Simultaneous Supply, Threshold and Width Optimization for Low-Power CMOS Circuits

With an aside on System based shutdown.

Gord Allan
PhD Candidate
ASIC Design
**Factors vs Power and Delay**

### Changing Vt
- **Threshold**

### Changing VDD
- **Supply**

### Changing Width
- **W**

### Changing Threshold
- **Vt**

**Delay**

\[ Delay = A \frac{C}{(V*K)} \]

**P\text{\_load}**

\[ P_{\text{load}} = C*V^2 \]

**P\text{\_sc}**

\[ P_{\text{sc}} = f(t_{\text{tr}})*f(V^3)*f(K) \]

**P\text{\_stat}**

\[ P_{\text{stat}} = I_{\text{leak}}*V \]

\[ A = 1/(1-n) * [2n/(1-n) + \ln(3-4n)], \quad n = Vt/VDD \]
Reducing $V_t$ and $VDD$

- Lower $V_t$ so that we can reduce $VDD$ further
- Ideal when Power Dynamic = Power Static

Example: - 600 gates - 18 gates deep
- Critical Delay = 3.3 ns

<table>
<thead>
<tr>
<th>$V_t$</th>
<th>$VDD$</th>
<th>Static</th>
<th>Dyn</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>2.5V</td>
<td>.03aW</td>
<td>260fW</td>
<td>260fW</td>
</tr>
<tr>
<td>0.13</td>
<td>0.4V</td>
<td>9 fW</td>
<td>6 fW</td>
<td>15fW</td>
</tr>
</tbody>
</table>
An Aside: System Shutdown

If you’re not using something - TURN it OFF!

If it doesn’t need speed, TURN it DOWN!


But When and How?

• When - Using user history to predict best time to sleep
  - There are costs (time and power) to go to sleep/awake
  - Voltage regulation problems.

• How - Hard Enough in normal circuits (clock gating, supply red)
  - What to do when dynamic power is an issue?
  (cutting off VDD is tough and poses other problems)
Back to Optimization

How do we pick VDD, Vt, and Ws given a clk freq?

- Draw Circuit
- Assign Delay Estimates (based on fan-out)
- Find Critical Path
- Assign Maximum Delays (weighted to estimates) along most critical path
- Assign Maximum Delays to other gates on next most critical path, etc...

\[ 4 \times \frac{2}{3} = 2.7 \text{ nS} \]
\[ 1.3 \text{ nS} \]
\[ 2 \text{ ns} \]
\[ 4 \text{ nS} \]
\[ 2 \text{ ns} \]
\[ 1 \text{ nS} \]
\[ 1 \text{ nS} \]

Eg. \( T_{\text{clk}} = 8 \text{ ns} \)
Pseudo-Code Procedure

```
for(VDD from 0.1 to 3.3)
    for(Vt from 0.1 to 0.7)
        for each gate
            for(W from 1 to 100)
                calculate delay
                if lower than Dmax pick W
                calculate total power dissipation
```

- Gives optimal VDD, Vt and W for all gates such that timing is met. NB: Pwr - f(switching activity)
- Complexity depends on number of gates and quantization of parameters
- Binary Search technique is used for large circuits
Results

Circuit A: s349 - 226 gates, depth 28
   \( \alpha = 0.5 \)
   Benchmark: Vdd=3.3V, Vt=0.7V
   Optimum: Vdd=0.7V, Vt=0.1V
   Power Savings of 54x

Circuit B: s526 - 596 gates, depth 18
   \( \alpha = 0.005 \)
   Benchmark: Vdd=2.5V, Vt=0.7V
   Optimum: Vdd=0.4V, Vt=0.13V
   Power Savings of 18x
Problems for Future Work

Perception - Static power is Bad!

Shutdown

Variation of low Vt due to process issues.
- Drags improvements from 20x to 6x with 50% variation
- How do we get a reliable and efficient low Vt?

System on a chip supplies - want > 1V. Noise issues.

Variation of Switching activities and their effect.

Low Vt - Great for pass logic! Exploit the benefits.

Multiple Vt circuits - advantages?
Primary References


### Appendix: More Results

**Circuit A: s349 - 226 gates, depth 28**  
Benchmark: Vdd=3.3V, Vt=0.7V  
- Optimum 1: Vdd=0.7V, Vt=0.1V  
  - Savings of: Power 54x  
  - Area of 64%  
- Optimum 2: Vdd=0.6V, Vt=0.1V  
  - Savings of: Power 27x  
  - Area of 59%

**Circuit B: s526 - 596 gates, depth 18**  
Benchmark: Vdd=2.5V, Vt=0.7V  
- Optimum 1: Vdd=0.3V, Vt=0.1V  
  - Savings of: Power 67x  
  - Area of 8%  
- Optimum 2: Vdd=0.4V, Vt=0.13V  
  - Savings of: Power 18x  
  - Area of 20%