Clock Skew Budgets

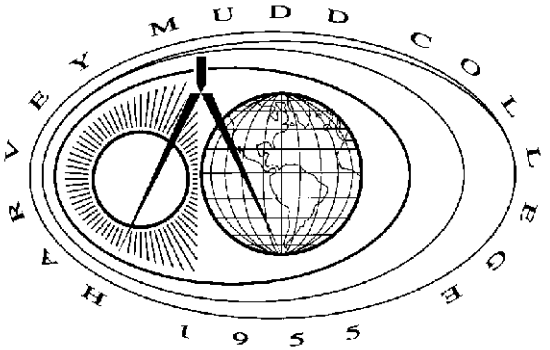
David Harris

David_Harris@hmc.edu

February, 2002

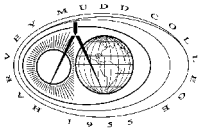
Harvey Mudd College

Claremont, CA



Outline

- Introduction
- Statistical Clock Skew Budgets
- Data Delay Variation Effects
- Sources of Clock and Data Skew
- Example: McKinley Clock Skew Budget

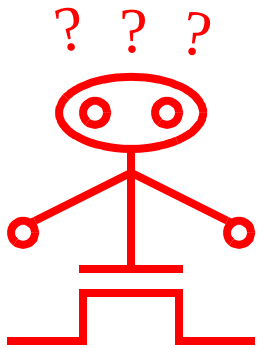


Questions for Design

1) Will my chip work?

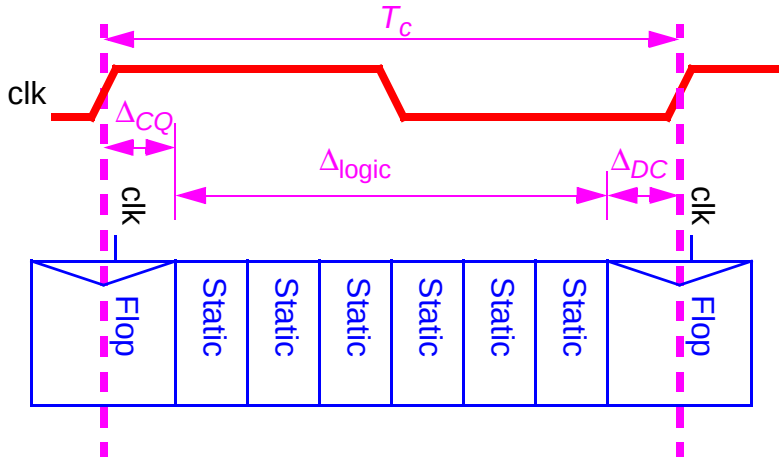
2) How fast will my chip work?

3) What paths should I modify to make my chip work faster?

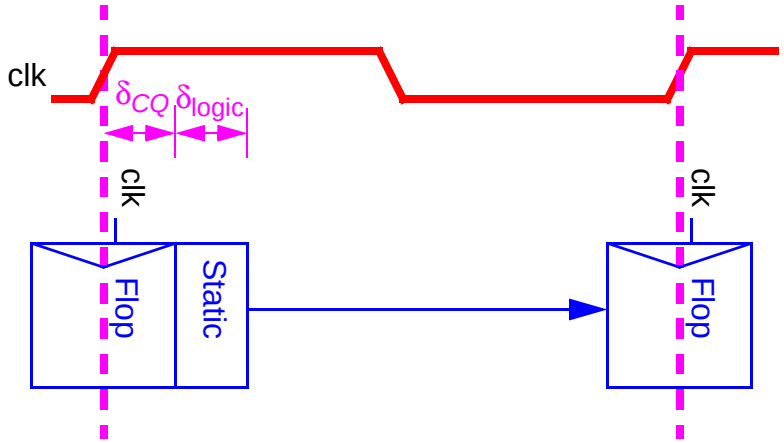


Timing Constraints

Setup Time: $\Delta_{CQ} + \Delta_{logic} + \Delta_{DC} < T_c$



Hold Time: $\delta_{CQ} + \delta_{logic} > \Delta_{CD}$



Clock:

T_c: Cycle Time

Flop:

Δ_{CQ}: Prop. Delay

δ_{CQ}: Cont. Delay

Δ_{DC}: Setup Time

Δ_{CD}: Hold Time

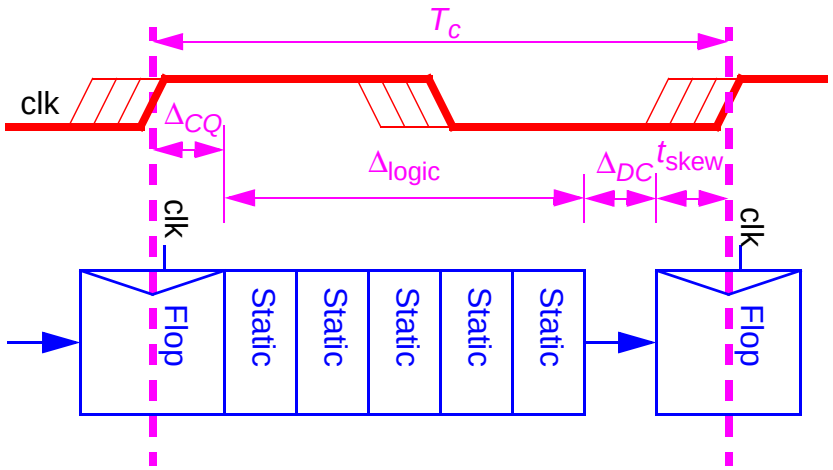
Logic:

Δ_{logic}: Prop. Delay

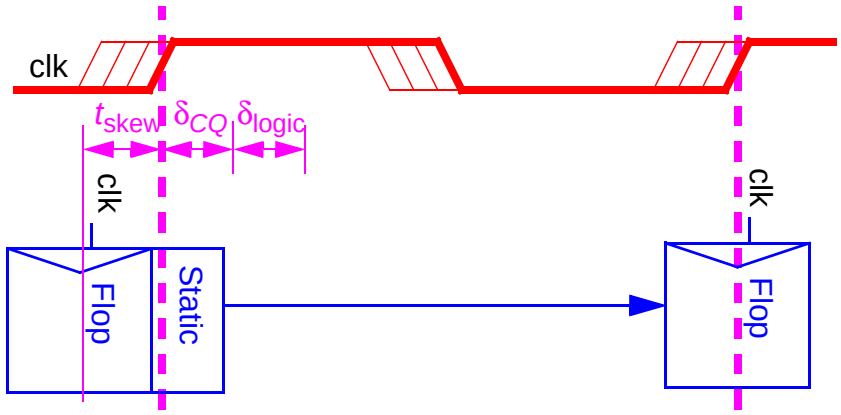
δ_{logic}: Cont. Delay

Timing Constraints with Clock Skew

Setup Time: $\Delta_{CQ} + \Delta_{logic} + \Delta_{DC} < T_c - t_{skew}$



Hold Time: $\delta_{CQ} + \delta_{logic} > \Delta_{CD} + t_{skew}$



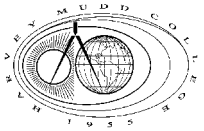
Will my chip work?

Setup time problems can be solved by increasing the clock period.

Hold time problems require redesigning chip

- Yield: design so nearly all chips work

Expect to be more conservative budgeting clock skew for hold time constraints than for setup time constraints.



How fast will my chip work?

Essential question for some products (e.g. Gigabit Ethernet)

For other products, this does not require a precise answer at design time

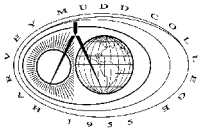
- Ex: microprocessors should run as fast as possible
- Sort parts during manufacturing test into speed bins

Some microprocessors have been designed ignoring skew in timing analysis

- Design decent clock network to minimize skew
- Increase hold times to account for skew
- Ignore skew in setup time calculations
- Accept whatever speed chips run at
- Pros: most engineers & tools don't need to understand skew
- Cons: inefficient if skew impacts different paths differently

If we do consider skew, how do we develop a skew budget?

- Systematic delay variations only
- Systematic, random, drift, jitter
- Systematic, random, drift, jitter with data delay variations



What paths should I modify to make my chip work faster?

If skew impacts all paths equally, fix the slowest path.

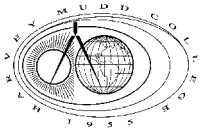
But skew may not impact all paths equally

- Local vs. global skew (clock domains)
- Skew-tolerant circuit techniques
- Half-cycle and multi-cycle paths

This talk will focus on systems with single-cycle paths between edge-triggered flip-fops but multiple clock domains with different amounts of skew.

The most critical path is the one with the greatest sum of delay and skew

- $\Delta_{CQ} + \Delta_{logic} + \Delta_{DC} + t_{skew} < T_c$
- Not necessarily path with slowest logic delay
- Important to budget skew properly to avoid wasting time fixing wrong paths



Example: Setup Time Skew Budget

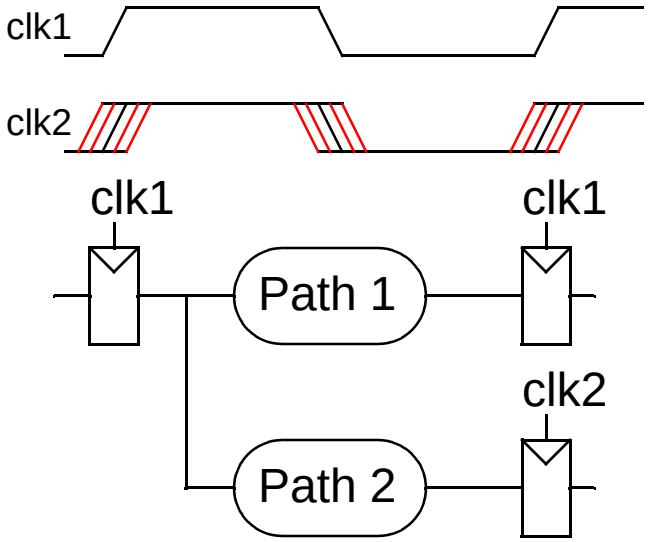
Path delays ($\Delta_{CQ} + \Delta_{logic} + \Delta_{DC}$):

- Path 1: 1000 ps
- Path 2: 1000 ps

Clocks

- Assume clk1 is ideal
- Arrival time of clk2 is random variable uniformly distributed over +/- 100 ps range

What is the expected cycle time?



Thought experiment: build 1000 chips, measure cycle times, find median

- Monte Carlo simulation
- T_c = 1025 ps

Think of this as a 25 ps clock skew budget

- Clock skew budget and actual clock skew are only indirectly related

Example: Hold Time Skew Budget

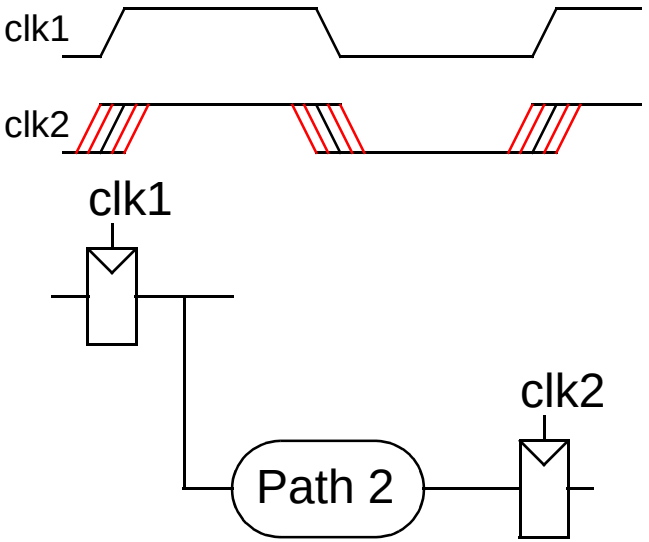
Suppose path 2 is short

- Clock skew effectively increases hold time
- $\delta_{logic} = (\Delta_{CD} + t_{skew}) - \delta_{CQ}$
- where t_{skew} is a budget set by designer

Clocks

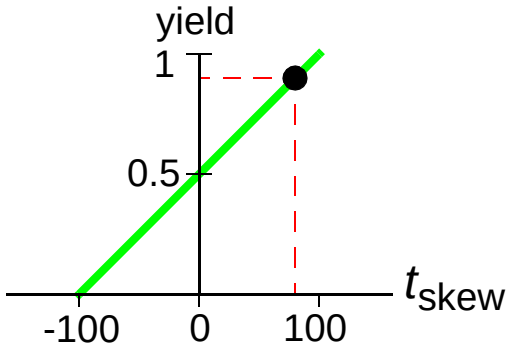
- Assume clk1 is ideal
- Arrival time of clk2 is random variable uniformly distributed over +/- 100 ps range

What fraction of chips will work?



Thought experiment: build 1000 chips given skew budget, find how many work

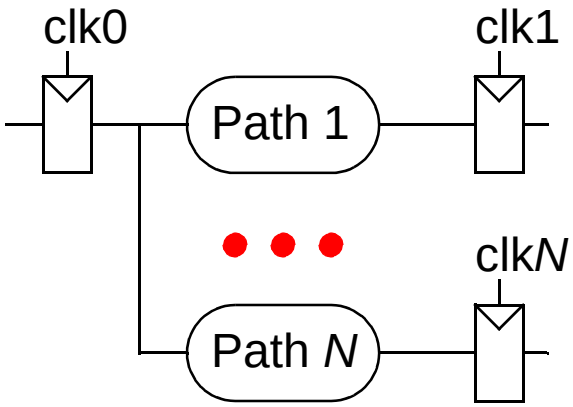
- Monte Carlo simulation
- Budget t_{skew} = 90 ps for 95% yield



Multiple Paths

What if there are *N* paths?

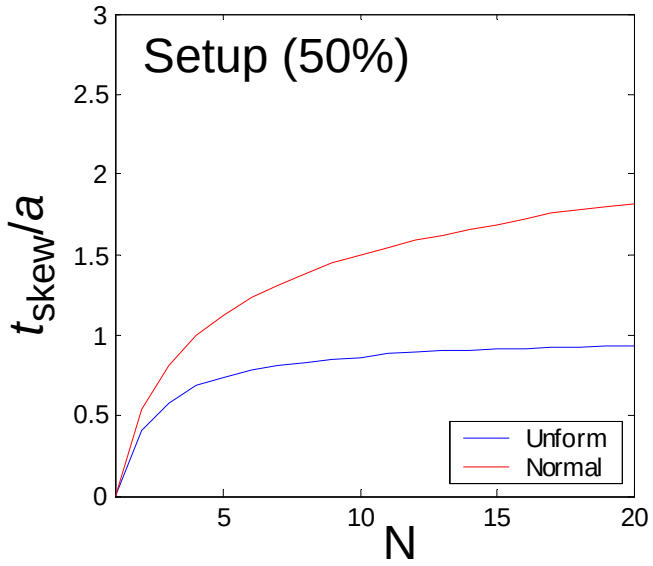
- Each path has uncertain clock arrival time

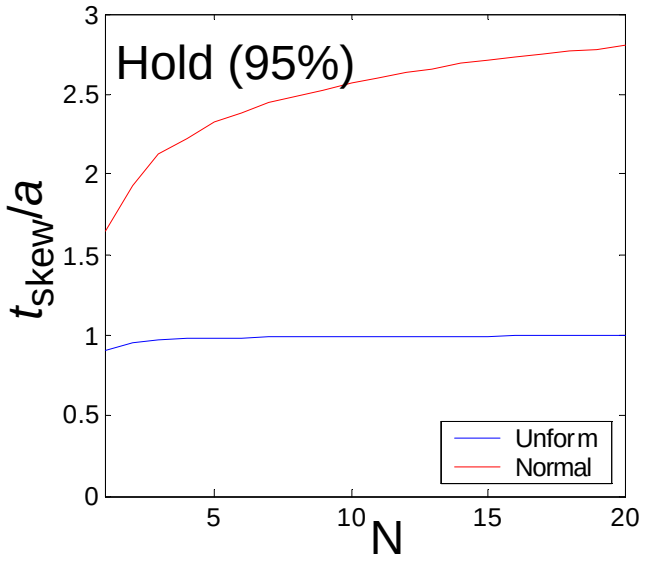


Consider distribution of clock arrival times:

- Uniform +/-*a* (*a* = half range)
- Normal (Gaussian) w/ standard deviation = *a*

Use Monte Carlo simulation to find skew budgets for setup and hold constraints





Different Skews Between Different Clocks

Again assume 1000 ps paths & unif. distribution

- clk2 far away, a = 100 ps
- clk3 closer, a = 60 ps

Consider Paths 1 and 2

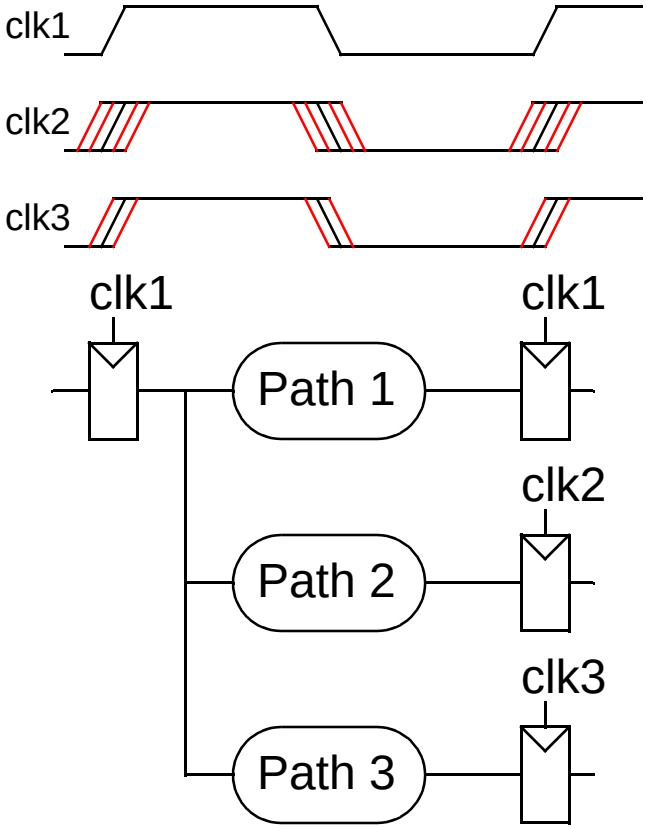
- T_c = 1025 ps
- $t_{skew}^{global\text{-}setup}$ = 25 ps

Consider Paths 1 and 3

- T_c = 1015 ps
- $t_{skew}^{local\text{-}setup}$ = 15 ps

If there were a large number of paths:

- $t_{skew}^{global\text{-}setup} \approx 100ps$
- $t_{skew}^{local\text{-}setup} \approx 60ps$



Local paths can be 40 ps longer than global without impacting performance

Data Delay Variations

But clocks are not the only part of the chip subject to delay variation

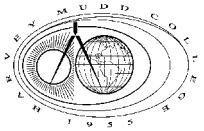
- Data delays differ from predicted values for similar reasons
- Call this *data skew*

Some paths will be longer than expected, some shorter

- Leads to lower performance than simulations predict
- But this is not a new effect
- Ignore the penalty (or put it in our typical models)

However, it is unlikely path seeing worst clock skew also sees worst data skew

- Thus clock skew budgets need not be as large as predicted



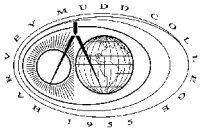
Data Delay Variation Example

Consider a chip with

- Global clock variation: Gaussian with stdev = 100 ps
- Local clock variation: Gaussian with stdev = 60 ps
- Path delay: Gaussian distribution around 1000 ps
- 500 nearly critical local paths with stdev = 50 ps
- 500 nearly critical global paths with stdev = 80 ps

Median Cycle Times	Local Paths	Global Paths
T_c (nominal)	1000	1000
T_c w/ clock skew	1179	1298
Effective t_{skew} (no data delay variation)	179	298
T_c w/ data skew	1150	1239
T_c w/ data and clock skew	**1234**	**1383**
Effective t_{skew} (w/ data delay variation)	234 (84)	383 (144)

Effective clock skew is lower when considering data delay variation



Sources of Delay Variation

Clock and data delay variations can be caused by the following sources

Systematic: skew that exists even when system is ideal (found by SPICE)

- Data paths that are faster than worst-case
- Flight time along final clock lines
- Nonuniform loading and wire lengths

Random: skew caused by intra-die process variation

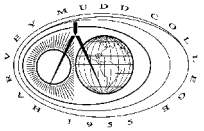
- FETs: L_e, V_t, t_{ox}
- Wires: width, thickness, spacing, resistivity

Drift: low-frequency variations in operating environment

- Temperature fluctuations in space and time

Jitter: high-frequency variations in operating environment

- Voltage variations on clock buffers and receivers
- PLL/DLL and input clock jitter



McKinley Clock Distribution Network

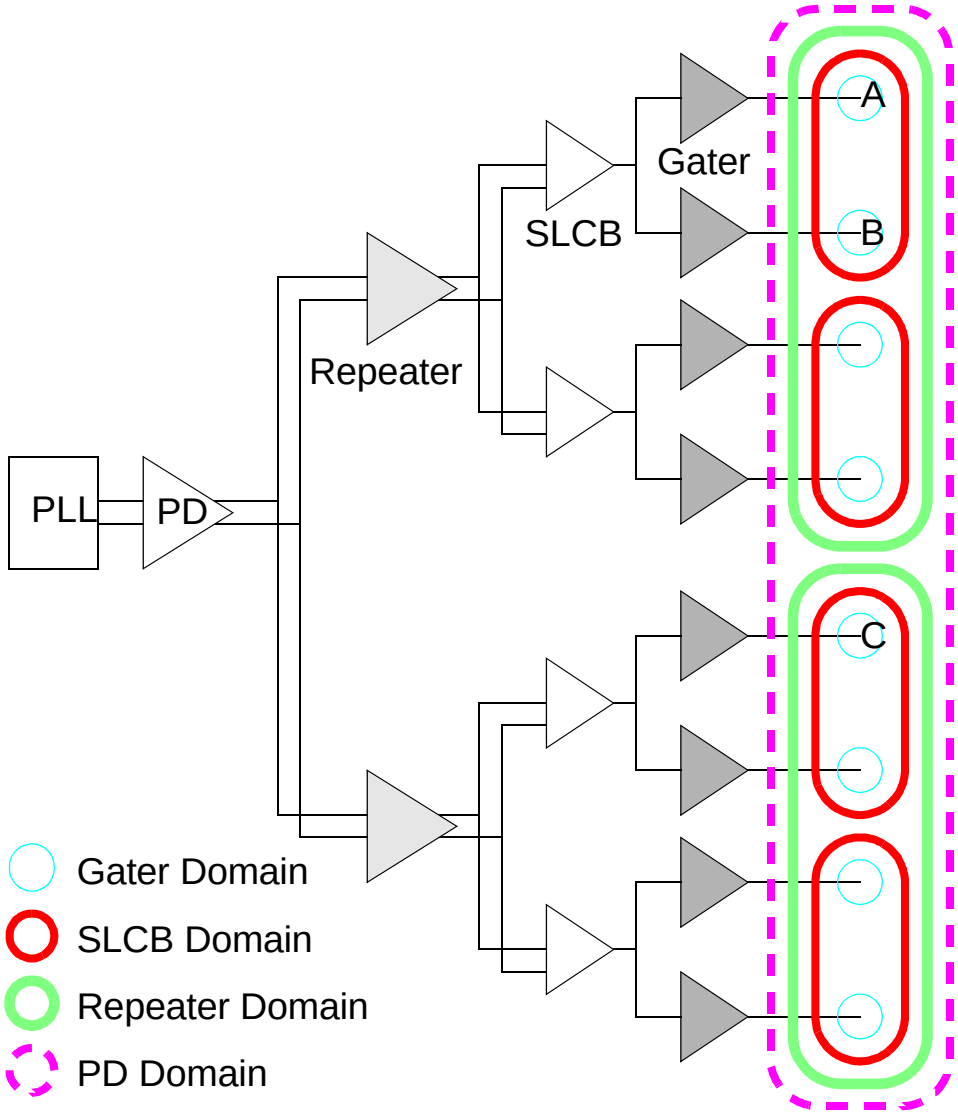
Develop a skew budget for the next-generation 1 GHz Itanium processor (code name McKinley)

Four-level skew hierarchy

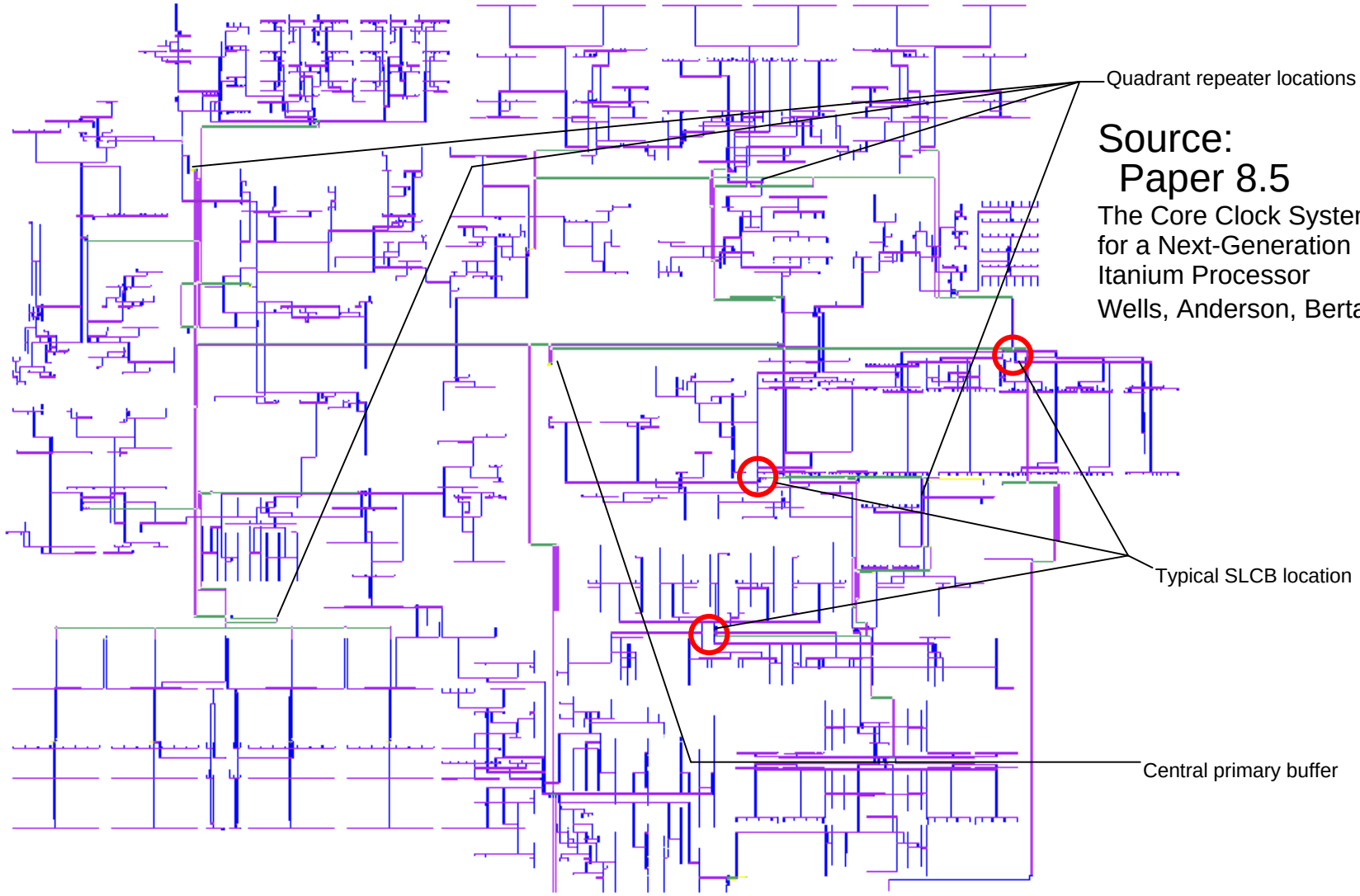
- Gater
- SLCB
- Repeater
- Primary Driver

Dominant Sources of Variability

- PLL Jitter
- Wire length mismatches
- Buffer voltage
- Buffer temperature
- Buffer L_e
- Buffer V_t
- Buffer loading



McKinley Clock Routing

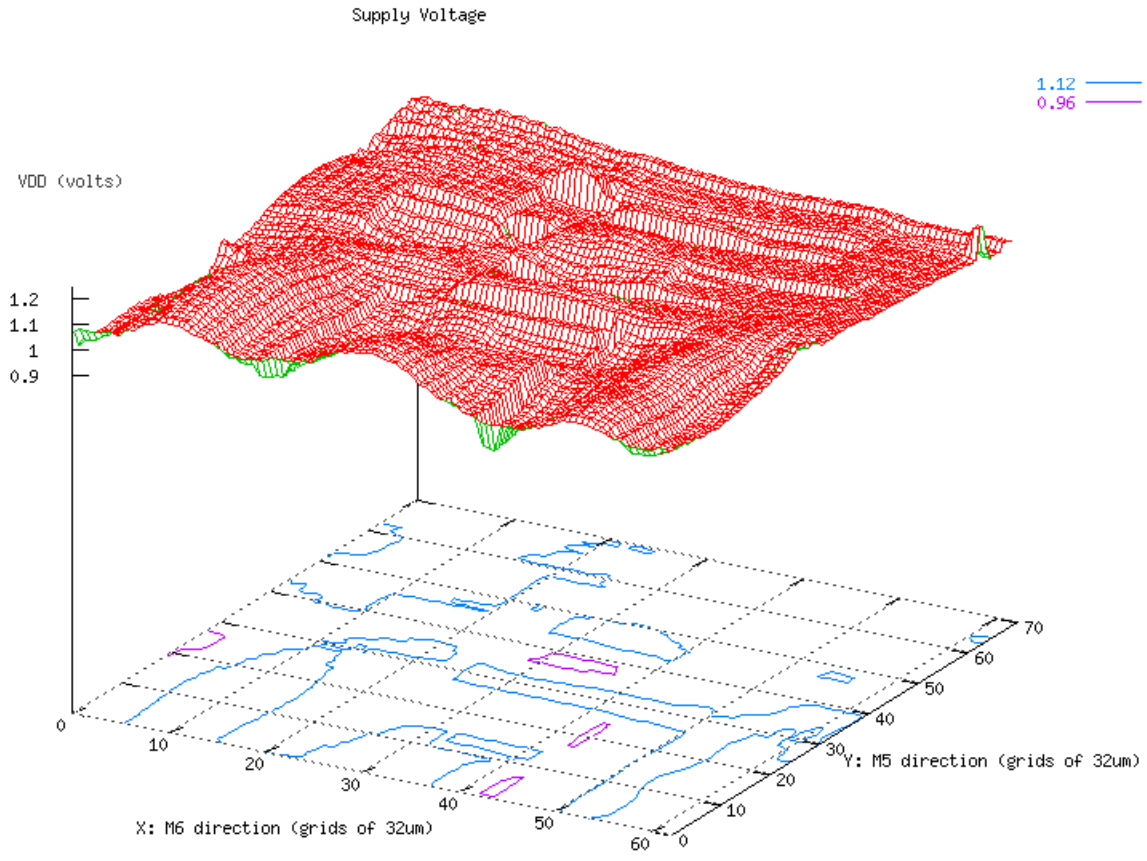


Source:
Paper 8.5
The Core Clock System for a Next-Generation Itanium Processor
Wells, Anderson, Berta

Voltage

Full chip power grid simulation: +/- 100 mV supply variation in time & space

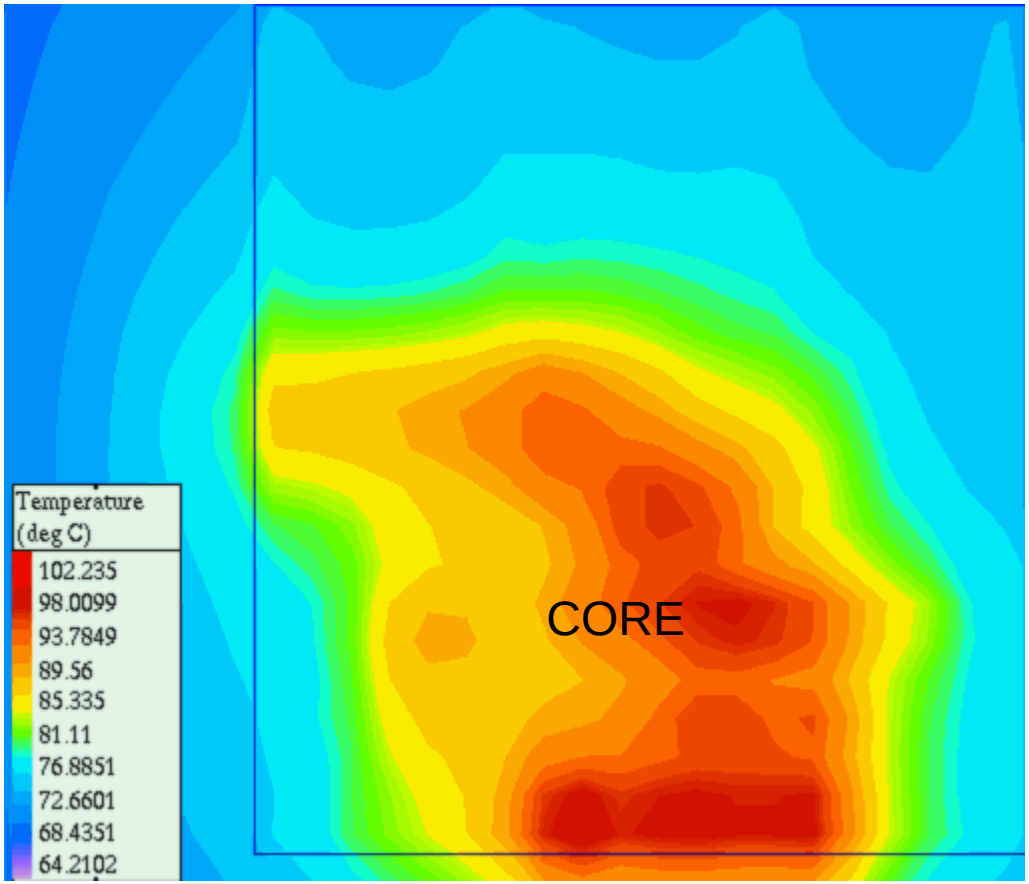
- Causes +/- 13% buffer delay variations (conservative)



Temperature

Full chip power and temperature simulation: 20 deg C variation across core

- Causes 1.5% variation in buffer delays



MOSFET Variations

180 nm bulk silicon process (S. Yang, IEDM 1998)

L_e: two components

- Systematic: half-range of 12.5 nm for transistors > 4mm apart
  - Less for nearby transistors
- Random: standard deviation of 3.3 nm between any transistors
- Net effect: 10% buffer delay variation (half-range)

V_t: inverse area dependence

- standard deviation in mV

	Narrow (W < 12.5 μm)	Wide (W > 12.5 μm)
NMOS	16.8	7.9
PMOS	14.6	6.5

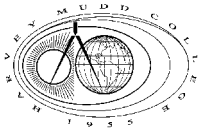
- Net effect: 2% buffer delay variation (standard deviation)

t_{ox}: difficult to distinguish effects

- lumped into L_e and V_t data

Skew Components

			Same Clock Edge (hold)				Cycle-to-Cycle (setup)			
Component	**Statistics**	**Type**	**Gater**	**SLCB**	**Repeater**	**PD**	**Gater**	**SLCB**	**Repeater**	**PD**
Primary Clock Driver Sources (150 ps PD Delay)										
PLL Jitter	half-range	Jitter					7.5	7.5	7.5	7.5
PD V	half-range	Jitter					19.5	19.5	19.5	19.5
PD T	half-range	Drift								
PD Le	half-range	Random								
PD Vt	stdev	Random								
Repeater Sources (150 ps Repeater Delay)										
PD to Repeater Wire Mismatch	half-range	Syst.				1				1
Repeater V	half-range	Jitter				19.5	19.5	19.5	19.5	19.5
Repeater T	half-range	Drift				1.1				1.1
Repeater Le	half-range	Random				15				15
Repeater Vt	stdev	Random				3				3
Repeater Load	half-range	Syst.				5				5
Second Level Clock Buffer Sources (250 ps SLCB Delay)										
Rep. to SLCB Wire Mismatch	half-range	Syst.			1.5	1.5			1.5	1.5
SLCB V	half-range	Jitter			36.4	36.4	36.4	36.4	36.4	36.4
SLCB T	half-range	Drift			2.1	2.1			2.1	2.1
SLCB Le	half-range	Random			28	28			28	28
SLCB Vt	stdev	Random			5.6	5.6			5.6	5.6
SLCB Load	half-range	Syst.			10	10			10	10
Gater Sources (180 ps Gater Delay)										
SLCB to Gater Wire Mismatch	half-range	Syst.		4	4	4		4	4	4
Gater V	half-range	Jitter		23.4	23.4	23.4	23.4	23.4	23.4	23.4
Gater T	half-range	Drift		1.4	1.4	1.4		1.4	1.4	1.4
Gater Le (global)	half-range	Random			18	18			18	18
Gater Le (local)	half-range	Random		9				9		
Gater Vt	stdev	Random		3.6	3.6	3.6		3.6	3.6	3.6
Gater Load	half-range	Syst.		7.5	7.5	7.5		7.5	7.5	7.5
Final Route Sources										
Local wire RC	half-range	Syst.	10	10	10	10	10	10	10	10



Data Delay Variation

An estimated number of nearly critical paths was taken from global timing analysis

- PD: 500
- Repeater: 500
- SLCB: 30000
- Gater: 3000

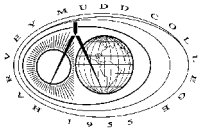
Same number of short paths requiring min-delay padding were assumed

Setup time constraint data delay variations

- High temperature, low voltage
- Margin: 0-50 ps slack (uniform)
- L_e variation: 65 ps (half-range)
- V_t variation: 8 ps (stdev)

Setup time constraint data delay variations

- Low temperature, high voltage
- Margin: 0-30 ps (uniform)
- Path delay variation: 30 ps uncertainty (stdev)



Skew Budget

A worst case (full range / 3 standard deviation) analysis predicts 495 ps skew!

- Unrealistically high and impossible to design with

Monte Carlo Analysis

- Choose median results for setup time
- Choose 95th percentile for hold time

	Gater	SLCB	Repeater	PD
Clock Skew Only				
t_{skew}^{setup}	232	267	302	312
t_{skew}^{hold}	20	106	229	280
Clock & Data Skew Combined				
t_{skew}^{setup}	285	309	313	322
t_{skew}^{hold}	44	117	226	275

Conclusions

Identify sources of skew impacting paths

- Different for setup and hold time calculations
- Different between clock domains

Use Monte Carlo simulation to estimate effect of skew on cycle time

In this example, jitter was the dominant source of setup skew

- Bypass and resistively isolate clock buffers to reduce jitter
- Thorough simulation of power grid noise gives less conservative results

For hold times, all paths must work. Be conservative in estimating skew.

For setup, the difference between global and local skew impacts design more than does the predicted value. Misestimation results in wasting effort optimizing the wrong paths.

Considering data delay variations reduces difference of global and local skews

- Without data delay, 80 ps difference between PD and Gater setup skew
- With data delay, 37 ps difference between PD and Gater setup skews