Previously

- Instruction Space
- Modeling (Area, Energy) from Architecture
- Application Structure
- Mismatch
  - Net area, energy
  - Efficiency
- Example: $W_{\text{task}}$ vs. $W_{\text{arch}}$

Today

- Application vs. Architecture
  - Mismatch
  - Net area, energy
  - Efficiency
  - Path Length vs. Context Depth
  - Impact of multiple dimensions

Context Depth

Two Application Drivers

- Latency goal
- Throughput goal
  - E.g. operate in real time

Throughput Goals

- Examples of application throughput goals we might have?
Throughput Goal

• Have 1GHz datapath (1 ns operator)
• Want to compute one result every 10ns
• How many operations can it perform per bit operator in the 10ns period?

Path Length: How much sequentialization is allowed?

Latency Goal Reuse

Path Length: How much sequentialization is allowed?

Latency Path Length

• How many primitive-operator delays before can perform next operation?
  – Reuse the resource

Critical Path

• Single cycle multiply and add
  \[ s_i = a_i * q_{i-1} + (1-a_i) * x_i \]
  \[ t_i = b_i * r_{i-1} + (1-b_i) * x_i \]
  \[ q_i = q_{i-1} + s_i \]
  \[ r_i = r_{i-1} + t_i \]

• What is critical path?

Application Needs

• Common critical paths?

<table>
<thead>
<tr>
<th>Application</th>
<th>Wapp</th>
<th>Lcritpath</th>
<th>Latath</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conway</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Run as fast as possible</td>
</tr>
<tr>
<td>LIFE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECC</td>
<td>1</td>
<td>1-10</td>
<td>100</td>
<td>100ns memory interface</td>
</tr>
<tr>
<td>Video</td>
<td>8</td>
<td>1-6</td>
<td>24</td>
<td>1GHz for 1024x1024 x30 frames/s</td>
</tr>
<tr>
<td>Audio</td>
<td>16</td>
<td>1-10</td>
<td>20,000</td>
<td>44KHz for 1GHz</td>
</tr>
<tr>
<td>FDTD</td>
<td>35</td>
<td>1-5</td>
<td>1-5</td>
<td></td>
</tr>
</tbody>
</table>
Model Area

\[ A_{\text{bit.elm}} = A_{\text{fixed}} + N_{\text{SW}}(N_{p,w}, p) \cdot A_{\text{SW}} + \left( \frac{c}{w} \right) \cdot n_{\text{bits}} \cdot A_{\text{mem.cell}} + d \cdot A_{\text{mem.cell}} \]

Relative Sizes

- Bit Operator: 4K\(F^2\)
- Bit Operator Interconnect: 240K\(F^2\)
- Instruction (with interconnect): 20K\(F^2\)
- Memory bit (SRAM): 300\(F^2\)

Efficiency with fixed Width

\(w=1, \quad 16K \text{ PEs}\)

Robust Point for C

- \(A_{\text{bit.elm}} = 240K + C \cdot 30K\)
- Given compute model:
  - What is area efficient robust point for C?
- HW5.2c
  - "energy competitive" = robust point

Ideal Efficiency (different model)

Two resources here:
- active processing elements
- operation description/state

Applications need in different proportions.
Technology Change

- \( A_{\text{bitelm}} = 240K + C \times 30K \)
- What happens to the robust point if we have a technology change that reduces the cost of memory cells?
  - E.g. from 300F^2 SRAM cells to 100F^2 DRAM cells (logic process)?

Multiple Parameters

C Robust Point depends on Width

Model Area

- \( A_{\text{bitelm}} = A_{\text{fixed}} + \frac{N_{SW}(N_{P, w, p}) \cdot A_{SW}}{w} \cdot \frac{C}{w} \cdot n_{\text{bits}} \cdot A_{\text{mem.cell}} \) + \( d \cdot A_{\text{retiming.memory}} \)

Robust Point for C given W

- \( A_{\text{bitelm}} = 240K + (C/W) \times 30K \)
- Given compute model:
  - What is area efficient robust point for C:
    - When \( W=8 \)?
    - When \( W=64 \)?

Robust Point depend on Width
Intermediate Architecture

- $w=8$
- $c=64$
- 16K PEs

Hard to be robust across entire space...

Processors and FPGAs
(architecture vs. two application axes)

- FPGA
  - $c=d=1$, $w=1$, $k=4$
- "Processor"
  - $c=d=1024$, $w=64$, $k=2$

Caveats

- Model abstracts away many details that are important
  - interconnect (day 16—20, 25)
  - control (day 23)
  - specialized functional units (day 15)
- Applications are a heterogeneous mix of characteristics

Modeling Message

- Architecture space is huge
- Easy to be very inefficient
  - Valuable to understand to pick right point
- Hard to pick one point robust across entire space
- Why we have so many architectures?

General Message

- Parameterize architectures
- Look at continuum
  - costs
  - benefits
- Often have competing effects
  - leads to maxima/minima

Big Ideas
[MSB Ideas]

- Instruction organization induces a design space (taxonomy) for programmable architectures
- Arch. structure and application requirements mismatch ⇒ inefficiencies
- Model ⇒ visualize efficiency trends
- Architecture space is huge
  - can be very inefficient
  - need to learn to navigate
Admin

- Homework
  - Marked 4.1
  - TODO 4.2, 4.3
- HW5.1 Due Today
- Reading
- Spring Break