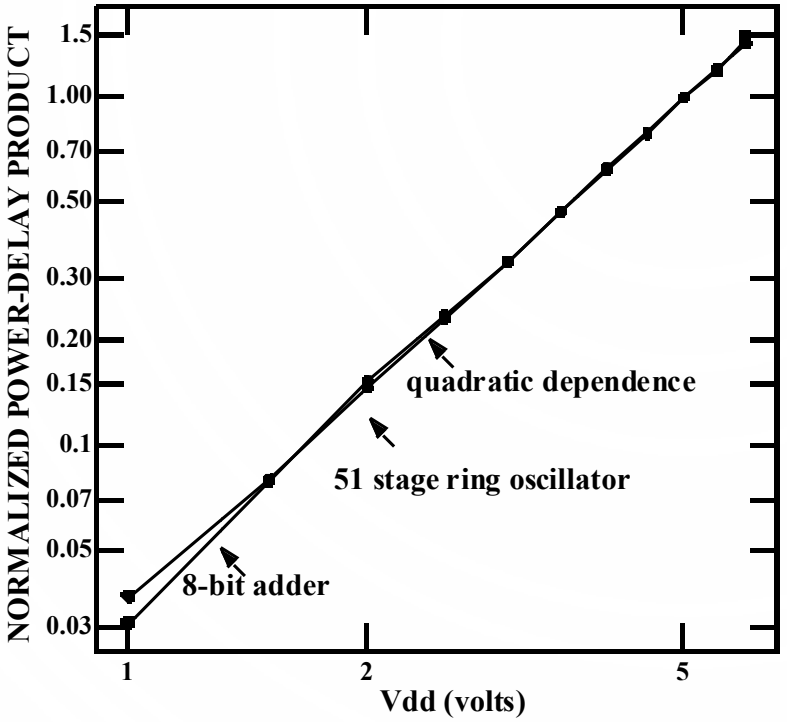
Low initial leakage

$\Rightarrow$ speedup comes for "free"



Scaling Supplies

Reducing V_{dd}



$$P \times t_d = E_t = C_L * V_{dd}^2$$

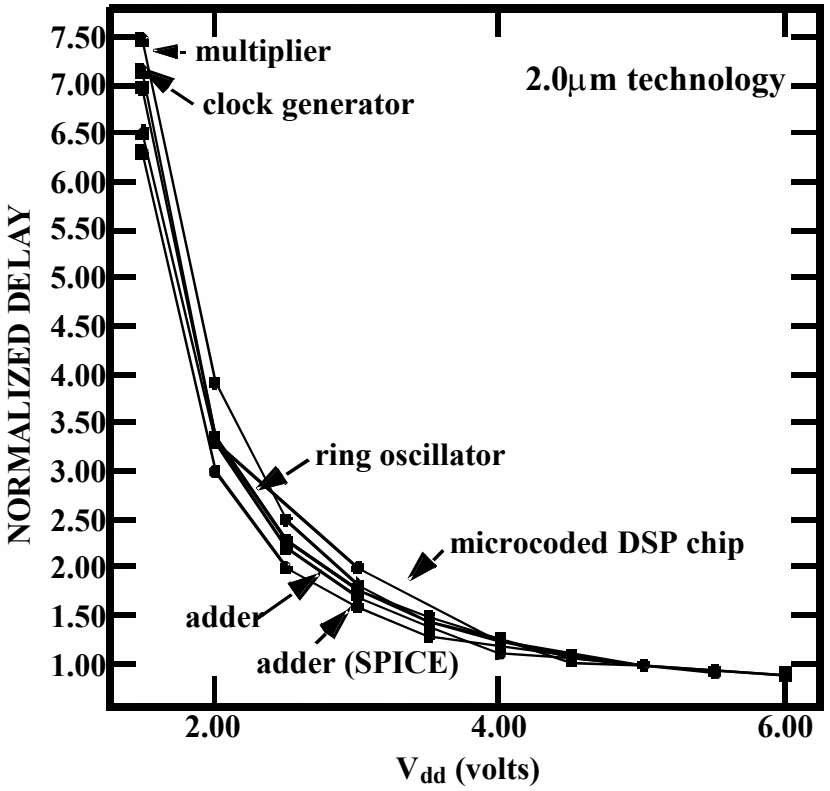
$$\frac{E_{(Vdd=2)}}{E_{(Vdd=5)}} = \frac{(C_L) * (2)^2}{(C_L) * (5)^2}$$

$$E_{(Vdd=2)} \approx 0.16\ E_{(Vdd=5)}$$

- Strong function of voltage (V^2 dependence).
- Relatively independent of logic function and style.
- Power Delay Product Improves with lowering V_{DD}.

Chandrakasan, JSSC'92

Lower V_{DD} Increases Delay



$$T_d = \frac{C_L * V_{dd}}{I}$$

$$I \sim (V_{dd} - V_t)^2$$

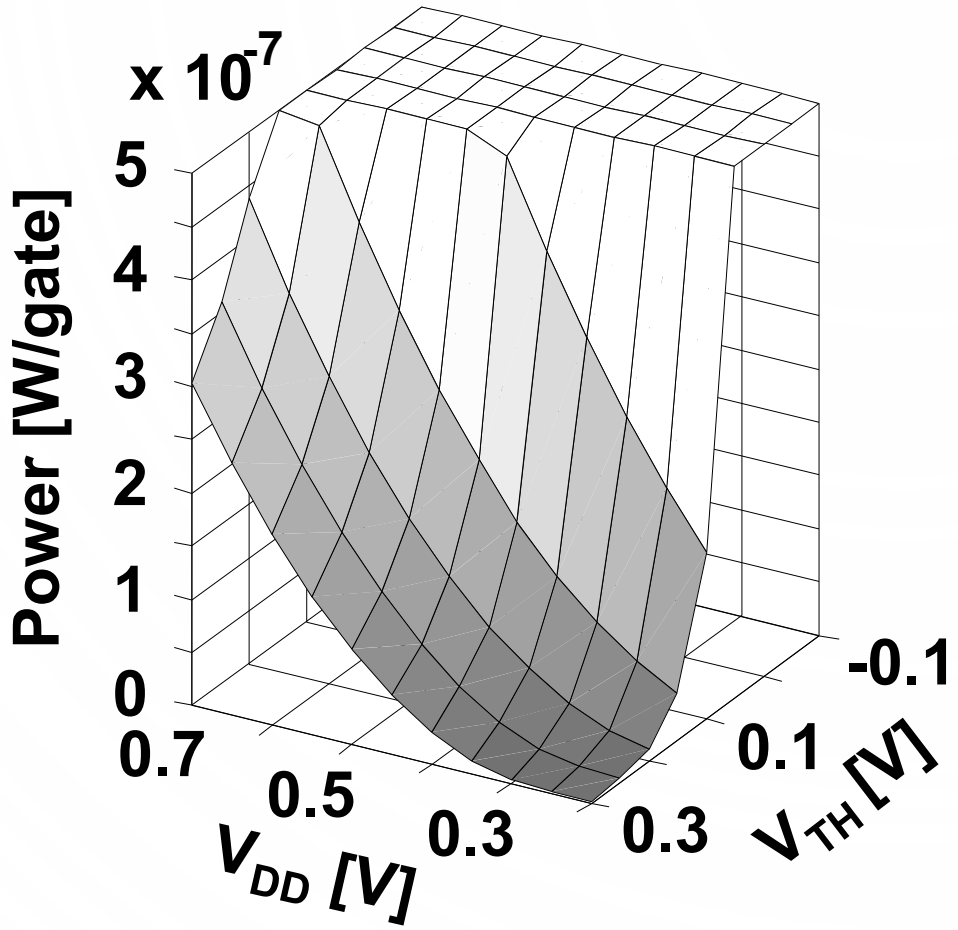
$$\frac{T_{d(Vdd=2)}}{T_{d(Vdd=5)}} = \frac{(2) * (5 - 0.7)^2}{(5) * (2 - 0.7)^2} \approx 4$$

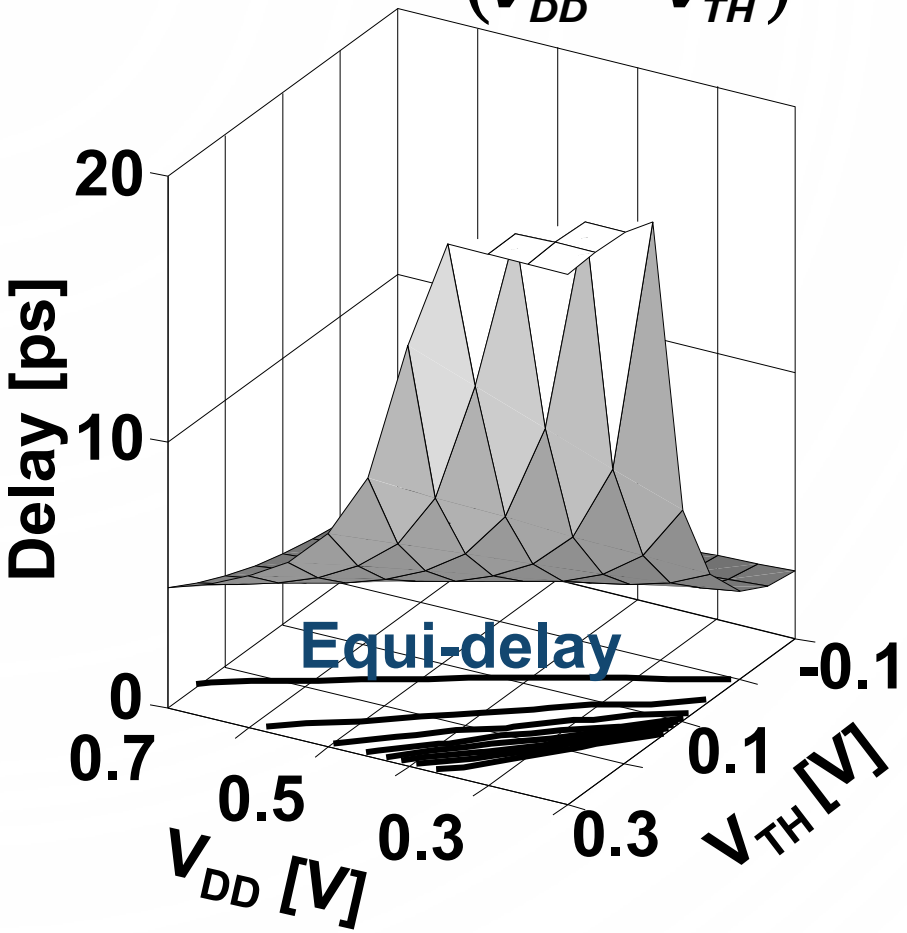
- **Relatively independent of logic function and style.**

Trade-off Between Power and Delay

$$Power = a \cdot f \cdot C \cdot V_{DD}^2 + I_0 \cdot 10^{-\frac{V_{TH}}{s}} \cdot V_{DD}$$

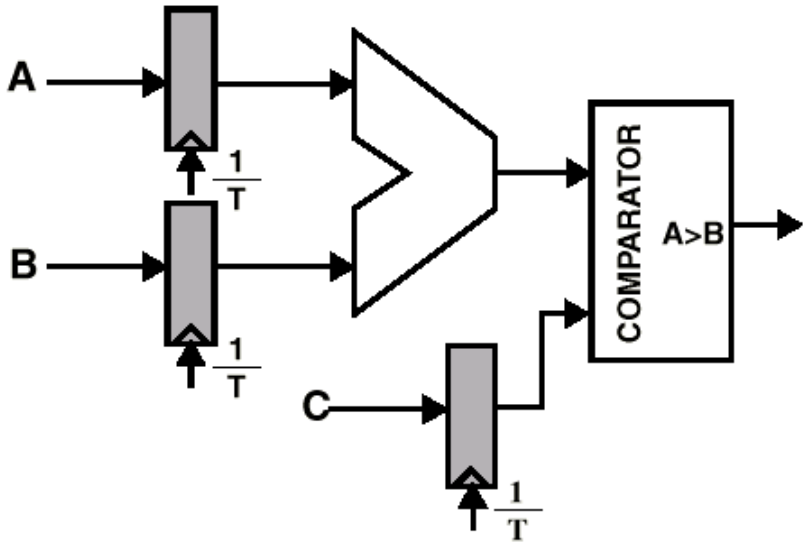
$$Delay \propto \frac{C \cdot V_{DD}}{(V_{DD} - V_{TH})^{1.3}}$$

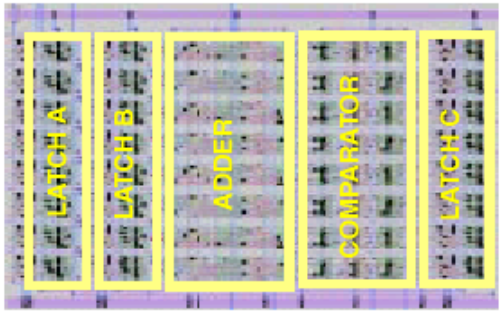




50nm node, FO3 INV

Architecture Trade-off for Fixed-rate Processing Reference Datapath



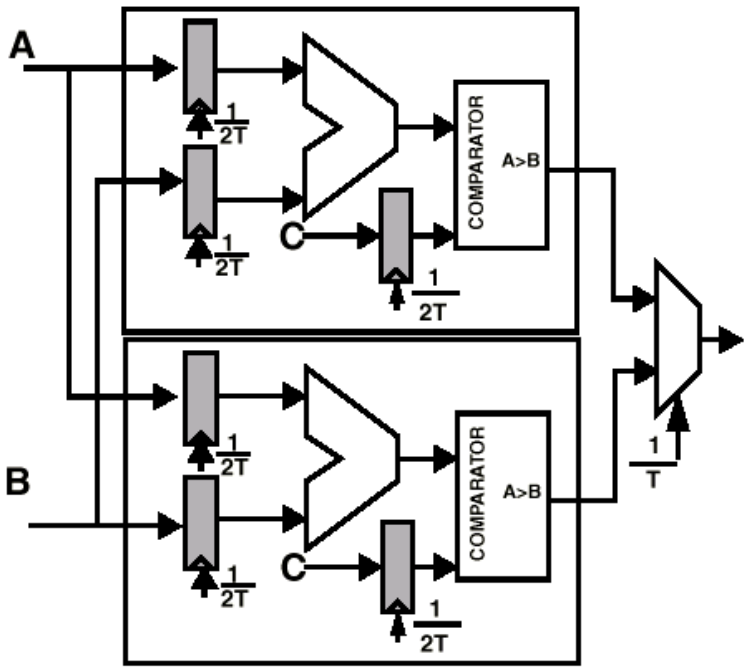


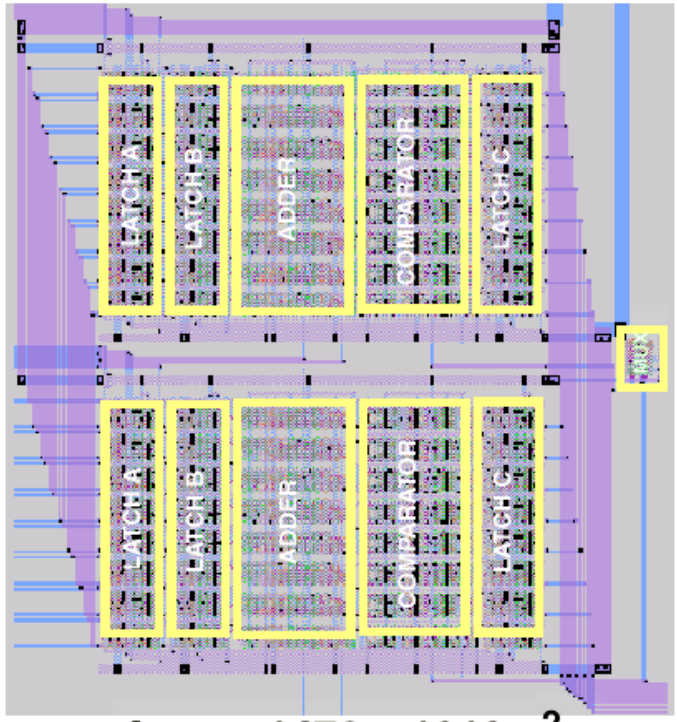
Area = 636 x 833 μ^2

- Critical path delay $\Rightarrow T_{adder} + T_{comparator}$ (= 25ns)
  $\Rightarrow f_{ref}$ = 40Mhz
- Total capacitance being switched = C_{ref}
- $V_{dd} = V_{ref} = 5V$
- Power for reference datapath = $P_{ref} = C_{ref} V_{ref}^2 f_{ref}$

from [Chandrakasan92] (*IEEE JSSC*)

Parallel Datapath

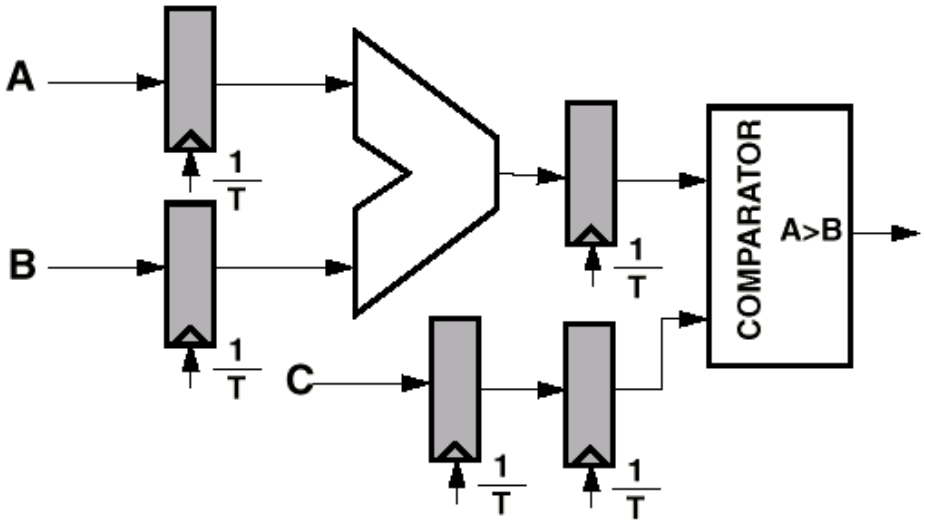


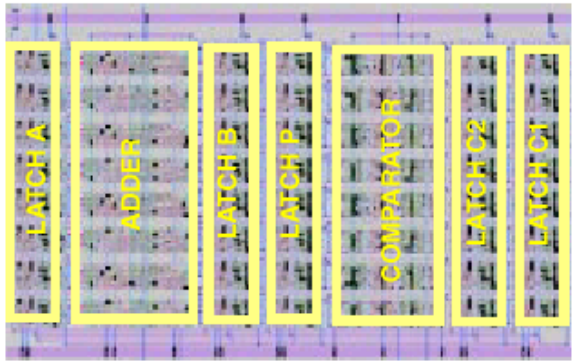


Area = 1476 x 1219 μ^2

- The clock rate can be reduced by half with the same throughput $\Rightarrow f_{par} = f_{ref} / 2$
- $V_{par} = V_{ref} / 1.7$, $C_{par} = 2.15C_{ref}$
- $P_{par} = (2.15C_{ref})\ (V_{ref}/1.7)^2\ (f_{ref}/2) \approx 0.36\ P_{ref}$

Pipelined Datapath





Area = 640 x 1081 μ^2

- Critical path delay is less $\Rightarrow$ max [T_{adder} , $T_{comparator}$]
- Keeping clock rate constant: $f_{pipe} = f_{ref}$
  Voltage can be dropped $\Rightarrow$ $V_{pipe} = V_{ref} / 1.7$
- Capacitance slightly higher: $C_{pipe} = 1.15C_{ref}$
- $P_{pipe} = (1.15C_{ref})\ (V_{ref}/1.7)^2\ f_{ref} \approx 0.39\ P_{ref}$

A Simple Datapath: Summary

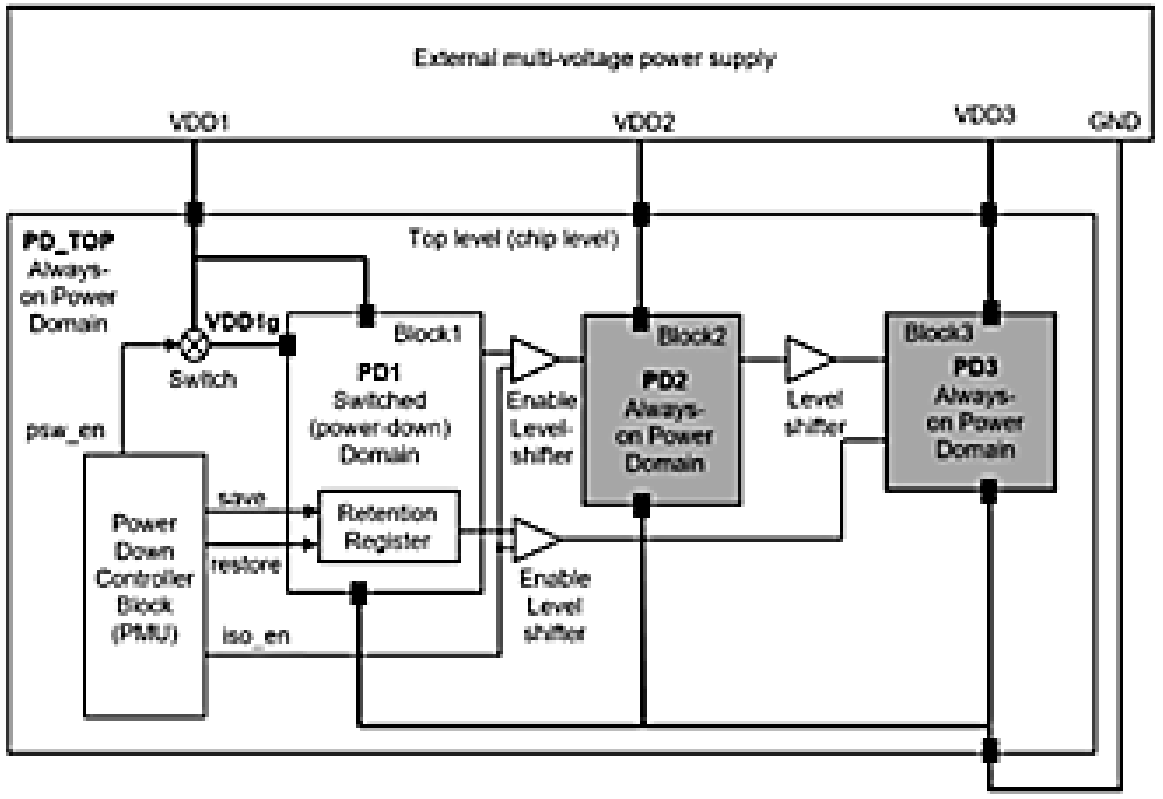
Architecture type	Voltage	Area	Power
Simple datapath (no pipelining or parallelism)	5V	1	1
Pipelined datapath	2.9V	1.3	0.39
Parallel datapath	2.9V	3.4	0.36
Pipeline-Parallel	2.0V	3.7	0.2

Multiple Supplies

Multiple Supply Voltages

- Block-level supply assignment ("power domains" or "voltage islands")
  - Higher throughput/lower latency functions are implemented in higher V_{DD}
  - Slower functions are implemented with lower V_{DD}
  - Often called "Voltage islands"
  - Separate supply grids, level conversion performed at block boundaries
- Multiple supplies inside a block
  - Non-critical paths moved to lower supply voltage
  - Level conversion within the block
  - Physical design challenging
  - (Not used in practice)

Power Domains

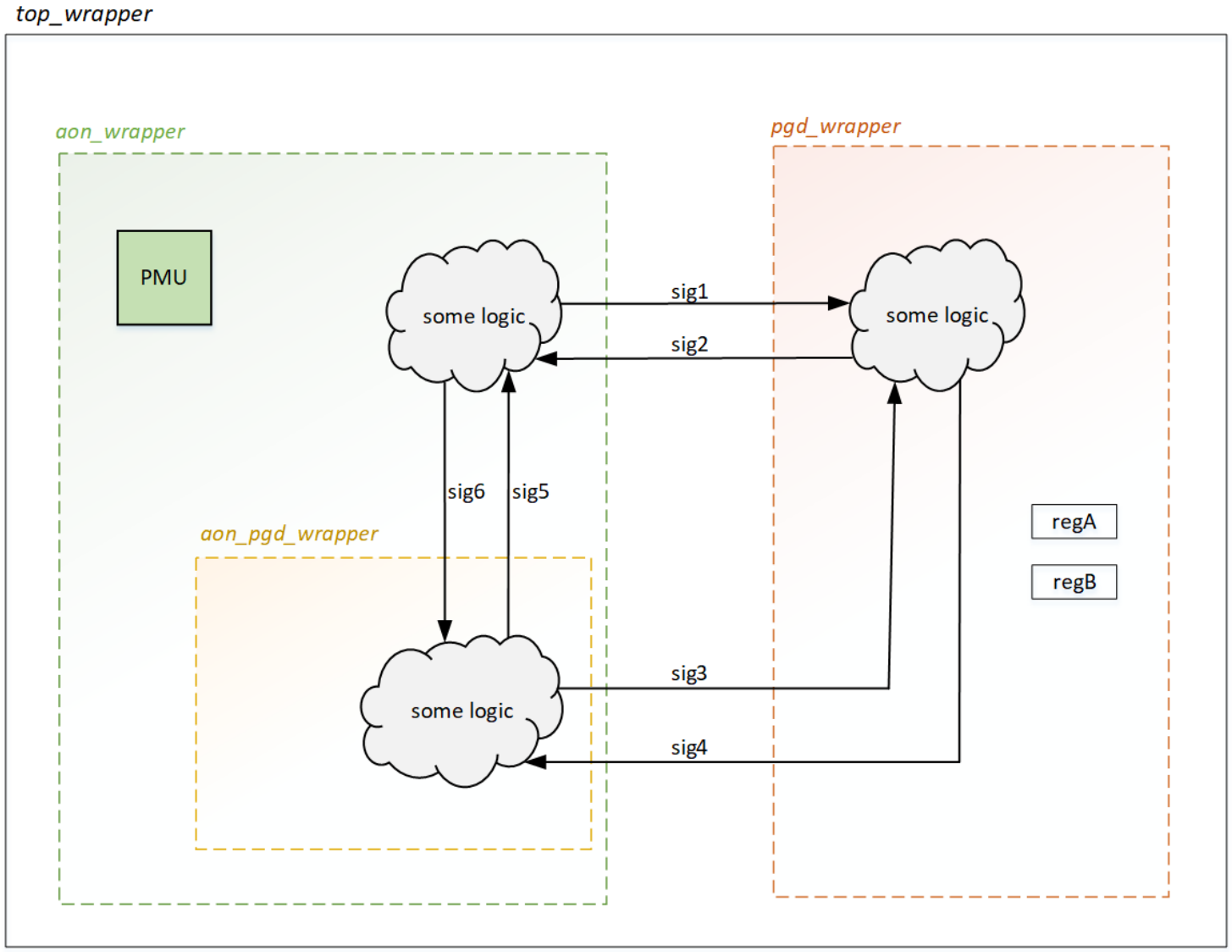


[Synopsys]

Typical Low Power SoC

Utilize Unified Power Format (UPF) to capture design intent
Common Power Format (CPF) is similar

Power Domain Design Intent



https://vlsitutorials.com/upf-low-power-vlsi/

There are primarily 3 power domains –

- Logic inside aon_wrapper [but not inside aon_pgd_wrapper] is always-on.
- Logic inside pgd_wrapper can be power gated.
- Logic inside aon_pgd_wrapper can be power gated but won't be power gated when pgd_wrapper is powered ON.

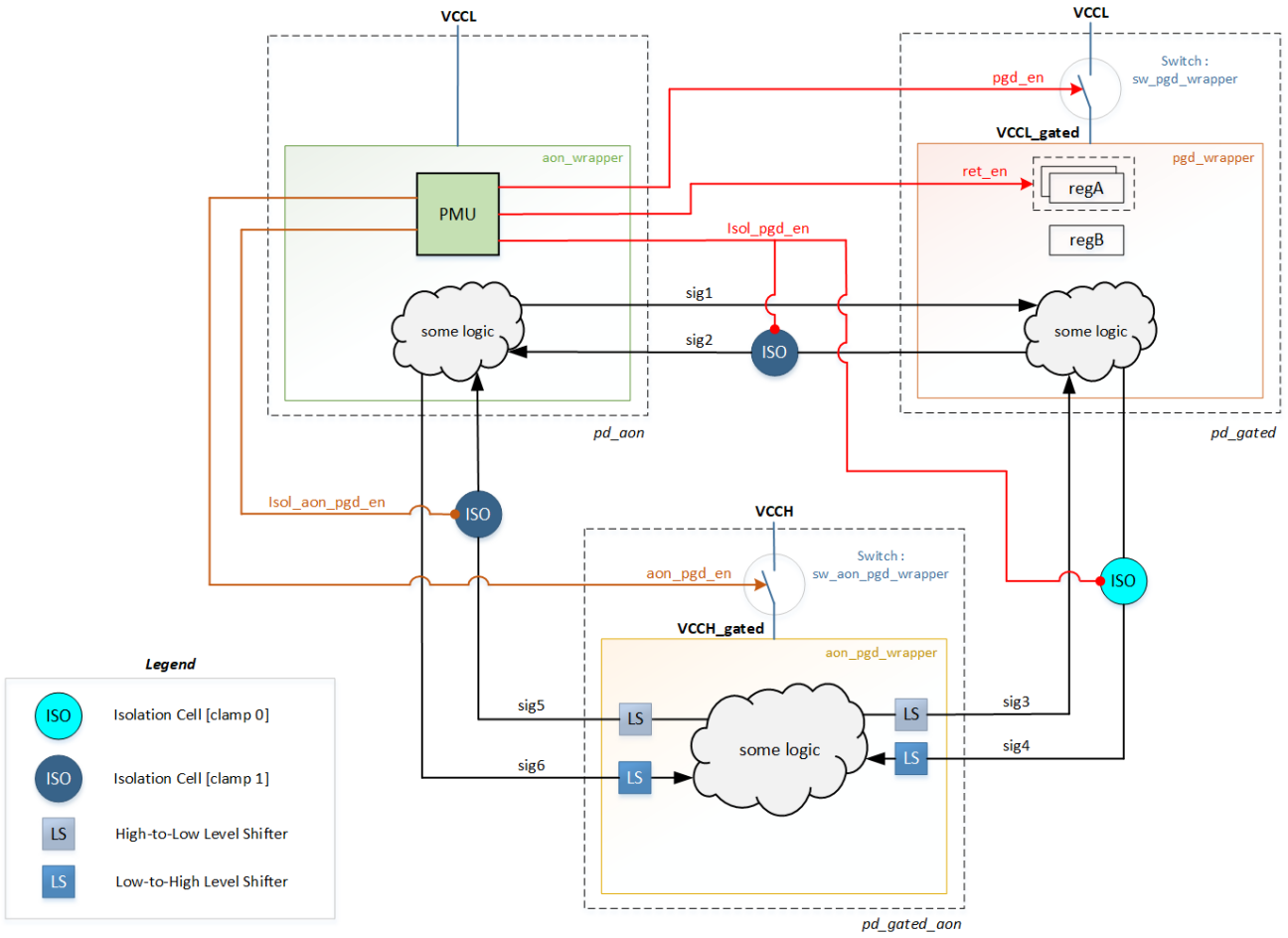
There are two voltage domains –

- The supply voltage to logic inside aon_wrapper [but not inside aon_pgd_wrapper] and logic inside pgd_wrapper is 0.9V.
- The supply voltage to logic inside aon_pgd_wrapper is 1.1V.

There are two registers – reg A and reg B. The state of reg A needs to be retained in power gated state.

There are six signals sig1-sig6 coming to and from different logic blocks.

Unified Power Format



 # Create Power Domains

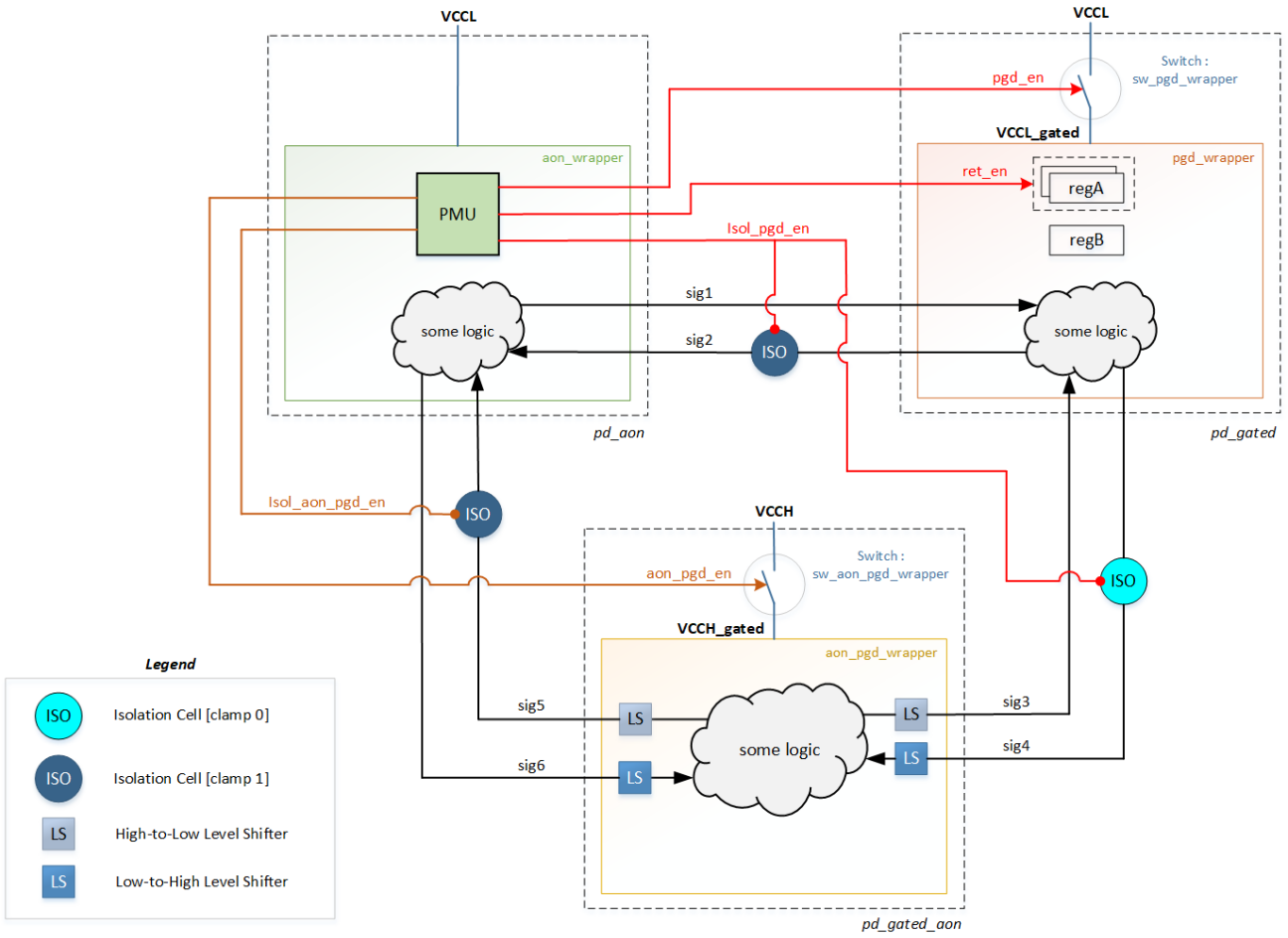
create_power_domain pd_top -include_scope
create_power_domain pd_aon -elements {aon_wrapper}
create_power_domain pd_gated -elements {pgd_wrapper}
create_power_domain pd_gated_aon -elements {{aon_wrapper/aon_pgd_wrapper}}

There are primarily 3 power domains –

- Logic inside aon_wrapper [but not inside aon_pgd_wrapper] is always-on.
- Logic inside pgd_wrapper can be power gated.
- Logic inside aon_pgd_wrapper can be power gated but won't be power gated when pgd_wrapper is powered ON.

https://vlsitutorials.com/upf-low-power-vlsi/

Unified Power Format



 # Create Supply Ports

create_supply_port VCCL -direction in -domain pd_top
create_supply_port VCCH -direction in -domain pd_top
create_supply_port GND -direction in -domain pd_top

 # Create Supply Nets

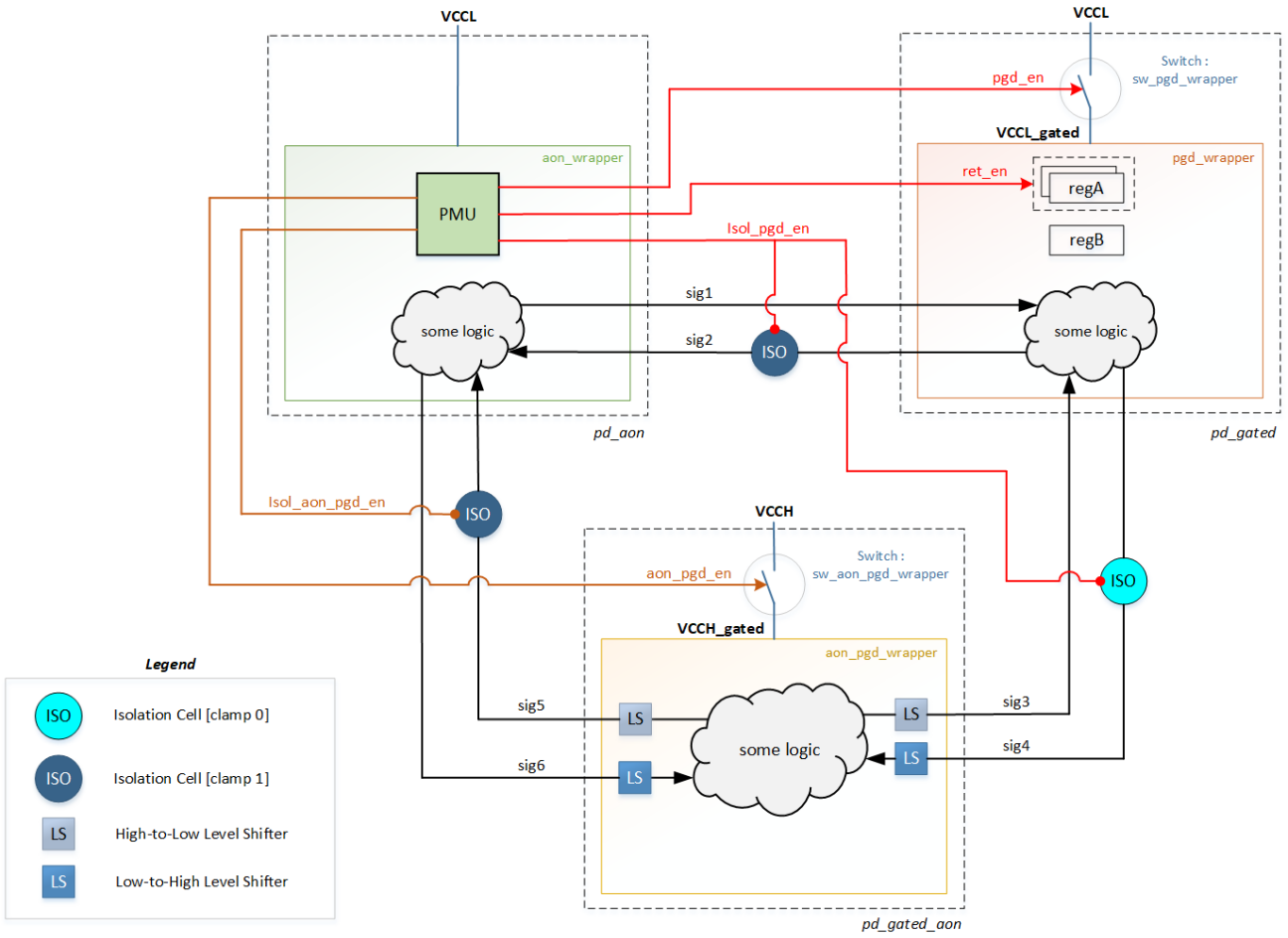
create_supply_net VCCL -domain pd_top
create_supply_net VCCH -domain pd_top
create_supply_net GND -domain pd_top
create_supply_net VCCL -domain pd_aon -reuse
create_supply_net GND -domain pd_aon -reuse
create_supply_net VCCH -domain pd_gated_aon –reuse
create_supply_net VCCH_gated -domain pd_gated_aon
create_supply_net GND -domain pd_gated_aon -reuse
create_supply_net VCCL -domain pd_gated -reuse
create_supply_net VCCL_gated -domain pd_gated
create_supply_net GND -domain pd_gated -reuse

https://vlsitutorials.com/upf-low-power-vlsi/

There are two voltage domains –

- The supply voltage to logic inside aon_wrapper [but not inside aon_pgd_wrapper] and logic inside pgd_wrapper is 0.9V.
- The supply voltage to logic inside aon_pgd_wrapper is 1.1V.

Unified Power Format



 # Connect Supply Nets with corresponding Ports

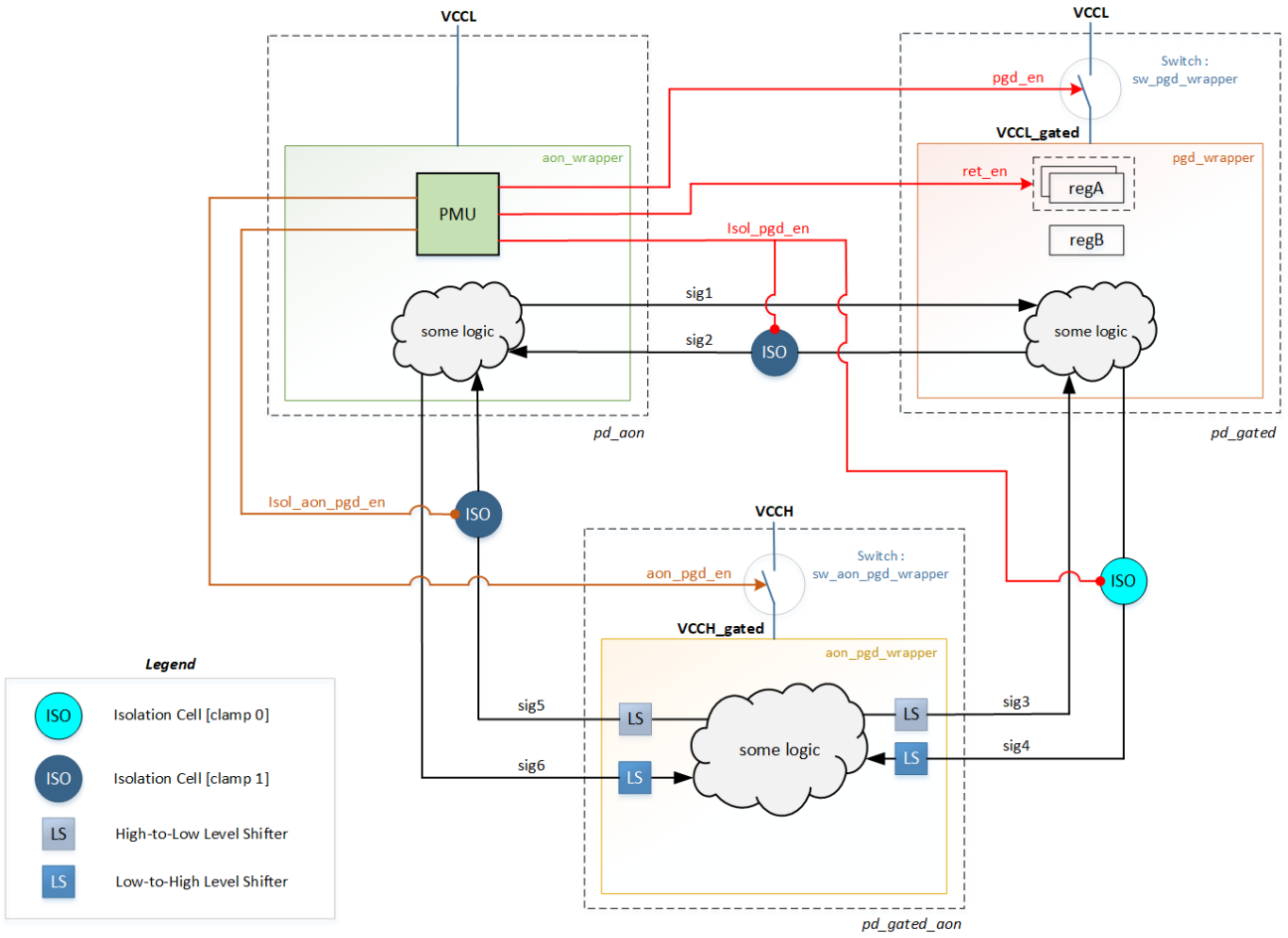
connect_supply_net VCCL -ports VCCL
connect_supply_net VCCH -ports VCCH
connect_supply_net GND -ports GND

 # Establish Connections

set_domain_supply_net pd_top -primary_power_net VCCL -primary_ground_n GND
set_domain_supply_net pd_aon -primary_power_net VCCL -primary_ground_n GND
set_domain_supply_net pd_gated_aon -primary_power_net VCCH_gated -primary_ground_net GND
set_domain_supply_net pd_gated -primary_power_net VCCL_gated -primary_ground_net GND

https://vlsitutorials.com/upf-low-power-vlsi/

Unified Power Format

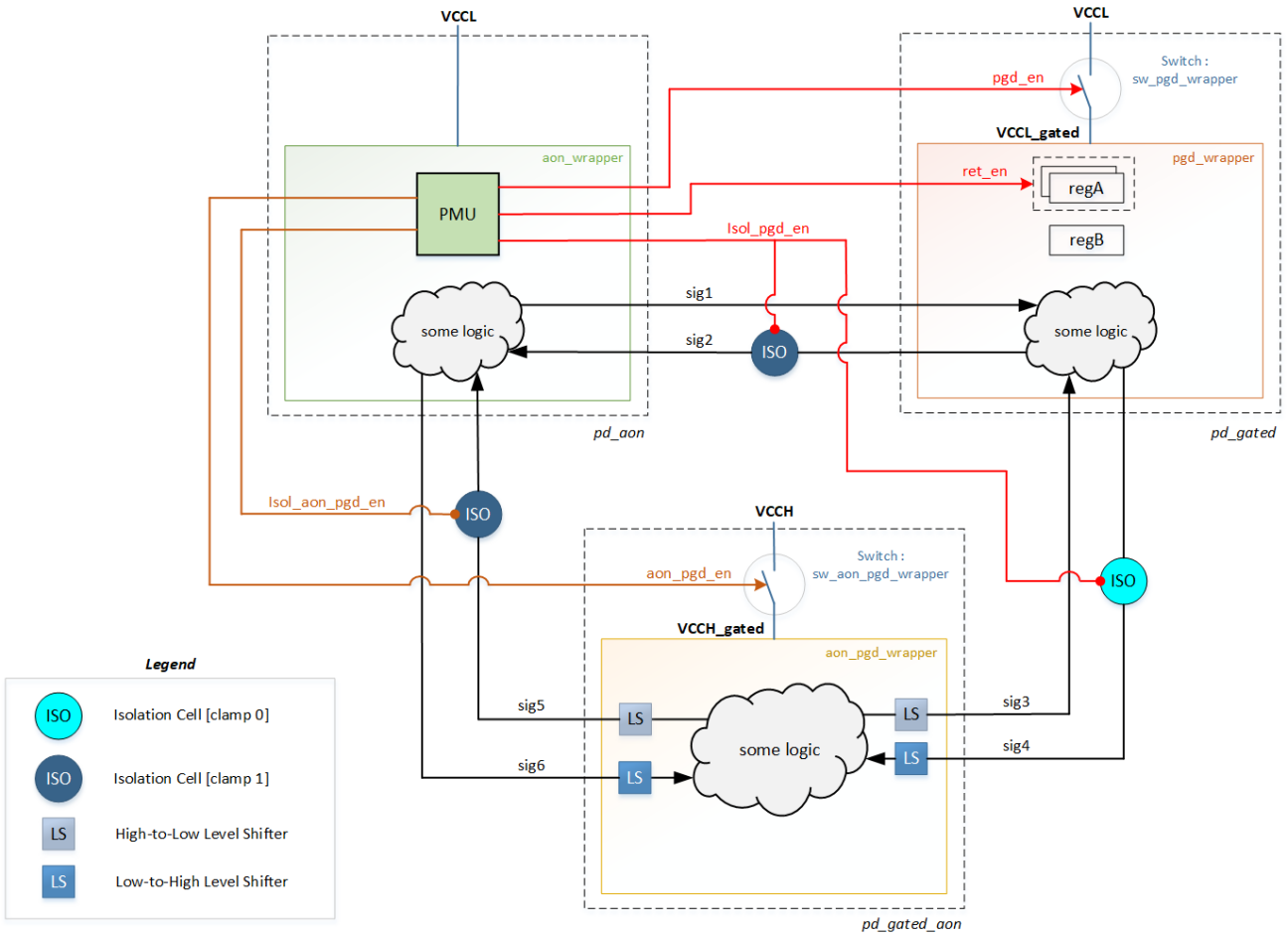


 # Shut-Down Logic for pgd_wrapper & aon_pgd_wrapper

create_power_switch sw_pgd_wrapper \
-domain pd_gated \
-input_supply_port "sw_VCCL VCCL " \
-output_supply_port "sw_VCCL_gated VCCL_gated" \
-control_port "sw_pgd_en aon_wrapper/pmu/pgd_en" \
-on_state "SW_PGD_ON sw_VCCL {!sw_pgd_en}"
create_power_switch sw_aon_pgd_wrapper \
-domain pd_gated_aon \
-input_supply_port "sw_VCCH VCCH " \
-output_supply_port "sw_VCCH_gated VCCH_gated" \
-control_port "sw_aon_pgd_en aon_wrapper/pmu/aon_pgd_en" \
-on_state "SW_AONPGD_ON sw_VCCH {!sw_aon_pgd_en}"

https://vlsitutorials.com/upf-low-power-vlsi/

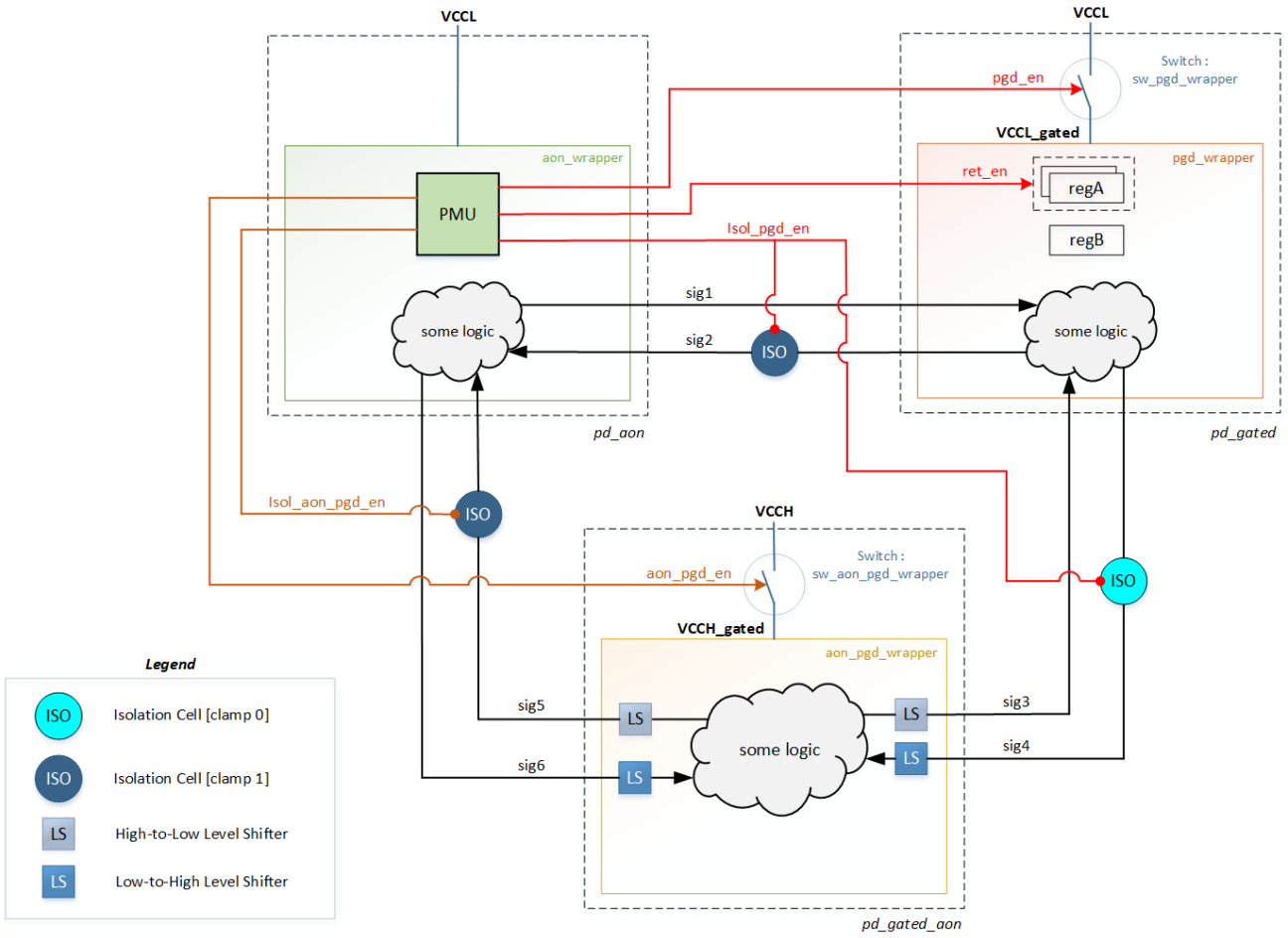
Unified Power Format




```
# Isolation strategy
set_isolation isol_clamp1_sig_from_pgd \
-domain pd_gated \
-isolation_power_net VCCL \
-isolation_ground_net GND \
-clamp_value 1 \
-elements {pgd_wrapper/sig2}
set_isolation_control isol_clamp1_sig_from_pgd \
-domain pd_gated \
-isolation_signal aon_wrapper/pmu/isol_pgd_en \
-isolation_sense low \
-location parent
set_isolation isol_clamp0_sig_from_pgd \
-domain pd_gated \
-isolation_power_net VCCL \
-isolation_ground_net GND \
-clamp_value 0 \
-elements {pgd_wrapper/sig4}
set_isolation_control isol_clamp0_sig_from_pgd \
-domain pd_gated \
-isolation_signal aon_wrapper/pmu/isol_pgd_en \
-isolation_sense low \
-location parent
...
```

https://vlsitutorials.com/upf-low-power-vlsi/

Unified Power Format

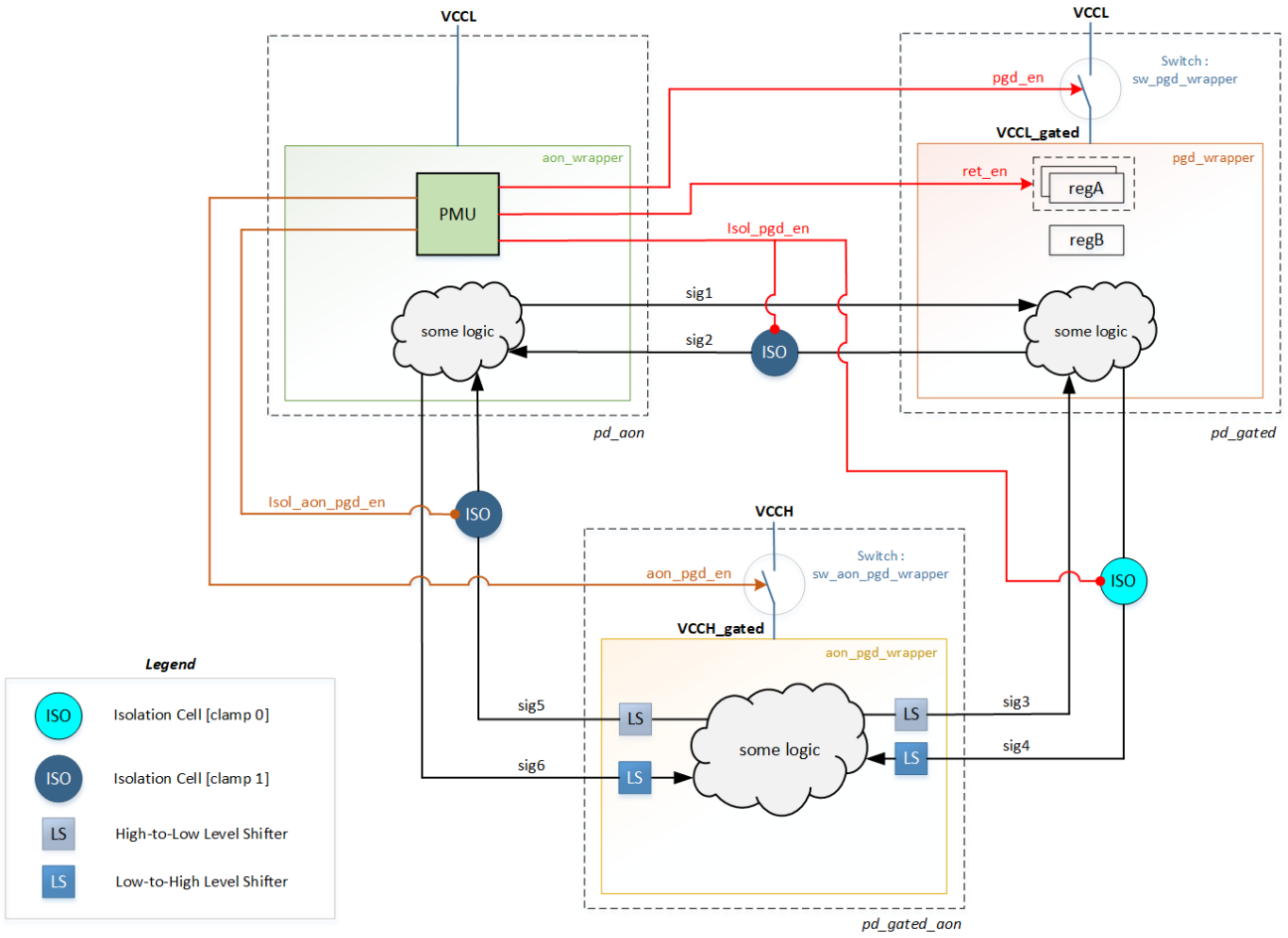


 # Level Shifter strategy

set_level_shifter LtoH_sig_to_aonpgd \
-domain pd_gated_aon \
-applies_to inputs \
-rule low_to_high \
-location self
set_level_shifter HtoL_sig_from_aonpgd \
-domain pd_gated_aon \
-applies_to outputs \
-rule high_to_low \
-location self

https://vlsitutorials.com/upf-low-power-vlsi/

Unified Power Format



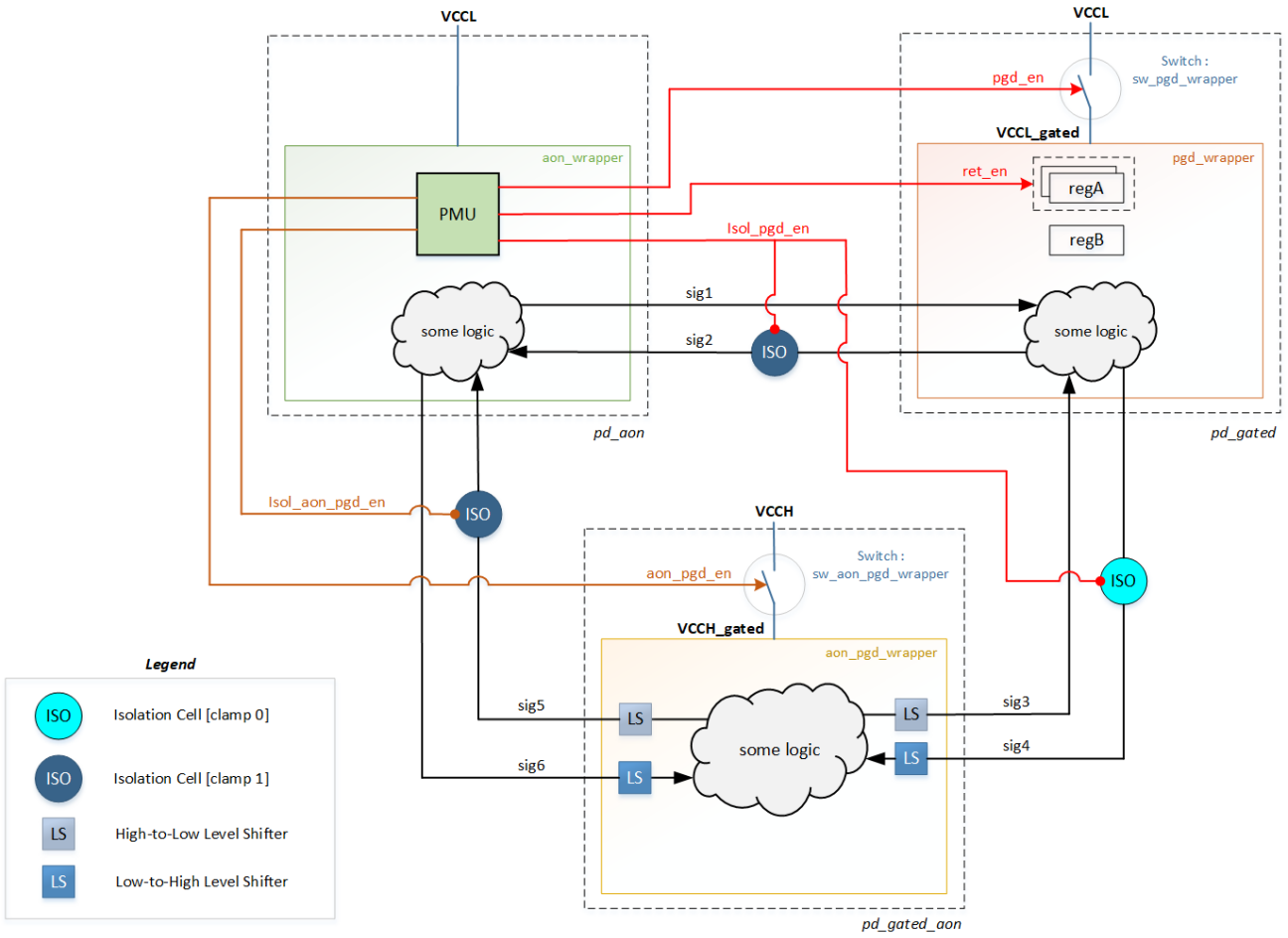
 # Retention strategy

set_retention pgd_retain \
-domain pd_gated \
-retention_power_net VCCL \
-retention_ground_net GND \
-elements {pgd_wrapper/regA}
set_retention_control pgd_retain \
-domain pd_gated \
-save_signal {aon_wrapper/pmu/ret_en high} \
-restore_signal {aon_wrapper/pmu/ret_en low}

There are two registers – reg A and reg B.
The state of reg A needs to be retained in power gated state.

https://vlsitutorials.com/upf-low-power-vlsi/

Unified Power Format

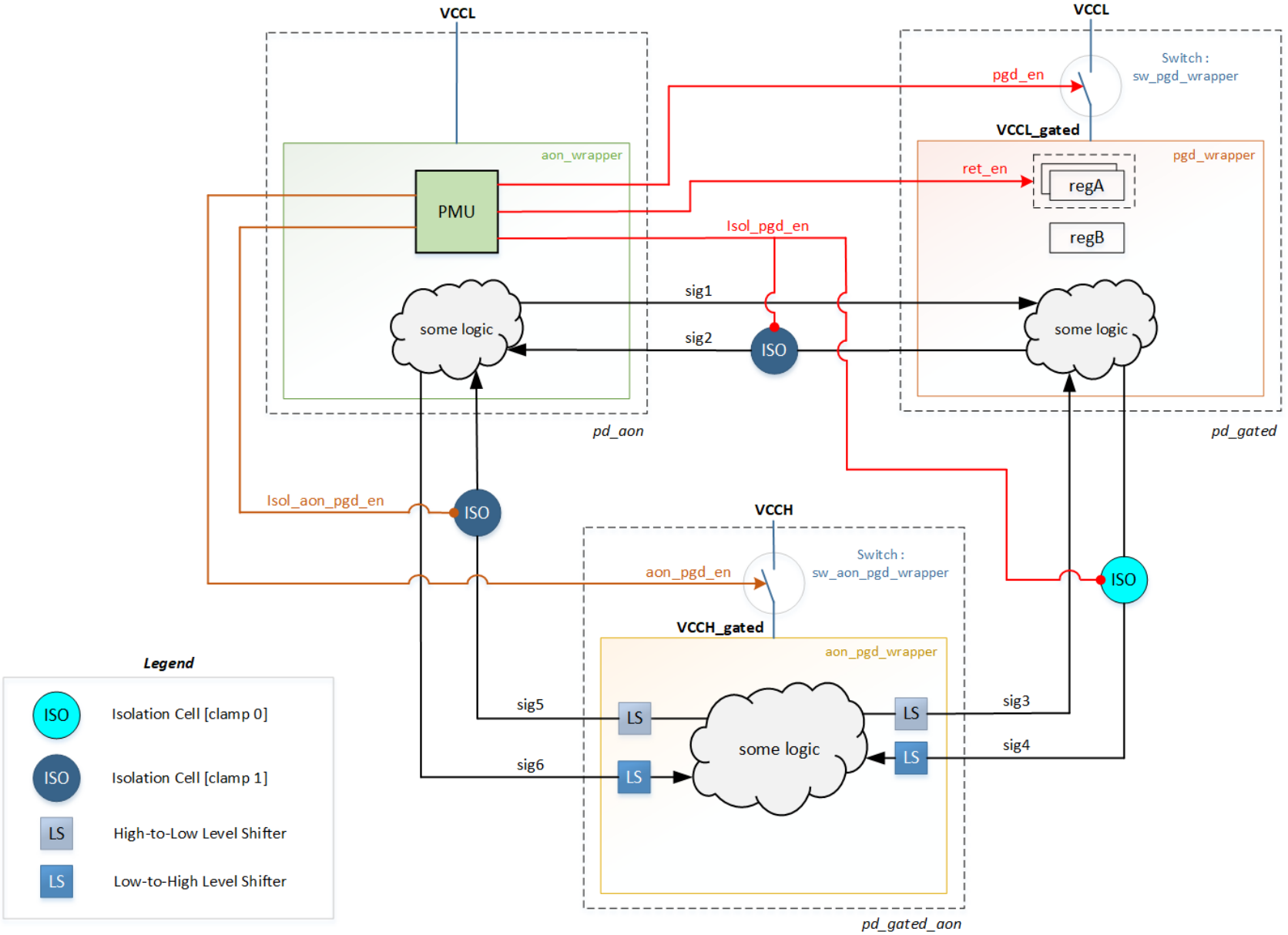


https://vlsitutorials.com/upf-low-power-vlsi/

```
# Create Power State Table
add_port_state VDDH \
-state {HighVoltage 1.1}
add_port_state VDDL \
-state {LowVoltage 0.9}
add_port_state sw_aon_pgd_wrapper/sw_VCCH_gated \
-state {HighVoltage 1.1} \
-state {aonpgd_off off}
add_port_state sw_pgd_wrapper/sw_VCCL_gated \
-state {LowVoltage 0.9} \
-state {pgd_off off}

create_pst pwr_state_table \
-supplies {VCCH VCCL VDDH_gated VDDL_gated}
add_pst_state PRE_BOOT \
-pst pwr_state_table \
-state { HighVoltage LowVoltage aonpgd_off pgd_off}
add_pst_state AONPGD_ON \
-pst pwr_state_table \
-state { HighVoltage LowVoltage HighVoltage pgd_off}
add_pst_state PGD_ON \
-pst pwr_state_table \
-state { HighVoltage LowVoltage aonpgd_off LowVoltage}
add_pst_state ALL_ON \
-pst pwr_state_table \
-state { HighVoltage LowVoltage HighVoltage LowVoltage}
```

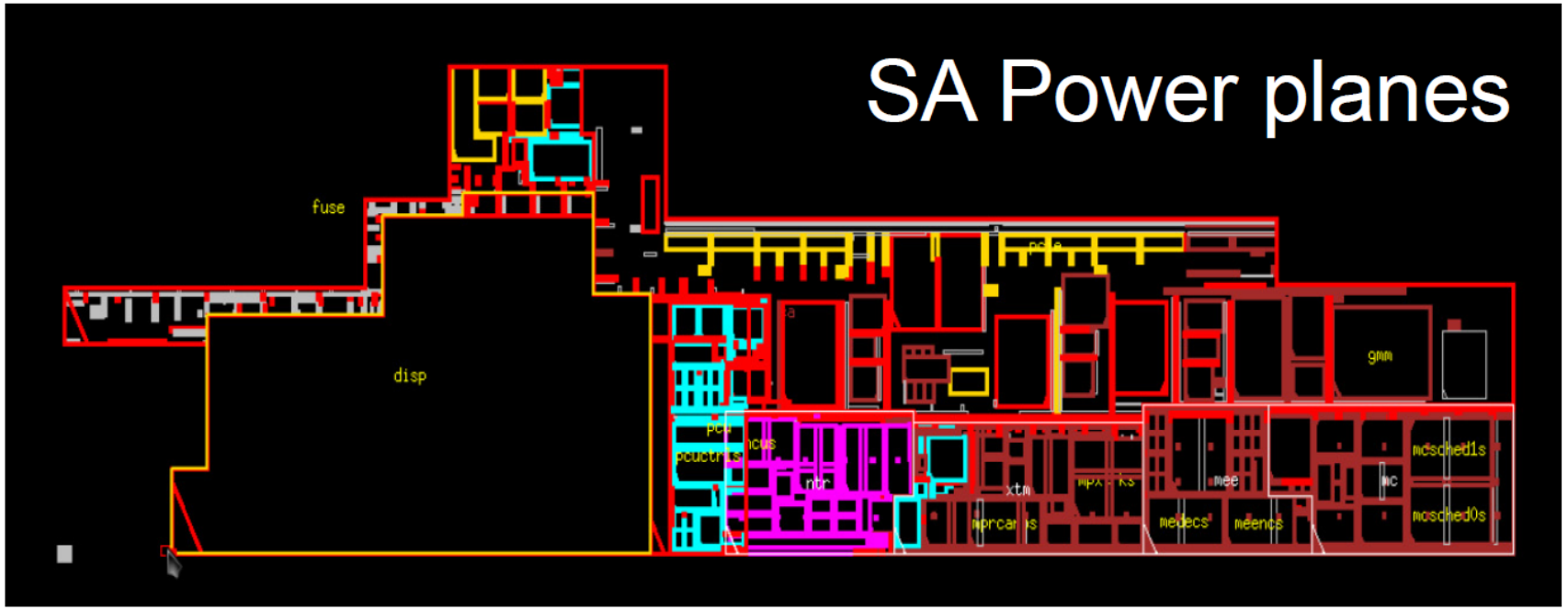
Unified Power Format



https://vlsitutorials.com/upf-low-power-vlsi/

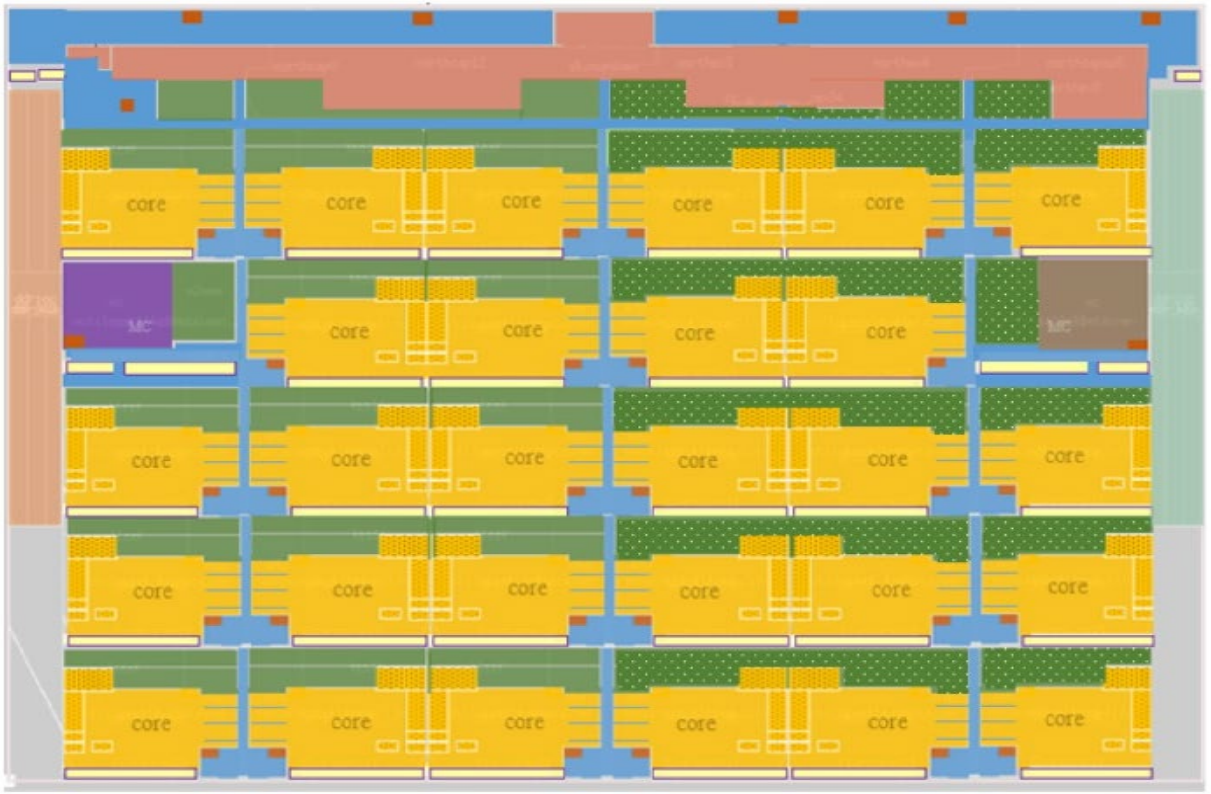
Practical Examples

- Intel Skylake (ISSCC'16)
  - Four power planes indicated by colors



Practical Examples

- Intel 28-core Skylake-SP (ISSCC'18)

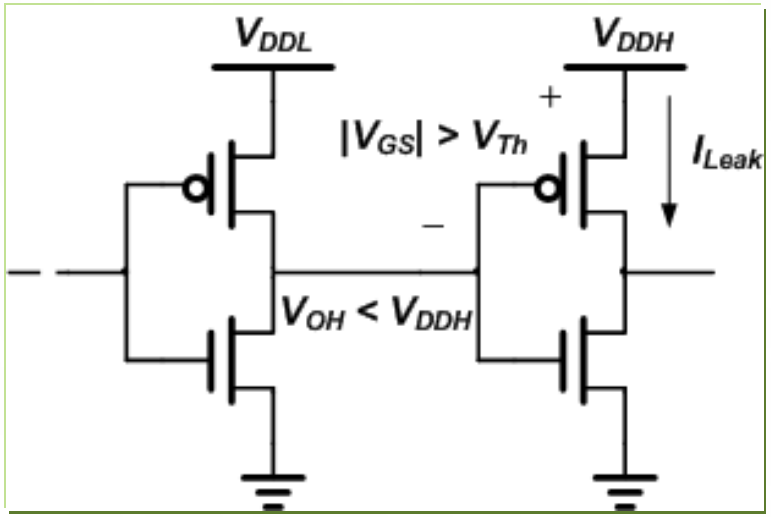


Vcc: core supply (per core)
Vccclm: Un-core supply
Vccsa: System Agent supply
Vccio: Infrastructure supply
Vccsfr: PLL supply
Vccddrd: DDR logic supply
Vccddra: DDR I/O supply

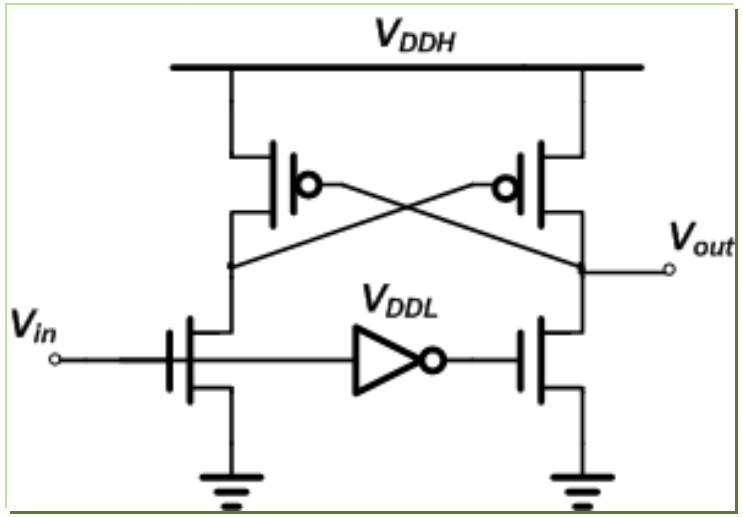
- 9 primary VCC domains are partitioned into 35 VCC planes

Leakage Issue

- Driving from V_{DDL} to V_{DDH}

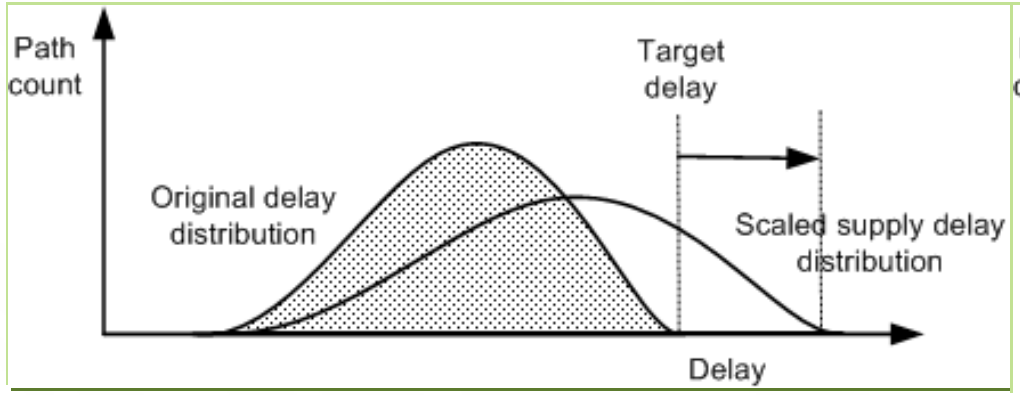


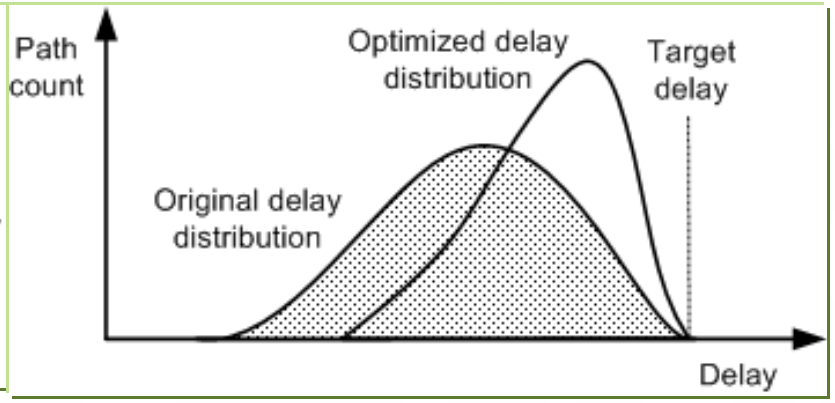
- Level converter



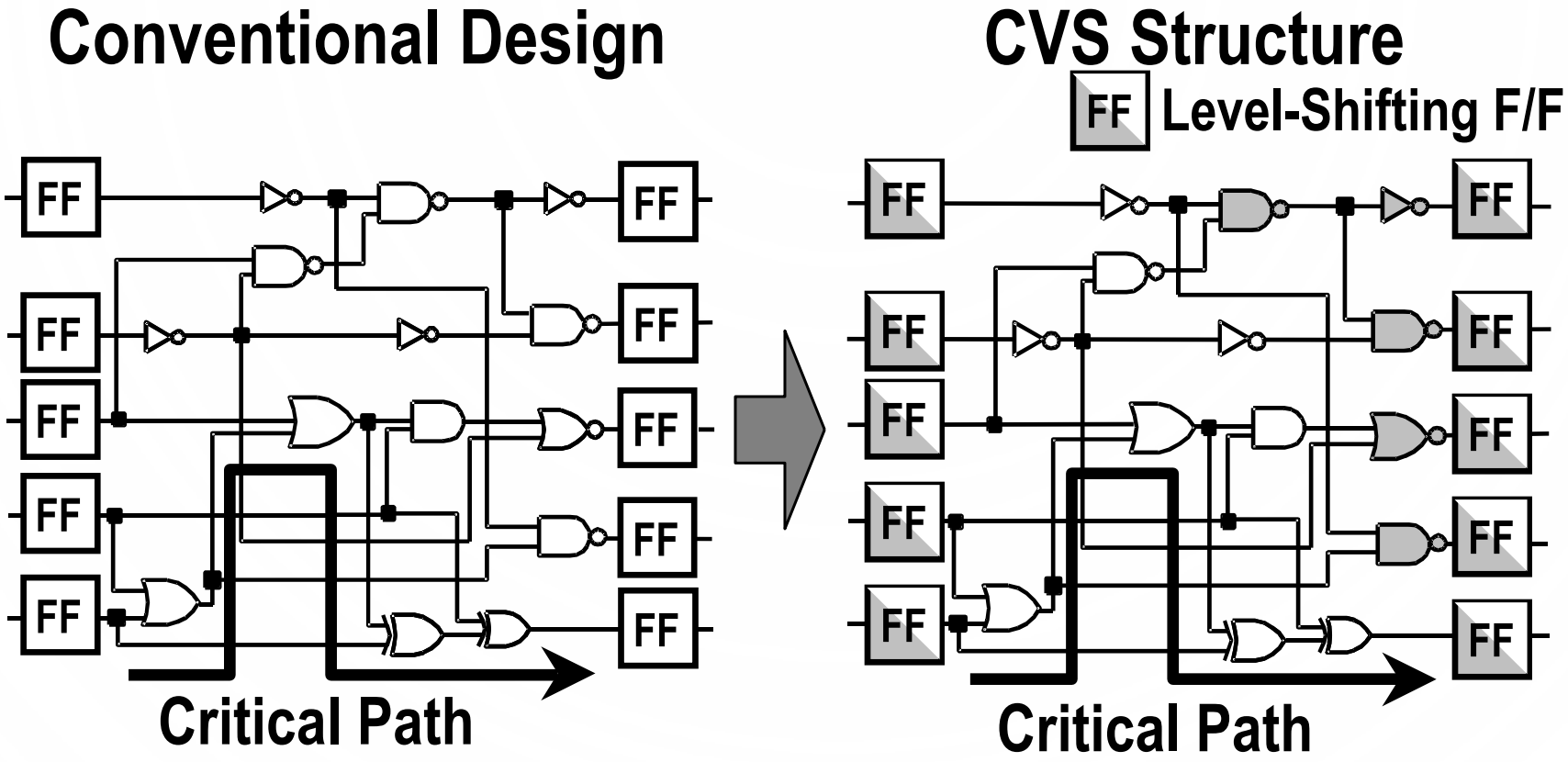
Multiple Supplies Within A Block

- Downsizing, lowering the supply on the critical path will lower the operating frequency
- Downsize (lowering supply) non-critical paths
  - Narrows down the path delay distribution
  - Increases impact of variations





Multiple Supplies in a Block



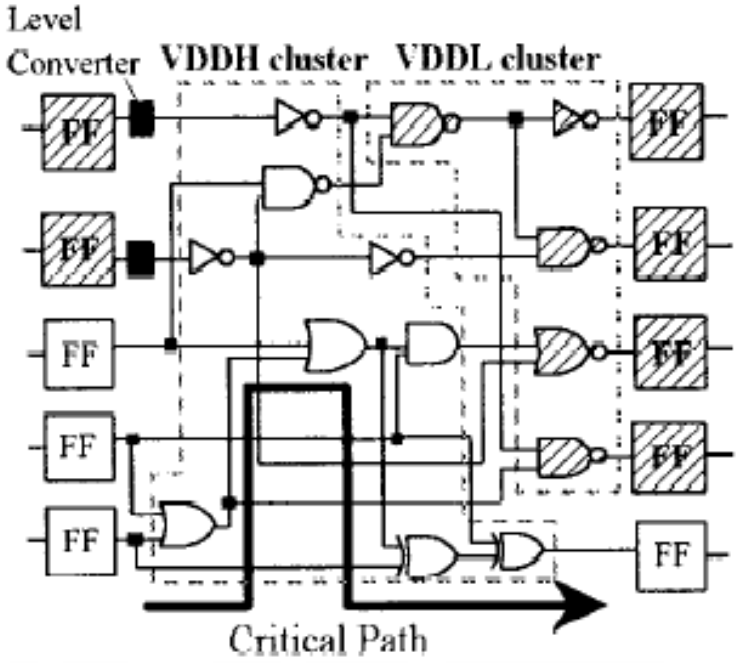
Lower V_{DD} portion is shaded

"Clustered voltage scaling"

M.Takahashi, ISSCC'98.

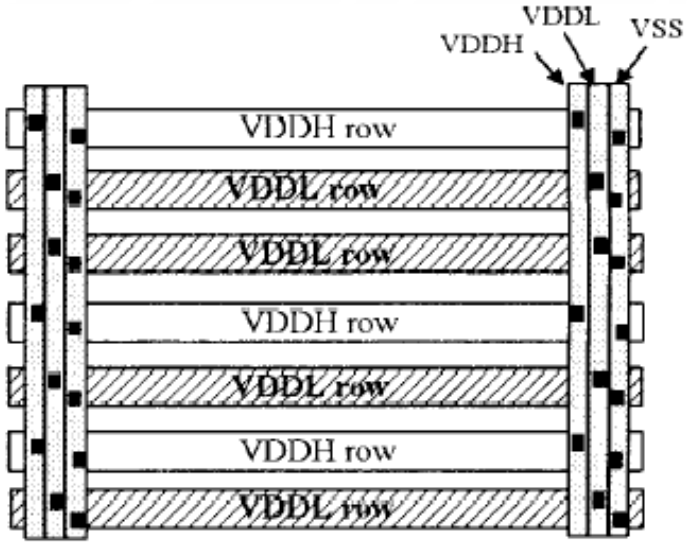
Multiple Supplies in a Block

CVS

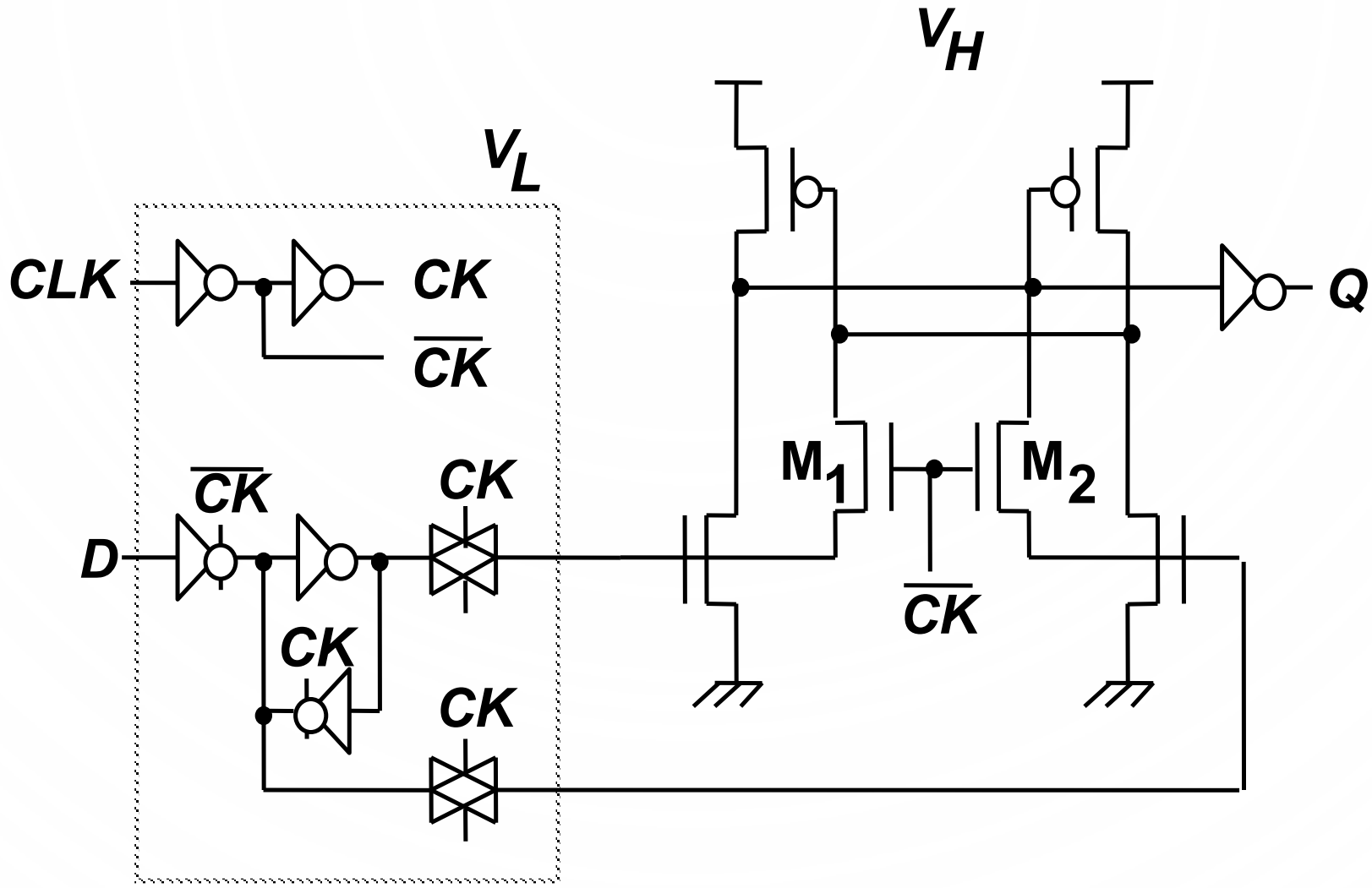


Usami'98

Layout:



Level-Converting Flip-Flop



Summary

- Power-performance tradeoffs
  - Sizing
  - Supplies
  - Thresholds
- Lowering supplies
- Multiple supply voltages

Next Lecture

- Low-power design
  - Dynamic voltage scaling
  - Clock gating