With the rapid move to ultra deep sub-micron (UDSM) designs and feature size processes of 0.18 micron and below, ensuring the integrity of signals as they traverse conductors on a chip is becoming a challenge. Crosstalk between signals, due to increased capacitive coupling, is the most severe problem that impacts the timing of signals on a chip causing functional failures and performance degradation.

1 What is Crosstalk?

If there are any two wires running close together on a chip, they experience interaction of three kinds:

- Resistive Coupling: The material between the interconnects is a good insulator, so this sort of coupling is negligible.
- Inductive Coupling: The wires are likely to have some self inductance and some mutual inductance. Inductance causes oscillations in the output waveform. The effect is usually small and is thus ignored.
- Capacitive Coupling: The two wires will have coupling capacitance (essentially like any parallel plate capacitor) between them. To control the die size, the width of the metal is continuously being decreased whereas to keep the metal resistance low, its height is being increased. Furthermore, the wire lengths are longer. Thus, wires have now become longer and thinner and as a result the wire to wire capacitance has increased. With more and more interconnect layers on the chip, the distance from the higher metal layers to the substrate increases, thereby decreasing the substrate capacitance component of the total net capacitance. Capacitive coupling is the dominant contributor to interconnect delay.

Coupling can cause noise leading to spurious logic transitions as well as change in delay (both increase and decrease in delay).

2 Capacitive Coupling

Charge sharing model is a simple and commonly used crosstalk noise model. Consider Figure 2. Suppose $C_1$ is charged to $V_1$ when switch $S$ is open. After $S$ is closed, the total charge is conserved and hence shared between $C_1$ and $C_2$. The resulting voltage $V$ is then given by the expression:

$$C_1 V_1 = V(C_1 + C_2)$$

Figure 1: Two coupled lines. $C_0$ is the ground capacitance while $C_c$ is the coupling capacitance.
Now a similar technique can easily be applied to Figure 2 when the aggressor voltage is $V_A$ and the victim $V$ is quiet $^1$. This gives:

$$V_V = V_A \left( \frac{C_C}{C_C + C_0} \right)$$  \hspace{1cm} (2)$$

Note that this model is not accurate as it ignores victim and aggressor resistances completely but it is simple to calculate. If there are multiple aggressors, an estimate of the total $V_V$ can be got by a simple sum of the various noise voltages induced by the aggressors. From Equation 2, it is clear that even when the victim line is quiet (i.e. not having any driving voltage), a noise voltage $V_V$ gets induced into it due to capacitive coupling with the aggressors. If this noise voltage exceeds the logic threshold voltage, a logic transition may be caused on the gate begin driven by the victim line.

Next lets consider the effect of coupling on delay. We will take delay as a simple product of total resistance and capacitance of the wires $^2$. Equation 2 is not simple to handle. We would like to have an expression where we can handle each line separately without bothering about the second coupled line. To do this we need an equivalent circuit as in Figure 3. Also for sake of simplicity assume that all voltage transitions begin at the same time. i.e. there are no offsets in transitions. Then we have three cases:

1. $V_A$ and $V_V$ having opposite transitions: If $R_A = R_V$ and $C_A = C_V$, then $\frac{dV_A}{dt} = -\frac{dV_V}{dt}$. The current through $C_C$ is given by $C_C \left( \frac{dV_A}{dt} + \frac{dV_V}{dt} \right)$. The same current should flow through equivalent capacitor $C_{CV}$. But the current through $C_{CV}$ is given by $C_{CV} \frac{dV_V}{dt}$. So we get

$$C_{CV} = C_C \left( 1 + \frac{\frac{dV_A}{dt}}{\frac{dV_V}{dt}} \right)$$  \hspace{1cm} (3)$$

   In case, when rise(fall) times of $V_A$ and $V_V$ are equal or both are step inputs, then the above equation reduces to $C_{CV} = 2C_C$. This is the commonly used worst case coupling capacitance value (though this is not correct as we will see in the last

---

$^1$Usually, when talking about crosstalk, we talk about aggressors and a victim. Victim is the line on which we want to calculate the coupling effect while aggressors are all the other lines which couple with the victim.

$^2$This is not very accurate as the resistance and capacitance in case of a wire are distributed rather than lumped.
section). The number with which $C_C$ gets multiplied to obtain the equivalent capacitance $C_{CV}$ (here 2) is sometimes referred to as the switching factor.

2. $V_A$ and $V_V$ having same transitions: In a similar fashion as above we can obtain the switching factor in this case as $(1 - (dV_A/dt)/(dV_V/dt))$. Note that this can be negative, meaning that coupling actually decreases the delay rather than increasing it.

3. $V_A$ is quiet: In this case $C_{CV} = C_C$ simply.

The fourth possible case when $V_V$ is quiet does not make sense in case of calculation of delay on the victim line. This case is considered just for noise analysis and has already been dealt with in Equation 2. The delay due to coupling usually accounts for more than half of the total interconnect delay in modern VLSI circuits. Usually, an arrival time window (i.e. earliest when a signal should arrive and latest when a signal should arrive) is specified for the sinks of any interconnect \(^3\). So, reduction as well as increase in delay due to coupling can prove to be fatal to correctness of any circuit.

3 More Complications

The above analysis of crosstalk noise and delay is very simplistic (actually too simplistic to be used in practice for modern designs). The aggressor and victim may have different rise times, may arrive at different times (so there might not be a perfect overlap between victim and aggressor waveforms). The absolute worst case can be much worse than a switching factor of 2. Charge sharing model for coupling noise is also an oversimplification. Also the interconnects have distributed $R,C$ so they have to be treated more like transmission lines, than simple resistance and capacitance. More accurate methods using circuit simulators like *Spice* are too time consuming. More than the actual performance degradation, it is the uncertainty caused by incorrect estimation of crosstalk effects which troubles the VLSI designers. Crosstalk estimation still remains an active research problem.

Since capacitive coupling is the major contributor to signal delay as well signal integrity, another area of research is crosstalk reduction techniques. For example a simple crosstalk reduction technique employed is to place a wire which does not switch (e.g. ground or power line) between two wires which switch a lot. As the coupling capacitance rapidly decreases with increasing distance, the middle wire essentially acts as a shield reducing the coupling between the two switching wires.

4 Questions?

If you have any questions you can email me at puneet@ucsd.edu.

\(^3\)The upper bound arises from setup time constraints while the lower bound arises from hold time constraints.