Latches with Dynamic Logic

Clock evaluates logic and opens subsequent latch:

Static signals driving dynamic logic must be either non-inverting or stable before evaluation
Latches with Dynamic Logic

Clock opens latch and evaluates subsequent logic:

Static signals driving dynamic logic must be either non-inverting or stable before latch opens.

L1 latch
L2 latch
\( \phi = 0 \)
\( \phi = 1 \)
Long path
Short path

Latches with Dynamic Logic

Clock evaluates logic and opens subsequent latch:

Static signals driving dynamic logic must be either non-inverting or stable before evaluation.

Phase1-domino precharges
Phase1-domino evaluates
Phase2-domino precharges
Phase2-domino evaluates
L1 latch
L2 latch
\( \phi = 0 \)
\( \phi = 1 \)
Short path
Latches with Dynamic Logic

Clock opens latch and evaluates subsequent logic:
Static signals driving dynamic logic must be either non-inverting or stable before latch opens.

Phase1-domino precharges
Phase2-domino precharges
Phase1-domino evaluates
Phase2-domino evaluates

L1 latch
L2 latch

Dynamic Logic with Latches

Edges become hard
Time available to logic is $P - 2T_{D-Q}$

From [Harris]
Two-Phase Clocking with Latches

$T_{ov}$ is the overlap time between the phases
– can be positive or negative

Duty cycles can be larger or smaller than 50%

Very common example is two-phase non-overlapping clocking

50% Duty Cycle

C1 and C2 are two ideal phases

Cycle boundary latches (CBL)
Mid-cycle latches (MCL)
Soft-Edge Properties of Latches

- **Slack passing** – logical partition uses left over time (slack) from the *previous* partition
- **Time borrowing** – logical partition utilizes a portion of time allotted to the *next* partition

Bernstein et al, Chapter 8, Partovi, Chap 11

Slack Passing

- **L1 phase time** is from the falling edge of C2 to the falling edge of C1
- **L2 phase time** is from the falling edge of C1 to the falling edge of C2
- **L1 phase delay** is the sum of S1 logic delay and L1 delay
- **L2 phase delay** is the sum of S2 logic delay and L2 delay
- Phase delays can be greater or less than phase times
Slack Passing

From [Bernstein et al]

\[ t_{S2} = t_{S2a} + t_{S2b} \]
\[ t_{S1} = t_{S1a} + t_{S1b} \]

L1 phase delay: \( t_{L1,phase} = t_{S2} + t_{L2} \)
L2 phase delay: \( t_{L2,phase} = t_{S1} + t_{L1} \)

Cycle delay: \( t_{cycle} = t_{L2,phase} + t_{L1,phase} \)
Cycle time: \( t_{cycle} = t_{L2} + t_{L1} \)
Slack Passing

---

Slack Passing Example

---

From [Bernstein et al]
Slack Passing Example (2)

From [Bernstein et al]

Slack Passing

\[ T_{L_{\text{max}}} = P + PW - T_{C_{\text{lk}} - Q} - T_{D - Q} - T_{SU} - T_{SK} - T_{JS} \]
Time Stealing

From [Bernstein et al]

Time Stealing Example

From [Bernstein et al]
Time Stealing Example (2)

From [Bernstein et al]

Skew-Tolerant Domino

- General Reference:
  Harris, Horowitz, “Skew-tolerant domino circuits”
  ISSCC’97, JSSC 11/97

Also slides from D. Harris’s Web site:
http://www3.hmc.edu/~harris/index.html
Domino Logic with Latches

Time available to logic is $P - 2T_{D-Q}$

Clock Skew

- Evaluation begins at latest rising edge
- Latch input setup before earliest falling edge
- Clock skew twice each cycle

Time penalty: $T_L = P - (2T_{D-Q} + 2T_{sk})$
Non-Balanced Phase Delays

Logic may not exactly fit half-cycle

- No flexibility to borrow time

Time penalty: $T_L = P - (2T_{D,Q} + 2T_{sk}) - T_{imb}$

Skew-Tolerant Domino

Overlap clocks:

- $x$ evaluates before $y$ precharges
- Implicit latch between $\phi_1$ and $\phi_2$
- No need for latch between domino phases

From [Harris]
Multiple Phases

In general: N overlapping clock phases

\[ T = t_e + t_p \]

- Identical, each delayed \( T/N \) from the previous

Precharge Phase

Precharge: \( t_p = t_{prech} + t_{skew} \)
**Evaluation Phase**

Overlapping: \( t_e = T/N + t_{skew} + t_{hold} \)

---

**Skew Tolerance**

We’ve found the optimal duty cycle:

- **Precharge:** \( t_p = t_{prech} + t_{skew} \)
- **Overlap:** \( t_e = T/N + t_{skew} + t_{hold} \)

Solve for the maximum tolerable skew:

\[
    t_{skew-max} = \frac{N-1}{N} \left( T - t_{prech} - t_{hold} \right) \frac{1}{2}
\]

- Skew tolerance increases with \( N \)

From [Harris]
Time Borrowing

- Excess overlap allows time borrowing into next phase (e.g., gate $X$)

$$t_{\text{borrow}} = t_{\text{skew-max, global}} - t_{\text{skew, global}} = \frac{N-1}{N} T_c - t_{\text{prech}} - t_{\text{hold}} - t_{\text{skew, local}} - t_{\text{skew, global}}$$