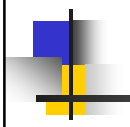


Gate Delay Calculation Considering the Crosstalk Capacitances

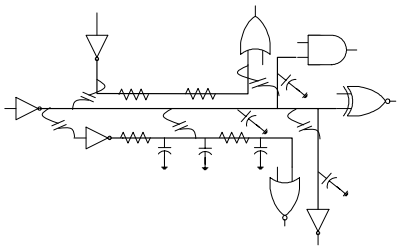
Soroush Abbaspour and Massoud Pedram
University of Southern California
Los Angeles CA

Asia and South Pacific Design Automation Conference 2004

Outline

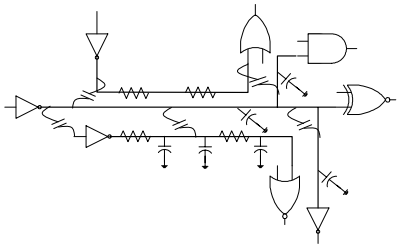
- Motivation
- Background
- Effective Capacitance Calculation
  - RC Loads
  - Coupled Capacitive Loads
  - Coupled RC Loads
- Conclusions

Motivation



- The stage delay in a VLSI circuit consists of the gate propagation delay and wire propagation delay.
- This paper focuses on the problem of calculating the gate propagation delay.
- For highest accuracy, we most carefully consider the effect of the resistive shielding and the capacitive coupling.

Motivation (Cont'd)



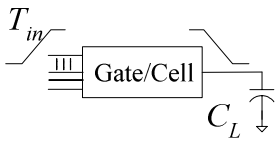
- Because of the complexity of the problem, the simplest way is to ignore:
  - coupling capacitance between interconnects, and
  - resistive shielding of the interconnects
- We only consider the capacitive loading on cell/gate propagation delay.

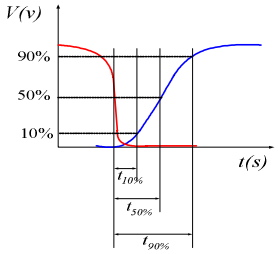
Gate Propagation Delay for Capacitive Loads

- In the case of a purely capacitive load, the gate propagation delay is a function of:
  - input transition time, and
  - output load.
- In commercial ASIC cell libraries, it is possible to characterize various output transition times as a function of the input transition time and output capacitance, i.e.,

$$t_\alpha = f_\alpha(T_{in}, C_L)$$

  - α% denotes the percentage of the output transition.
  - t_α is the output delay with respect to the 50% point of the input signal.
  - f_α is the corresponding delay function.





Gate Propagation Delay Algorithm: Capacitive Load

Draw_Output_Waveform (δ, T_{in}, C_L)

1. For α=10%, 50%, and 90% do

   t_α=**Calc_Delay** (δ, T_{in}, C_L, Table(T_{in}, C_L, α))

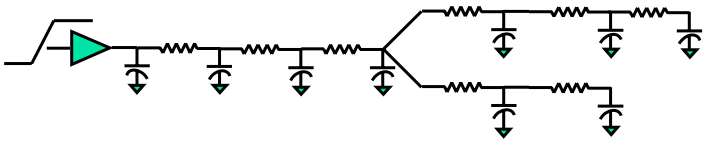
2. Draw the output waveform according to above data.

Calc_Delay (δ, T_{in}, C_L, Table(T_{in},C_L,α))

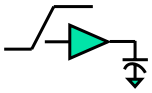
1. From Table(T_{in},C_L,α) according to T_{in} and C_L, find the 50% input to α% output propagation delay, add δ to this value, and call it t_α
2. Return t_α

Resistance Shielding and Effective Capacitance Approach

- In VDSM technologies, we cannot neglect the effect of interconnect resistances of the load.



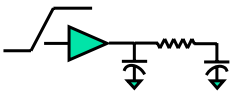
- Using the sum of all load capacitances as the capacitive load provides an overly pessimistic approximation.



Sum of all capacitive load

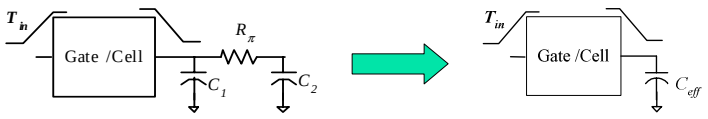
Resistance Shielding

- A more accurate approximation for an n^{th} order load seen by the gate/cell (i.e., a load with n distributed capacitances to ground) is to use a second order RC-π model.



RC-π load for effective capacitance algorithms

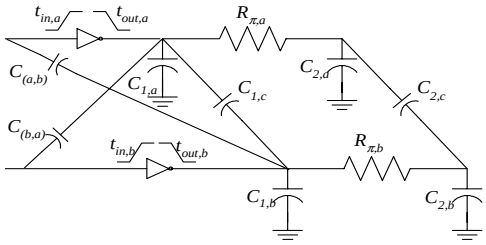
- For efficient and accurate gate delay calculation, we then convert the second order RC-π model into an effective capacitance value and use that value as the output load of the gate.



$C_{eff} = C_1 + kC_2$ where $0 \le k \le 1$

Crosstalk Capacitance

- As technology scales, the effect of coupling capacitances becomes more noticeable.
- Given a complex load with two types of capacitive couplings (load-to-load and input-to-load couplings), one can use moment matching techniques to model the load and the parasitic elements as an RC network (see below).
- Notice that the gate propagation delay calculation becomes challenging, potentially requiring large multi-dimensional lookup tables.



Outline

- Motivation
- Background
- Effective Capacitance Calculation
  - RC Loads
  - Coupled Capacitive Loads
  - Coupled RC Loads
- Conclusions

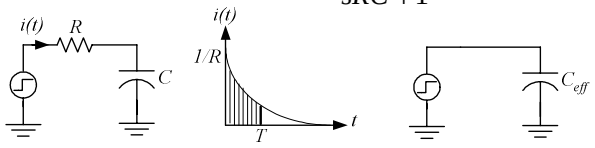
Prior Work

- C. Ratzlaff, S. Pullela, and L. Pillage, "Modeling the RC Interconnect effects in a Hierarchical Timing Analyzer," CICC 1992.
- M. Sriram and S. M. Kang, "Fast Approximation of the Transient Response of Lossy Transmission Line Trees," DAC 1993.
- R. Macys, S. McCormick, "A New Algorithm for Computing the "Effective Capacitance" in Deep Sub-micron Circuits," CICC 1998,.
- C.V. Kashyap, C.J. Alpert, A. Devgan, "An effective capacitance based delay metric for RC interconnect" ICCAD 2000.
- S. Abbaspour, M. Pedram, "Calculating the effective capacitance for the RC interconnect in VDSM technologies," ASP-DAC 2003.

A New Effective Capacitance Calculation Algorithm

- Consider a unit step voltage source that drives an RC circuit. The current flowing into the *RC* circuit in Laplace domain is calculated as:

$$I(s) = V(s)Y(s) = \frac{C}{sRC+1}$$



a) RC element b) Current waveform c) Effective capacitiance

- By calculating the total charge induced into the capacitance up to time T and equating with the total charge induced into the corresponding effective capacitance, we can write:

$$C_{eff} = C(1 - e^{\left(-T/RC\right)})$$

Effective Capacitance Calculation (Cont'd)

- Similarly, the effective capacitance for an RC-π model load can be written as:

$$C_{eff} = C_1 + (1 - e^{(-k\,t_{out}/R_\pi C_2)})C_2$$

  - k is a dimensionless constant
  - t_{out} is the gate output transition time.

- As in Macys' work, we define:

$$\alpha = C_1 / (C_1 + C_2) \qquad \beta = t_{out} / R_\pi C_2 \qquad \gamma = C_{eff} / (C_1 + C_2)$$

- Based on Macy's output-transition-time and gate-configuration independent table that relates these three parameters, we can rewrite the effective capacitance equation as:

$$\frac{C_{eff}}{C_1 + C_2} = \gamma = \alpha + \left(1 - e^{-k\beta}\right)(1 - \alpha)$$

Effective Capacitance Calculation (Cont'd)

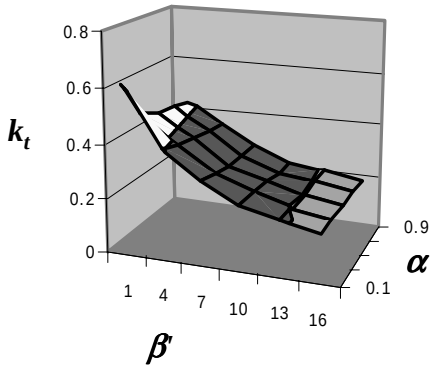
- If we replace the output transition time (t_{out}) by 50% propagation delay we can rewrite the equation as:

$$\frac{C_{eff}}{C_1 + C_2} = \gamma = \alpha + \left(1 - e^{-k_t \beta'}\right)(1 - \alpha)$$

  - where β' is the ratio between the 50% propagation delay and the $R_\pi C_2$ product.
  - k_t is a fixed value which can be obtained from a lookup table (compiled from circuit simulation results), and which is constant for the calculated α and β'.

Effective Capacitance Calculation (Cont'd)

- This plot shows k_t values for 50% propagation delays for different α and β' values in a 0.1μm CMOS technology.



Advantage of The Proposed Equation to Macys' Equation

- An analytical expression for effective capacitance
- Our approach results in a more stable effective capacitance estimation.

$$H_{\gamma}^{t_{out}} = \frac{\frac{\partial t_{out}}{t_{out}}}{\frac{\partial \gamma}{\gamma}} = \frac{(\partial t_{out})/(t_{out})}{\left(\partial \frac{C_{eff}}{C_1+C_2}\right)/\left(\frac{C_{eff}}{C_1+C_2}\right)} = = \frac{\frac{\partial t_{out}}{t_{out}}}{\frac{\partial C_{eff}}{C_{eff}}} = \frac{\partial t_{out}}{\partial C_{eff}} \cdot \frac{C_{eff}}{t_{out}}$$

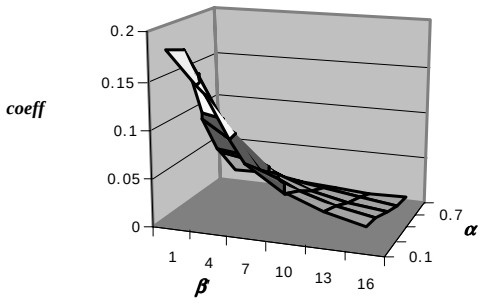
$$H_{k_t}^{t_{out}} = \frac{\frac{\partial t_{out}}{t_{out}}}{\frac{\partial k_t}{k_t}} = \frac{\partial t_{out}}{\partial k_t} \cdot \frac{k_t}{t_{out}} = \frac{\partial t_{out}}{\partial C_{eff}} \cdot \frac{\partial C_{eff}}{\partial k_t} \cdot \frac{k_t}{C_{eff}} \cdot \frac{C_{eff}}{t_{out}}$$

$$= \frac{\partial t_{out}}{\partial C_{eff}} \cdot \frac{C_{eff}}{t_{out}} \cdot \frac{\partial C_{eff}}{\partial k_t} \cdot \frac{k_t}{C_{eff}} = H_{\gamma}^{t_{out}} \cdot \frac{\partial C_{eff}}{\partial k_t} \cdot \frac{k_t}{C_{eff}}$$

$$\Rightarrow coeff \square \frac{\partial C_{eff}}{\partial k_t} \cdot \frac{k_t}{C_{eff}} = \frac{k_t{}^2 \beta'(1-\alpha)e^{-k_t\beta'}}{\alpha + (1-e^{-k_t\beta'})(1-\alpha)}$$

Advantage (Cont'd)

- This coefficient is always less than 1. Therefore our output transition time equation is less sensitive to parameter error



Gate Propagation Delay Algorithm: RC Load

Draw_for_*RCπ*_Load (T_{in}, Load Parameters)
1. For α=10, 50, and 90 do
   Find_Transition_Point (T_{in}, C_1, R_π, C_2, Table (50%-α%),Table(k_t))
2. Draw output waveform according to the results

Find_Transition_Point (T_{in} , C_1,C_2,R_π,Table(50%-α%), Table(k_t))
1. Guess an initial value for C_{eff}
2. Compute α value (Macys' notation) from the load values
3. Obtain t_α from *Table(50%-α%)* based on values of C_{eff} and T_{in}
4. Compute β' from t_α and load elements
5. Find k_t from *Table(k_t)* according to α and β'
6. Calculate C_{eff}
7. Find the new value of t_α for the obtained C_{eff} from *Table(50%-α%)*
8. Compare the new t_α with the old t_α
9. If not within acceptable tolerance, then return to step 4 until t_α converges
10. Return t_α

Proof of Convergence

- Theorem 1: The iterative algorithm, always converges independently of the initial guess. Furthermore, its solution is unique.
- Proof: Because

$$\left|\frac{d\left(k_t\beta'\right)}{dC_{eff}}\left(e^{-k_t\beta'}\right)C_2\right| < \left|\frac{d\left(k_t\beta'\right)}{dC_2}\cdot\frac{dC_2}{dC_{eff}}\left(e^{-k_t\beta'}\right)C_2\right|$$

$$= \left|k_t\beta'\frac{dC_2}{dC_{eff}}\left(e^{-k_t\beta'}\right)\right| \cong \left|k_t\beta'\left(e^{-k_t\beta'}\right)\right| < 1 \qquad for\ \forall(k_t\beta') > 0$$

- Therefore,

$$\left|\frac{d}{dC_{eff}}\left(C_1 + (1-e^{-k_t\beta'})C_2\right)\right| = \left|\frac{d\left(k_t\beta'\right)}{dC_{eff}}\left(e^{-k_t\beta'}\right)C_2\right| < 1$$

Experimental Results

Table 1: Simulation results for output waveform evaluation algorithm for RCπ load (proposed in Section 2) for 0.1µm technologies (3 iterations).

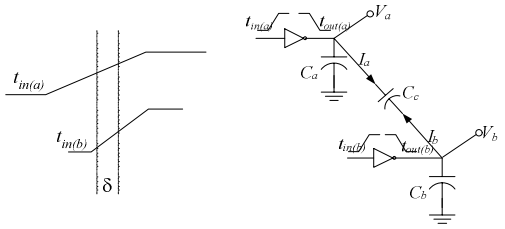
Driver and Load Parameters					10% propagation delay (from 50% of input to 10% of output)			50% propagation delay (from 50% of input to 50% of output)			90% propagation delay (from 50% of input to 90% of output)		
T_{in} (pS)	$W_P(\lambda)/W_n(\lambda)$	C_1(fF)	R(Ω)	C_2(fF)	Our Approach (pS)	Hspice (pS)	Error%	Our Approach (pS)	Hspice (pS)	Error%	Our Approach (pS)	Hspice (pS)	Error%
400	100/50	500	100	900	169	165	2.4	711	703	1.1	1705	1696	0.5
200	150/75	1500	200	400	147	144	2.1	640	635	0.8	1515	1491	1.6
100	50/25	250	80	1000	190	186	2.1	1186	1168	1.5	2960	2890	2.4
100	150/75	1000	500	1250	90	87	3.4	486	476	2.1	1825	1802	1.3
600	80/40	350	150	1000	199	193	3.1	845	834	1.3	2095	2080	0.7
300	200/100	1000	450	1650	102	105	2.8	401	392	2.3	1650	1647	0.1
50	200/100	1500	650	2050	85	84	1.2	461	456	1.1	2047	2037	0.4
250	20/10	450	350	650	426	419	1.7	2586	2572	0.5	6420	6348	1.1
150	160/80	850	1000	1500	86	82	4.9	349	336	3.9	1642	1621	1.3
550	150/75	1300	400	1900	185	181	2.2	721	707	2.0	2560	2495	2.6
350	120/60	1600	500	1500	205	203	0.9	1012	1000	1.2	3145	3097	1.5
450	30/15	500	100	600	149	142	4.9	515	503	2.4	1190	1149	3.7
Avg.							**2.6**			**1.7**			**1.4**

***All inputs are rising.

Outline

- Motivation
- Background
- Effective Capacitance Calculation
  - RC Loads
  - Coupled Capacitive Loads
  - Coupled RC Loads
- Conclusions

Problem Statement



- Two CMOS drivers, a and b, are given where their corresponding input transition times are $t_{in}(a)$ and $t_{in}(b)$
  - δ_a and δ_b denote the 50% transition points of the input waveforms of driver a and b.
  - There is a $\delta=\delta_b-\delta_a$ delay between their input waveforms
  - Furthermore, the output waveform of drivers a and b are $t_{out}(a)$ and $t_{out}(b)$, respectively.
- The objective is to find the output waveforms of the two drivers.

Problem Statement (Cont'd)

- In fact, we must solve a nonlinear equation:

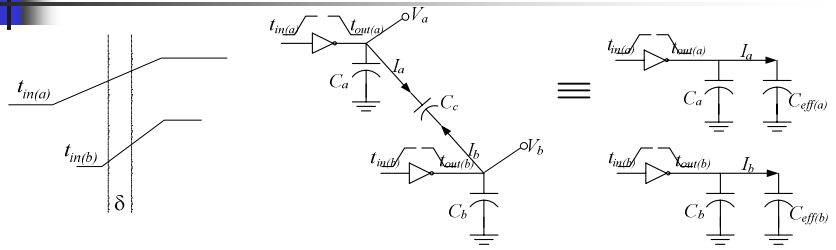
$$S(T_{in}, C_L, C_C, T_{out}) = 0$$

where

$$T_{in} = \begin{bmatrix} t_{in(a)} \\ t_{in(b)} \end{bmatrix} \quad T_{out} = \begin{bmatrix} t_{out(a)} \\ t_{out(b)} \end{bmatrix} \quad C_L = \begin{bmatrix} C_a \\ C_b \end{bmatrix} \quad C_C = \begin{bmatrix} C_c \end{bmatrix}$$

which is a daunting task. So we must look for a different approach.

Miller Capacitance Based Solution



- Equating the current sources in grounded capacitance and the coupling capacitance, we end up:

$$C_{eff,a} = C_a + C_c\left(1 - \frac{\Delta V_b}{V_{th,a}}\right) \qquad where \quad \Delta V_b = V_b^1 - V_b^0$$

$$C_{eff,b} = C_b + C_c\left(1 - \frac{\Delta V_a}{V_{th,b}}\right) \qquad where \quad \Delta V_a = V_a^1 - V_a^0$$

where $V_{th,a}$ identifies the transition point of interest, say $V_{th,a}=0.5V_{dd}$ and ΔV_b is the output voltage transition of driver **b** from when the output waveform of driver **a** transits from 0 to its transition point.

Gate Propagation Delay Algorithm: Coupled Capacitive Load

Find_Waveforms_Capacitive_Cross_Coupled($\delta(a),\delta(b),t_{in}(a),\ t_{in}(b),\ C_a,\ C_b,\ C_c$)

1. Guess an initial value for output capacitive load as $C_L(a)$ and $C_L(b)$
2. $t_{out,a}$=**Draw_Output_Waveform**($\delta(a),t_{in}(a),C_L(a)$)
3. $t_{out,b}$=**Draw_Output_Waveform**($\delta(b),tin(b),C_L(b)$)
4. Repeat until the output waveform converges
   For ($V_{th,a}$,$V_{th,b}$)={(50%,50%),(50%,90%),
   (90%,50%), (50%,10%), (10%,50%)} do
   Find_Output($t_{out,a}$,$t_{out,b}$,$V_{th,a}$,$V_{th,b}$,C_L(a), C_L(b),C_c)

Find_Output($t_{out,a}$,$t_{out,b}$,$V_{th,a}$, $V_{th,b}$, C_L(a), C_L(b), C_c)

1. $C_{L(a)} \leftarrow C_{L(a)} + C_C(1-\frac{\Delta V_b}{V_{th,a}})$ and $C_{L(b)} \leftarrow C_{L(b)} + C_C(1-\frac{\Delta V_a}{V_{th,b}})$
2. Update $t_{out,a}$ and $t_{out,b}$
3. If $t_{out,a}$ and $t_{out,b}$ *tolerance* are within acceptable range, then return $t_{out,a}$ and $t_{out,b}$
4. **Find_Output** ($t_{out,a}$,$t_{out,b}$,$V_{th,a}$,$V_{th,b}$,C_L(a),C_L(b),C_c)

Proof of Convergence

- **Theorem 2:** The "Find_Waveforms_Capacitive_Cross_Coupled" algorithm converges to its unique solution independently of the initial guess for the value of the effective capacitance to ground.
- **Proof:** the algorithm can be written as :

$$\begin{cases} T_{out} = F(T_{in}, C_L + C_{EFF}) \\ C_{EFF} = G(C_C, T_{out}) \end{cases} \Rightarrow T_{out} = F(T_{in}, C_L + G(C_C, T_{out}))$$

- which is equivalent to prove:

$$\left| \frac{\partial}{\partial T_{out}} \left(F(T_{in}, C_L + G(C_C, T_{out})) \right) \right| < 1$$

- where:

$$G(C_C, T_{out}) = C_c \begin{bmatrix} 1 - k_{g1} t_{out(a)} \\ 1 - k_{g2} t_{out(b)} \end{bmatrix} \qquad F(T_{in}, C_L + C_{eff}) = \begin{bmatrix} k_{f1}\left(C_a + C_c\left(1 - k_{g1} t_{out(a)}\right)\right) \\ k_{f2}\left(C_b + C_c\left(1 - k_{g2} t_{out(b)}\right)\right) \end{bmatrix}$$

Proof of Convergence (Cont'd)

- Therefore, it can be inferred :

$$\left|\frac{\partial}{\partial T_{out}}\left(F(T_{in}, C_L + G(C_C, T_{out}))\right)\right| = \left|\begin{bmatrix} \frac{\partial F_a}{\partial t_{out(a)}} & \frac{\partial F_a}{\partial t_{out(b)}} \\ \frac{\partial F_b}{\partial t_{out(a)}} & \frac{\partial F_b}{\partial t_{out(b)}} \end{bmatrix}\right|$$

$$= \left|\frac{\partial F_a}{\partial t_{out(a)}} \cdot \frac{\partial F_b}{\partial t_{out(b)}} - \frac{\partial F_a}{\partial t_{out(b)}} \cdot \frac{\partial F_b}{\partial t_{out(a)}}\right| = \left|k_{f1} k_{f2} C_c^{\,2} k_{g1} k_{g2}\right|$$

- The worst case values confirms above inequality, which turns out the correctness of the theorem.

Experimental Results

Table 2: Simulation results for capacitive load considering crosstalk (cf. section 3) for 0.1µm technologies (3 iterations).

Driver A			Driver B			Load Parameters			50% propagation delay of driver A			50% propagation delay of driver B		
T_{in} (ps)	$W_p(\lambda)/W_n(\lambda)$	50% Coord. (pS)	T_{in} (ps)	$W_p(\lambda)/W_n(\lambda)$	50% Coord. (ps)	Ca (fF)	Cb (fF)	Cc (fF)	Our Appoach (*pS*)	Hspice (*pS*)	Error%	Our Approach (*pS*)	Hspice (*pS*)	Error%
100	100/50	50(R)	200	150/75	150 (F)	500	400	500	645	637	1.3	465	458	1.5
100	100/50	50(R)	200	150/75	250 (F)	500	400	500	570	550	3.6	475	464	2.4
100	100/50	50(F)	200	150/75	150 (F)	500	400	500	281	271	3.7	185	177	4.5
200	150/75	100(F)	350	250/125	750 (F)	1000	900	1500	705	697	1.1	253	241	4.9
200	150/75	100(R)	350	250/125	750 (F)	1000	900	1500	602	595	1.2	681	670	1.6
350	120/60	175(F)	150	80/40	650 (R)	700	800	500	521	505	3.2	995	975	2.1
350	120/60	175(R)	150	80/40	650 (R)	700	800	500	415	405	2.5	402	391	2.8
400	250/125	200(R)	350	120/60	500 (R)	1100	900	1000	341	325	4.9	286	279	2.5
400	250/125	200(F)	350	120/60	900 (R)	1100	900	1000	356	346	2.9	1189	1149	3.5
Avg.											**2.7**			**2.9**

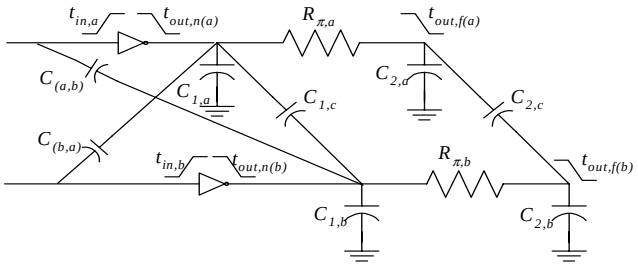
***F: Falling input, R: Rising input

Outline

- Motivation
- Background
- Effective Capacitance Calculation
  - RC Loads
  - Coupled Capacitive Loads
  - Coupled RC Loads
- Conclusions

Problem Statement

- **Problem Statement:** The problem statement is the same as the one in for coupled capacitive loads, except that the load is now the one that is depicted in following figure. We are interested in determining the output waveforms at the near ends.



Gate Propagation Delay Algorithm: Coupled RC Loads

Find_Output_Waveforms ($t_{in,a}$, $t_{in,b}$, Load Parameters)

1. Model each coupling capacitance as a capacitance to ground
2. ($t_{out,n}(a)$,$t_{out,f}(a)$)=**Draw_for_RCπ_Load**($t_{in,a}$, Load Parameters)
3. ($t_{out,n}(b)$,$t_{out,f}(b)$)=**Draw_for_RCπ_Load**($t_{in,b}$, Load Parameters)
4. Repeat
   For ($V_{th,a}$, $V_{th,b}$)={(50%,50%),(50%,90%), (90%,50%), (50%,10%), (10%,50%)} do
   Update_Voltage_Waveforms (Voltage Waveforms, $V_{th,a}$, $V_{th,b}$, Load Parameters)
5. Until the output waveforms converges

Update_Voltage_Waveforms (Voltage Waveforms, $V_{th,a}$, $V_{th,b}$, Load Parameters**)**

1. Update equivalent Miller capacitance values
2. **Draw_for_RCπ_Load (**$t_{in,a}$, Load Parameters**)**
3. **Draw_for_RCπ_Load (**$t_{in,b}$, Load Parameters**)**
4. If voltage waveforms are within acceptable tolerance, then return values
5. **Update_Voltage_Waveforms (**Voltage Waveforms, $V_{th,a}$, $V_{th,b}$, Load Parameters**)**

Experimental Results

Table 3: Simulation results for general load considering crosstalk (cf. Section 4) for 0.1µm technologies (3 iterations).

Driver A			Driver B			Load Parameters										50% propagation delay of driver A			50% propagation delay of driver B		
T_{in} (ps)	$W_p(\lambda)$/ $W_n(\lambda)$	50% Coor. (pS)	Tin (ps)	$W_p(\lambda)$/ $W_n(\lambda)$	50% Coor. .(pS)	$C_{1,a}$	$R_{1,\pi}$	$C_{2,a}$	$C_{1,b}$	$R_{2,\pi}$	$C_{2,b}$	$C_{1,c}$	$C_{2,c}$	$C_{(a,b)}$	$C_{(b,a)}$	Our App. (pS)	Hspice (pS)	Err. (%)	Our App. (pS)	Hspice (pS)	Err. (%)
100	100/50	50(R)	200	150/75	150(F)	300	100	800	800	300	999	400	500	100	150	675	630	7.1	1109	1050	5.6
100	100/50	50(F)	200	150/75	250(F)	600	200	400	500	400	800	450	650	200	350	985	945	4.2	470	442	6.3
200	150/75	100(F)	350	250/125	750(F)	500	300	400	600	200	900	350	250	300	100	717	688	4.2	485	466	4.1
350	120/60	175(F)	150	80/40	650(R)	600	500	500	400	200	500	450	500	350	400	535	512	4.5	1430	1390	2.9
400	250/125	200(R)	350	120/60	500(R)	700	350	900	500	300	600	350	250	450	150	513	489	4.9	522	495	5.4
200	150/75	100(R)	350	250/125	750(F)	800	500	950	450	350	900	250	450	350	300	511	489	4.5	407	391	4.0
400	250/125	200(F)	350	120/60	900(R)	250	150	650	350	125	800	650	350	125	275	312	291	7.2	1345	1296	3.8
Avg.																		5.2			4.6

**F: Falling input, R: Rising input

Conclusion

- Gate delays can vary widely as a function of input slews, driver sizes, input transition time skews, and output loads.
- We presented three efficient iterative algorithms with provable convergence property for calculating the effective capacitance.
- The error for calculating the gate propagation delay is quite small 1-6% depending on the complexity of the load and coupling configuration.