## Paging: Faster Translations

### CPU

<table>
<thead>
<tr>
<th>V</th>
<th>VPN</th>
<th>PFN</th>
<th>ASID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>7</td>
<td>2</td>
</tr>
</tbody>
</table>

### Context Switch

1. Push
2. ASID

### Address Translation Cache

- TLB Hit
- pid

---

Memory

---

16, 32, 64, 128, 256
H/W: steps on each mem. ref (WITHOUT TLB).

1. extract VPN from the V.A.

\[ \text{VPN offset} \]

2. Get the PTE from the PT.

\[ \text{PTEAddr} = PTBR + \text{VPN x Size of (PTE)} \]

3. \[ \text{Access Memory (PTE)} \] EXPENSIVE

4. Get the PFN from PTE.

5. Form the PA = PFN + offset.

6. \[ \text{Reg} \leftarrow \text{Access Memory (PA)} \]
H/W: on each mem. ref w/ TLB

1. extract VPN from the VA.
2. Lookup TLB to see if the entry for VPN is present in the TLB.
3. if yes \[\text{TLB Hit}\]
   - get thePFN from the TLB.
   - form the PA.
   - Access Memory (PA)
4. if no \[\text{TLB Miss}\]
   - get the PTE from the PT.
   - Memory Access
   - updated the TLB with the PTE.
   - retry the instr. again!

int num = X, MOV 80, l, eax.
<table>
<thead>
<tr>
<th></th>
<th>H/W</th>
<th>OS-managed TLB</th>
<th>OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>same</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>If no TLB Miss</td>
<td>Raise (TLB Miss Exception)</td>
<td></td>
</tr>
</tbody>
</table>

Try runs the TLB Miss exception handler:
- Get PTE from PT.
- Updates the TLB with this translation using special instr. privileged instr.
  - Retries the current instruction.
int array [10];

hit rate = 70%