Introduction

- So far, have considered transistor-based logic in the face of technology scaling
- Interconnect effects are also of concern
  o Can impact speed
  o Can significantly impact energy consumption in a digital integrated circuit
    • (Can also think of in terms of clock distribution network – for example)
- Aggregate effects of interconnect can be even worse because larger die sizes exacerbate the above problems

Interconnect Parasitics

- Wiring of today’s on-chip interconnect (IC) gives rise to:
  o Capacitive parasitics
  o Resistive parasitics
  o Inductive parasitics

- All parasitics:
  o Can cause increase in propagation delay
  o Can adversely impact energy dissipation and power distribution
  o Can introduce extra noise sources which effect reliability

- This is a hard problem to model – interconnect is everywhere so places all over the chip are sources of the aforementioned problems; from modeling perspective, simplifications could be considered – for example:
  o Ignore inductive effects if resistance R of wire is high (i.e. the wire is long or has a small cross section) OR rise and fall times are low
  o If the wire is short OR the cross-section is high OR IC material has low resistivity, one might only use a capacitive model
  o If separation between neighboring wires is high, could ignore inter-wire capacitances

Capacitance:

Picture: wire-to-substrate and wire-to-wire capacitances

Wire-to-substrate: \[ C = \frac{\varepsilon_{di}}{t_{di}} WL \]

Wire-to-wire: \[ C = \frac{\varepsilon_{di}}{d} HL \]
Resistance:

- The resistance of a wire is proportional to its length $L$ and inversely proportional to its cross-sectional area $A$

$$R = \frac{\rho L}{WH}$$

  - $\rho$ is the resistivity of the wire in $\Omega$ meters; example values include:
    - Cu: $1.7 \times 10^{-8} \Omega$ meters
    - Al: $2.7 \times 10^{-8} \Omega$ meters

- Transitions between routing layers (through vias) can result in additional resistance
  - **Slide**: Metal layers
  - This resistance can be reduced by increasing via size
    - But, current can crowd around the perimeter of the via; this effect can eventually reduce the effectiveness of this design technique
  - Example point of reference:
    - In 250 nm technology, AL contacts $\sim 5-20 \, \Omega$ for metal to poly and $1-5 \, \Omega$ for metal-to-metal
  - **Quantitative Example**:
    - CMOS, Nanomagnetic Logic clock

Inductance:

- Effects, consequences include: noise, reflections, inductive coupling

- Changing current passing through an inductor generates a voltage drop: $\Delta V = L \frac{di}{dt}$

**Interconnect in the face of device scaling**

- If transistor-based logic scales, interconnect must scale too

- Let's consider a transistor-like IC scaling model:
  - Could start with an ideal scaling factor $S$ (as before), but length does not scale well

- Generally speaking:
  - **Local IC** scales with transistors
  - **Global IC** does not scale well
    - Global IC includes connectivity between large modules, I/O, the clock distribution network, etc.
    - As transistor sizes scale, the clock goes to more transistors
    - Another complication (was) die size ... was increasing $\sim 6\%$ per year and now $2X$ per decade
      - Has slowed down. **Any thoughts as to why?**
- In scaling models, must differentiate between local and global wires; gives rise to 3 scaling models:
  1. Local wires: \( S_L = S > 1 \)
  2. Constant length wires: \( S_L = 1 \)
  3. Global wires: \( S_L = S_c < 1 \)

(of course, < 1 means that global wires do not scale well)

- A first order approximation of scaling

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Relation</th>
<th>Local</th>
<th>Constant</th>
<th>Global</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W, H, t )</td>
<td>( 1/S )</td>
<td>( 1/S )</td>
<td>( 1/S )</td>
<td></td>
</tr>
<tr>
<td>( L )</td>
<td>( 1/S )</td>
<td>1</td>
<td>( 1/S_c )</td>
<td></td>
</tr>
<tr>
<td>( C )</td>
<td>( LW/t )</td>
<td>( 1/S )</td>
<td>1</td>
<td>( 1/S_c )</td>
</tr>
<tr>
<td>( R )</td>
<td>( L/WH )</td>
<td>( S )</td>
<td>( S^2 )</td>
<td>( S^2/ S_c )</td>
</tr>
<tr>
<td>( RC )</td>
<td>( L^2/Hz )</td>
<td>1</td>
<td>( S^2 )</td>
<td>( S^2/ S_c^2 )</td>
</tr>
</tbody>
</table>

See slides + note my board comments

- Take aways:
  o Technology scaling does not reduce wire delay (see RC time constant)
  o Constant delay predicted for local wires
  o Delay of global wires increases
    ▪ More logic, more capacitance, more layers of metal, necessary smaller geometries
  o No perfect solutions; for example:
    ▪ Try to scale wire thicknesses at different rates
    ▪ To improve delay, helps to keep R down, therefore make \( W \times H \) as large as possible –
      aim for high aspect ratio as this also improves packing density
    ▪ However, helps performance, hurts capacitance

**Industry Outlook from ITRS:**

- Industry very concerned with power
  o Added metric of (Watts per GHz of frequency) / cm²
- Some predict this metric will plateau as technology scales
  o Advent of new materials, low k dielectrics will help
  o History here … there was an Al → Cu transition owing to the lower \( \rho \) of copper compared to Al
    ▪ However, not many material lower than Cu – Ag (1.59 x 10⁻⁸ \( \Omega \) m)?
- Also, problems could get worse
  o The number of metal layers has increased as technology scales (see slides)
  o Therefore, volume, capacitance of IC could increase
- Alternative technologies being investigated and will be discussed:
  o RF, optical, CNTs, 3D…
Recap:

(First, quick review of EDP, PDF performance metrics – from Lecture 03)

Dynamic power:

- Energy stored on capacitor:
  \[ E_C = \int_0^\infty V_{dd}(t) V_{out}(t) dt = V_{dd} \int_0^\infty C_L \frac{dV_{out}}{dt} V_{out} dt = C_L \int_0^\infty V_{out} dV_{out} = \frac{C_L V_{dd}^2}{2} \]

- Power dissipation – from charging, discharging capacitor
  \[ P_{dyn} = C_L V_{dd}^2 f \]

Direct path power:

- Direct path energy a function of the time that both NMOS, PMOS devices are conducting:
  \[ E_{direct \ path} = V_{dd} \frac{i_{peak} t_{sc}}{2} + V_{dd} \frac{i_{peak} t_{sc}}{2} = V_{dd} i_{peak} t_{sc} \]

- Therefore the power dissipation associated with direct path currents is given by:
  \[ P_{direct \ path} = V_{dd} i_{peak} t_{sc} f \]

Leakage power:

- Sub-threshold Leakage:
  \[ I_{sub} = K_1 W e^{\frac{-V_t}{V_d}} (1 - e^{\frac{-V}{V_d}}) \]

- Gate Leakage:
  \[ I_{ox} = K_2 W \left( \frac{V}{t_{ox}} \right)^2 e^{\frac{-t_{ox}}{V}} \]

To summarize…

\[ P_{total} = P_{dynamic} + P_{directpath} + P_{static} = (C_L V_{dd}^2 + V_{dd} i_{peak} t_{sc}) f + V_{dd} I_{leak} \]

What if we consider all of the above “simultaneously”?

1. If W, L decrease, (a) latency, (b) dynamic power, (c) density all improve.
   a. Not so easy to make W, L smaller
      i. Photolithography has some fundamental limitations (wavelength of UV light = 250 nm)
      ii. New candidates for further transistor scaling include EUV, imprint
      iii. The wavelength of light is what it is.

   This challenge has (so far) been met

b. \( t_{ox} \) must scale as well
   i. Layers less than 4 atoms thick difficult to reliably manufacture
   ii. With thin layers, electrons tunnel and get gate leakage current that results in static power dissipation

   Need new material – and one was found that enabled the 45 nm technology node

c. As device dimensions scale down, lithography is less precise – results in an increase in defects
i. Must scrap die  
ii. Or find architectural alternatives such that we can live with defects

2. If $V_{dd}$ decreases, power decreases  
   a. Decreasing $V_{dd}$ is the best way to lower $P$ given the quadratic dependence on $V_{dd}$  
   b. Problems:  
      i. $V$ already $\sim 0.9V - 1V$  
      ii. Could realistically go to $\sim 0.5V$  
      iii. Noise, other sources become issues  
   c. Also, need to lower $V_t$  
      i. If $V_{dd}$ reduced to 0.5V, only 0.5V between logic ‘1’ and logic ‘0’ (i.e. smaller margins)  
      ii. Also, $V_t$ determined (in part) by the number of atoms / concentration of dopant atoms; as feature size decreases, dopant concentration can experience “wide” swings  
      iii. If $V_t$ varies between 0.1 and 0.3 C, could be problematic  
   d. Oh, and performance decreases too

3. If $V_{dd}$ increases, $f$ goes up (but $P_{\text{dyn}}$ goes up in 2 ways – $V_{dd}$, $f$)

4. Lest we forget, a decrease in $W$, $L$ = an increase in the net number of devices

5. Up against practical limits  
   a. Could deal with $> 100$ W / cm$^2$ $\rightarrow$ not an engineering problem  
   b. Instead, it’s a practical problem $\rightarrow$ 100 W/cm$^2$ = practical limit of air cooling

(A big) solution to the issues outlined above is multi-core chips – let’s look at how they are affected by … interconnect

- Discussion based on “Design Tradeoffs for Tiled CMP On-Chip Networks” by Balfour and Dally  
  - Supercomputing 2007  
- Design issues brought up here equally relevant to other emerging technologies too…

Consider the following “sea” of processor cores:

Let’s look inside of a router first…
- Router has 2 main components:
  1. Datapath:
     o Handles storage and movement of a packet’s payload
     o Consists of input buffers, switch, & output buffers
  2. Control
     o Logic to coordinate packet resource allocation

- I’m going to talk about a “Virtual Channel Router”
  o Virtual channel router requires extra resources (HW), but can help overcome blocking issues
    - (Might see blocking issues with wormhole routing)
    - (VC allows packets to pass a blocked packet and make better use of idle bandwidth)

Example:
  1. Packet B enters node #1 from the network; B acquires channel p from node #1 \(\rightarrow\) node #2
  2. A 2\textsuperscript{nd} packet A has entered node #1 from the wst and needs to be routed east to node #3
  3. Meanwhile, B wants to leave node #2 and go south, but is blocked
  4. Now channels p and q are idle .. but cannot be used
     a. Packet A is blocked in node #1
     b. It cannot acquire channel p
     c. B blocks

**Figure**: Packet Routing

Now, assume 2 VCs per physical channel:
  1. B arrives at node #1 and acquires the bandwidth to go to channel p
  2. A arrives from the east, B tries to leave node #2 and is blocked
  3. A can use free bandwidth p and goto another VC on node #2
  4. Can also proceed onto node #3

This is a better use of resources
  - May have 1 physical channel, but more buffers

**What happens during packet routing?**

1. Let’s start with a flit of a packet arriving at the input unit of a router
   o Input unit consists of a flit buffers to hold arriving flits until they can be forwarded
   o Input unit also maintains state of virtual channel
     i. I: Idle
     ii. R: Routing
     iii. V: Waiting for virtual channel
     iv. A: Active
   o Once packet in router, heed to perform route computation to see where it goes; can then go to VC for allocation

2. Each head flit must advance through 4 stages of routing computation
   o It’s pipelined! Assume…
     o RC: Routing Computation
     o VA: Virtual Channel Allocation
     o SA: Switch Allocation
     o ST: Switch Traversal
Packet might move through like this:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head Flit</td>
<td>RC</td>
<td>VA</td>
<td>SA</td>
<td>ST</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Body Flit 1</td>
<td>**</td>
<td>SA</td>
<td>ST</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Body Flit 2</td>
<td></td>
<td>SA</td>
<td>ST</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tail Flit</td>
<td></td>
<td></td>
<td></td>
<td>SA</td>
<td>ST</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

** (second body flit arrives)

Important Points:
- $t_r$ (time through a single router) does not equal 1!
  - (more like 5 or 6 at least)
- Routing and VC allocation are per packet functions
  - Nothing for body flits to do
  - With no stalls, need 3 input buffers (for 3 flits)
  - With stalls, need # of buffers = # of packets

Outlook:
- Ultimately, issues involved in routing process discussed above + router architecture + storage needed determine the bandwidth for the topology
  - Possibilities:
    - Even though you can devise a topology for ideal performance, it may not be feasible to implement
    - Or, 1 part may be technologically feasible (pitch) but another may not be (router or buffer)

Why can routers be hard to implement?

**Figure**: Possible router design in 8 metal layer chip

Consider how connections would actually be made on chip:
- *Discuss metal stack*
- *Show cross-sectional die photo*
- *Draw lines for input and output*

Now, let's go back to our picture and made some observations:
1. No lines of the same color can touch (it would be an electrical short)
2. We draw 1 line, but really many (1 line for each bit)
3. Router areas are by no means insignificant!

How can on-chip IC NWs affect performance?
Want to know – for a given IC NW topology – how long it takes to send a message:

- Note → initial #s in the absence of contention → a bit more on this later

Time: 

\[(\text{# of hops}) \times (\text{time in router}) + \text{time required for packet to traverse all channels } \]

\[+ \text{serialization latency}\]

(serialization latency = ceiling(length of message / bandwidth))

Therefore, if:

- Average # of hops \(= 6.25\)
- Average time for packet to traverse all channels \(= 5.3333\)
- Serialization latency \(= 3\)
- Time in router \(= 2\)
- Total time: \(= \sim 20.8\)

Slides:

- Results from Dally, Balfour paper
- Impact in the context of Amdahl’s Law
- Information processing tokens